

oVirt 3.0

Installation Guide

Installing and Configuring an oVirt Environment

Stephen Gordon

Tim Hildred

oVirt 3.0 Installation Guide

Installing and Configuring an oVirt Environment

Edition 1

Author Stephen Gordon
Author Tim Hildred

Copyright © 2012 oVirt Project.

Licensed under the Apache License, Version 2.0 (the "License"). A copy of the License is included in this documentation; in addition, you may obtain a copy of the License at

<http://www.apache.org/licenses/LICENSE-2.0>

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

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

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

All other trademarks are the property of their respective owners.

Preface	v
1. About this Guide	v
1.1. Audience	v
1.2. Overview	v
2. Document Conventions	v
2.1. Typographic Conventions	v
2.2. Pull-quote Conventions	vii
2.3. Notes and Warnings	vii
3. Getting Help and Giving Feedback	viii
3.1. Do You Need Help?	viii
3.2. We Need Feedback!	viii
I. Before you Begin	1
1. Overview	3
1.1. System Components	3
1.1.1. About the Virtual Machines	4
1.1.2. About SPICE	4
1.2. Installation Workflow	4
2. System Requirements	7
2.1. Hardware requirements	7
2.1.1. Management Server Requirements	7
2.1.2. Virtualization Host Requirements	8
2.2. Software requirements	10
2.2.1. Client Requirements	10
2.2.2. Directory Services	11
2.2.3. Firewall Configuration	11
II. Installing oVirt Engine	15
3. Engine Installation	17
3.1. Installation	18
3.2. Configuration	19
3.3. Connect to the Administration Portal	23
III. Installing Virtualization Hosts	25
4. Introduction to Virtualization Hosts	27
5. Installing the oVirt Node	29
5.1. Installation Media	29
5.1.1. Preparation Instructions	29
5.1.2. Preparing a Node USB Storage Device	31
5.1.3. Preparing a Node from a CD-ROM or DVD	35
5.2. Installation	36
5.2.1. Interactive Installation	37
5.3. Configuration	40
5.3.1. Logging In	40
5.3.2. Status	41
5.3.3. Network	41
5.3.4. Security	44
5.3.5. Logging	44
5.3.6. Kernel Dump	45
5.3.7. Remote Storage	46

5.3.8. oVirt-M	46
5.4. Using the Node	47
6. Installing VDSM	49
6.1. Installing VDSM From RPMs	49
6.2. To Add a Host	52
6.3. Activating a Host	55
IV. Environment Configuration	57
7. Planning your Data Center	59
7.1. Data Centers	59
7.1.1. Prerequisites for Setting up a Data Center	59
7.1.2. Working with Data Centers	60
7.1.3. Creating a New Data Center	61
7.2. Clusters	62
7.2.1. Creating a Cluster	63
8. Network Setup	65
8.1. Determine Network Requirements	65
8.2. Logical Networks	65
8.2.1. Adding Logical Networks	66
8.3. Set Up a Bond Device	69
9. Storage Setup	73
9.1. Storage Domains Overview	74
9.1.1. Adding NFS Storage	74
9.1.2. Adding iSCSI Storage	77
9.1.3. Adding FCP Storage	81
9.1.4. Adding Local Storage	82
9.2. Populate the ISO Domain	83
9.2.1. Uploading the VirtIO and Guest Tool Image Files	84
V. Appendices	85
A. Directory Services	87
A.1. IPA Server	87
A.1.1. Adding New Users	87
A.2. Active Directory	88
B. Additional Utilities	91
B.1. Domain Management Tool	91
B.1.1. Syntax	91
B.1.2. Examples	92
B.2. Configuration Tool	93
B.2.1. Syntax	94
B.2.2. Examples	95
B.3. ISO Uploader	95
B.3.1. Syntax	95
B.3.2. Examples	97
B.4. Log Collector	97
B.4.1. Syntax	97
B.4.2. Examples	100
C. Revision History	103

Preface

The oVirt platform is a richly featured virtualization management solution providing fully integrated management across virtual machines. It is based on the leading open source virtualization platform and provides superior technical capabilities. The platform offers scalable management of large numbers of virtual machines.

1. About this Guide

1.1. Audience

This guide is intended to provide detailed instruction on the installation and configuration of oVirt. A background in the administration of systems running Linux is recommended. Experience with previous versions of oVirt, while beneficial, is not required.

1.2. Overview

Completion of the procedures outlined within this guide will result in a working oVirt environment. The environment produced will be configured to the level such that it can be used to create, run, and manage virtual machines.

2. Document Conventions

This manual uses several conventions to highlight certain words and phrases and draw attention to specific pieces of information.

In PDF and paper editions, this manual uses typefaces drawn from the *Liberation Fonts*¹ set. The Liberation Fonts set is also used in HTML editions if the set is installed on your system. If not, alternative but equivalent typefaces are displayed. Note: Red Hat Enterprise Linux 5 and later includes the Liberation Fonts set by default.

2.1. Typographic Conventions

Four typographic conventions are used to call attention to specific words and phrases. These conventions, and the circumstances they apply to, are as follows.

Mono-spaced Bold

Used to highlight system input, including shell commands, file names and paths. Also used to highlight keycaps and key combinations. For example:

To see the contents of the file **my_next_bestselling_novel** in your current working directory, enter the **cat my_next_bestselling_novel** command at the shell prompt and press **Enter** to execute the command.

The above includes a file name, a shell command and a keycap, all presented in mono-spaced bold and all distinguishable thanks to context.

Key combinations can be distinguished from keycaps by the hyphen connecting each part of a key combination. For example:

¹ <https://fedorahosted.org/liberation-fonts/>

Press **Enter** to execute the command.

Press **Ctrl+Alt+F2** to switch to the first virtual terminal. Press **Ctrl+Alt+F1** to return to your X-Windows session.

The first paragraph highlights the particular keycap to press. The second highlights two key combinations (each a set of three keycaps with each set pressed simultaneously).

If source code is discussed, class names, methods, functions, variable names and returned values mentioned within a paragraph will be presented as above, in **mono-spaced bold**. For example:

File-related classes include **filesystem** for file systems, **file** for files, and **dir** for directories. Each class has its own associated set of permissions.

Proportional Bold

This denotes words or phrases encountered on a system, including application names; dialog box text; labeled buttons; check-box and radio button labels; menu titles and sub-menu titles. For example:

Choose **System** **Preferences** **Mouse** from the main menu bar to launch **Mouse Preferences**. In the **Buttons** tab, click the **Left-handed mouse** check box and click **Close** to switch the primary mouse button from the left to the right (making the mouse suitable for use in the left hand).

To insert a special character into a **gedit** file, choose **Applications** **Accessories** **Character Map** from the main menu bar. Next, choose **Search** **Find...** from the **Character Map** menu bar, type the name of the character in the **Search** field and click **Next**. The character you sought will be highlighted in the **Character Table**. Double-click this highlighted character to place it in the **Text to copy** field and then click the **Copy** button. Now switch back to your document and choose **Edit** **Paste** from the **gedit** menu bar.

The above text includes application names; system-wide menu names and items; application-specific menu names; and buttons and text found within a GUI interface, all presented in proportional bold and all distinguishable by context.

Mono-spaced Bold Italic or ***Proportional Bold Italic***

Whether mono-spaced bold or proportional bold, the addition of italics indicates replaceable or variable text. Italics denotes text you do not input literally or displayed text that changes depending on circumstance. For example:

To connect to a remote machine using ssh, type **ssh *username@domain.name*** at a shell prompt. If the remote machine is **example.com** and your username on that machine is john, type **ssh *john@example.com***.

The **mount -o remount *file-system*** command remounts the named file system. For example, to remount the **/home** file system, the command is **mount -o remount */home***.

To see the version of a currently installed package, use the **rpm -q *package*** command. It will return a result as follows: ***package-version-release***.

Note the words in bold italics above — *username*, *domain.name*, *file-system*, *package*, *version* and *release*. Each word is a placeholder, either for text you enter when issuing a command or for text displayed by the system.

Aside from standard usage for presenting the title of a work, italics denotes the first use of a new and important term. For example:

Publican is a *DocBook* publishing system.

2.2. Pull-quote Conventions

Terminal output and source code listings are set off visually from the surrounding text.

Output sent to a terminal is set in **mono-spaced roman** and presented thus:

```
books      Desktop  documentation  drafts  mss    photos  stuff  svn
books_tests Desktop1  downloads      images  notes  scripts svgs
```

Source-code listings are also set in **mono-spaced roman** but add syntax highlighting as follows:

```
package org.jboss.book.jca.ex1;

import javax.naming.InitialContext;

public class ExClient
{
    public static void main(String args[])
        throws Exception
    {
        InitialContext iniCtx = new InitialContext();
        Object          ref    = iniCtx.lookup("EchoBean");
        EchoHome        home   = (EchoHome) ref;
        Echo             echo   = home.create();

        System.out.println("Created Echo");

        System.out.println("Echo.echo('Hello') = " + echo.echo("Hello"));
    }
}
```

2.3. Notes and Warnings

Finally, we use three visual styles to draw attention to information that might otherwise be overlooked.



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.

3. Getting Help and Giving Feedback

3.1. Do You Need Help?

The oVirt Project hosts electronic mailing lists for discussion of oVirt software and technology. You can find a list of these mailing lists at <http://www.ovirt.org/project/community/>. Click on the name of any mailing list to subscribe to that list or to access the list archives.

3.2. We Need Feedback!

If you find a typographical error in this manual, or if you have thought of a way to make this manual better, we would love to hear from you!

Please submit bug reports at https://bugzilla.redhat.com/enter_bug.cgi?product=oVirt&component=doc-Installation-Guide.

If you have a suggestion for improving the documentation, try to be as specific as possible when describing it. If you have found an error, include the section number and some of the surrounding text so we can find it easily.

Part I. Before you Begin

Overview

oVirt provides IT departments with the tools to meet the challenges of managing complex environments. oVirt's state-of-the-art virtualization platform enables administrators to reduce the cost and complexity of large deployments. oVirt platform provides:

- High availability to quickly configure virtual machines for fault tolerance.
- Live migration to move virtual machines between physical hosts without interruption.
- System scheduler to create policies to dynamically balance compute resources.
- Power saver to create policies to conserve power and cooling costs.
- Image engine to create, manage and provision virtual machines.
- Storage virtualization to consistently access common storage from any server.
- Multi-level administration to enable administration of physical infrastructure as well as administration of virtual objects.
- Ability to convert existing virtual machines on foreign nodes to oVirt platform.
- A range of reports either from the reports module based on JasperReports, or from the data warehouse. The reports enable administrators to monitor and analyze information on virtual machines, hosts and storage usage and performance.

1.1. System Components

The oVirt platform consists of one or more hosts (either oVirt Nodes or pre-configured Linux systems on which *VDSM* is installed) and at least one engine. The virtual machines are run on the hosts. The system and all its components are managed through a centralized management system.

oVirt Engine

The oVirt Engine acts as a centralized management system that allows system administrators to view and manage virtual machines and images. The oVirt Engine provides a comprehensive range of features including search capabilities, resource management, live migrations and provisioning..

The engine provides a graphical user interface to administer the physical and logical resources within the virtual environment infrastructure. It can be used to manage provisioning, connection protocols, user sessions, virtual machine pools, images and high availability/clustering. The oVirt Engine exposes an Administration Portal, a User Portal, and an Application Programming Interface (API).

- The Administration Portal is used to perform setup, configuration, and management of the oVirt environment.
- The User Portal is used to start, stop, reboot, and connect to virtual machines. Users granted power user access by the environment's administrators are also able to create virtual machine templates and, virtual machines from this interface.
- The REST API provides an interface for automation of tasks normally accomplished manually by users. Scripts that make use of the REST API are able to be written in any language which supports accessing HTTP and HTTPS resources.

oVirt Node(s)

The oVirt Node is a fully featured virtualization platform for quick, easy deployment and management of virtualized guests. The Node is designed for management via the oVirt Engine. It provides a thin virtualization layer deployed across the server infrastructure.

Kernel-based Virtual Machine (KVM), which is a core component of the Linux kernel, is used to provide virtualization.

Pre-configured Linux Host(s)

The oVirt Engine also supports the use of pre-configured systems running Linux with support for *vdsm* as virtualization hosts.

1.1.1. About the Virtual Machines

oVirt platform enables you to create virtual machines that perform the same functions as physical machines. Using a standard Web browser, users can run virtual machines that behave like physical desktops. Multiple levels of permissions allow users with different roles to manage virtual machines to meet the requirements of the enterprise.

Supported Guests

oVirt is presently tested with the following virtualized guest operating systems, results with other guest operating systems may vary:

- Red Hat Enterprise Linux 3 (32 bit and 64 bit)
- Red Hat Enterprise Linux 4 (32 bit and 64 bit)
- Red Hat Enterprise Linux 5 (32 bit and 64 bit)
- Red Hat Enterprise Linux 6 (32 bit and 64 bit)
- Windows XP Service Pack 3 and newer (32 bit only)
- Windows 7 (32 bit and 64 bit)
- Windows Server 2003 Service Pack 2 and newer (32 bit and 64 bit)
- Windows Server 2008 (32 bit and 64 bit)
- Windows Server 2008 R2 (64 bit only)

1.1.2. About SPICE

The SPICE protocol allows the virtual machine to connect to the host with physical PC-like graphics performance. It supplies video at more than 30 frames per second, bi-directional audio (for soft-phones/IP phones), bi-directional video (for video telephony/video conferencing) and USB redirection from the client's USB port into the virtual machine. SPICE also supports connection to multiple monitors with a single virtual machine.

1.2. Installation Workflow

oVirt requires installation and configuration of several components to create a functioning virtualization environment. You must install and configure each component in the order shown in the checklist that follows:

Check System Requirements

- Check hardware requirements, as seen in [Section 2.1, "Hardware requirements"](#).

- Check software requirements, as seen in [Section 2.2, “Software requirements”](#).

oVirt Engine Installation

- Install oVirt Engine, as seen in [Chapter 3, Engine Installation](#).

Install Virtualization Hosts

- Install oVirt Node hosts, as seen in [Chapter 5, Installing the oVirt Node](#).

Plan Your Data Center

- Plan your oVirt Data Center(s), as seen in [Chapter 7, Planning your Data Center](#).

Setup Networks

- To configure networks, see [Chapter 8, Network Setup](#).

Setup Storage

- For an overview of the supported storage types, and how they are attached to the engine, see [Section 9.1, “Storage Domains Overview”](#).
- To populate your ISO storage domain with installation media, see [Section 9.2, “Populate the ISO Domain”](#).

Completion of the above steps will result in the creation of a functioning oVirt environment capable of running virtual machines.

System Requirements

This chapter outlines the hardware and software requirements for installing the oVirt platform. The requirements outlined herein are based on the minimum requirements for successful installation, configuration, and operation of a oVirt environment.

Production oVirt installations will have additional requirements in line with the relative workloads expected of them. The additional requirements for your specific implementation of the oVirt product need to be determined by your solution architect. Guidance on planning your oVirt environment is provided in [Chapter 7, Planning your Data Center](#).



Important — Listed Requirements are Mandatory

All listed requirements must be met *before* installation commences. Without the listed requirements installation of a fully functional oVirt environment as described in this guide will *not* be possible.

Before commencing the installation you must consult the latest version of the oVirt Release Notes, available at http://www.ovirt.org/wiki/Release_Notes.

2.1. Hardware requirements

This section outlines the minimum hardware required to install, configure, and operate a oVirt environment. To setup a oVirt environment it is necessary to have, at least:

- one machine to act as the management server,
- one or more machines to act as virtualization hosts - at least two are required to support migration and power management,
- one or more machines to use as clients for accessing the Administration Portal.
- storage infrastructure provided by NFS, iSCSI, SAN, or local storage.

The hardware required for each of these systems is further outlined in the following sections. The oVirt environment also requires storage infrastructure that is accessible to the virtualization hosts. Storage infrastructure must be accessible using NFS, iSCSI, FC, or locally attached to virtualization hosts.

2.1.1. Management Server Requirements

The minimum and recommended hardware requirements outlined here are based on a typical small to medium sized installation. The exact requirements vary between deployments based on sizing and load. Please use these recommendations as a guide only.

Minimum

- A dual core CPU.
- 4 GB of available system RAM that is not being consumed by existing processes.
- 25 GB of locally accessible, writeable, disk space.
- 1 Network Interface Card (NIC) with bandwidth of at least 1 Gbps.

Recommended

- A quad core CPU or multiple dual core CPUs.
- 16 GB of system RAM.
- 50 GB of locally accessible, writeable, disk space.
- 1 Network Interface Card (NIC) with bandwidth of at least 1 Gbps.

2.1.2. Virtualization Host Requirements

oVirt Nodes and Pre-configured Linux Hosts have a number of hardware requirements and supported limits.

2.1.2.1. CPU Requirements

Virtualization hosts must have at least one CPU. All CPUs must support

- the Intel® 64 or AMD64 CPU extensions, and
- the AMD-V™ or Intel VT® hardware virtualization extensions.

Additionally a maximum of 128 physical CPUs per virtualization host is currently supported. To check that your processor supports the required virtualization extensions, and that they are enabled:

- At the oVirt Node boot screen press any key and select the **Boot** or **Boot with serial console** entry from the list. Press **Tab** to edit the kernel parameters for the selected option. After the last kernel parameter listed ensure there is a **Space** and append the **rescue** parameter.
- Press **Enter** to boot into rescue mode.
- At the prompt which appears, determine that your processor has the virtualization extensions and that they are enabled by running this command:

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

If any output is shown, the processor is hardware virtualization capable. If no output is shown it is still possible that your processor supports hardware virtualization. In some circumstances manufacturers disable the virtualization extensions in the BIOS. Where you believe this to be the case consult the system's BIOS and the motherboard manual provided by the manufacturer.

- As an additional check, verify that the **kvm** modules are loaded in the kernel:

```
# lsmod | grep kvm
```

If the output includes **kvm_intel** or **kvm_amd** then the **kvm** hardware virtualization modules are loaded and your system meets requirements.

2.1.2.2. RAM Requirements

Virtualization hosts must have at least 10 GB of RAM. A minimum of an additional 1 GB for each virtual machine is also recommended. The amount of RAM required for each guest varies depending on:

- the guest operating system's requirements,
- the guests' application requirements, and

- memory activity and usage of guests.

The fact that KVM is able to over-commit physical RAM for virtualized guests must also be taken into account. This allows provisioning of guests with RAM requirements greater than physically present on the basis where not all guests will be at peak load concurrently. KVM does this by only allocating RAM for guests as required and shifting underutilized guests into swap.

Additionally a maximum of 1 TB of RAM per virtualization host is currently supported.

2.1.2.3. Storage Requirements

Virtualization hosts require local storage to store configuration, logs, kernel dumps, and for use as swap space. The minimum storage requirements of the oVirt Node are documented in this section. The storage requirements for pre-configured Linux hosts vary based on the amount of disk space used by their existing configuration but are expected to be greater than those of the oVirt Node.

It is recommended that each virtualization host has at least 10 GB of internal storage. The minimum supported internal storage for each Node is the total of that required to provision the following partitions:

- The root partitions require at least 512 MB of storage.
- The configuration partition requires at least 8 MB of storage.
- The recommended minimum size of the logging partition is 2048 MB.
- The data partition requires at least 256 MB of storage. Use of a smaller data partition may prevent future upgrades of the Node from the oVirt Engine. By default all disk space remaining after allocation of swap space will be allocated to the data partition.
- The swap partition requires at least 8 MB of storage. The recommended size of the swap partition varies depending on both the system the Node is being installed upon and the anticipated level of overcommit for the environment. Overcommit allows the oVirt environment to present more RAM to guests than is actually physically present. The default overcommit ratio is **0.5**.

The recommended size of the swap partition can be determined by:

- Multiplying the amount of system RAM by the expected overcommit ratio, and adding
- 2 GB of swap space for systems with 4 GB of RAM or less, or
- 4 GB of swap space for systems with between 4 GB and 16 GB of RAM, or
- 8 GB of swap space for systems with between 16 GB and 64 GB of RAM, or
- 16 GB of swap space for systems with between 64 GB and 256 GB of RAM.

Example 2.1. Calculating Swap Partition Size

For a system with 8 GB of RAM this means the formula for determining the amount of swap space to allocate is:

$$(8 \text{ GB} \times 0.5) + 4 \text{ GB} = 8 \text{ GB}$$



Important — Fkeraid Devices are not Supported

The oVirt Node does not support installation on fkeraid devices. Where a fkeraid device is present it must be reconfigured such that it no longer runs in RAID mode.

1. Access the RAID controller's BIOS and remove all logical drives from it.
2. Change controller mode to be non-RAID. This may be referred to as compatibility or JBOD mode.

Access the manufacturer provided documentation for further information related to the specific device in use.

2.1.2.4. PCI Device Requirements

Virtualization hosts must have at least one network interface with a minimum bandwidth of 1 Gbps. It is recommended that each virtualization host have two network interfaces with a minimum bandwidth of 1 Gbps to support network intensive activity, including virtual machine migration.

2.2. Software requirements



Important — Supported Locales

The oVirt Engine setup script, **engine-setup**, supports the **en_US.UTF-8**, **en_US.utf8**, and **en_US.utf-8** locales. Attempts at installation on systems where the locale in use is not one of these supported values will fail.

2.2.1. Client Requirements

To access the Administration and User Portals, you need a client with a supported web browser. The portals have been tested using the following clients and browsers:

- Client
 - Red Hat Enterprise Linux 5.5 (i386, AMD64 and Intel 64)
 - Red Hat Enterprise Linux 6.0 (i386, AMD64 and Intel 64)
 - Windows XP
 - Windows XP Embedded (XPe)
 - Windows 7 (x86, AMD64 and Intel 64)
 - Windows Embedded Standard 7
 - Windows 2008/R2 (x86, AMD64 and Intel 64)
 - Windows Embedded Standard 2009
 - Linux-based thin clients

- Browser
 - Internet Explorer 7 and higher on Windows, with the SPICE ActiveX control installed
 - Mozilla Firefox 3.5 and higher on Linux, with the SPICE plugin installed

2.2.2. Directory Services

The term directory service refers to the collection of software, hardware, and processes that store information about an enterprise, subscribers, or both, and make that information available to users. A directory service consists of at least one instance of Directory Server and at least one directory client program. Client programs can access names, phone numbers, addresses, and other data stored in the directory service.

The oVirt platform provides its own internal `admin` user. Authentication for other users is supported by attaching directory services domains using the provided domain management tool, **ovirt-manage-domains**. Currently the two supported providers of directory services for use with the oVirt Engine are Identity, Policy, and Audit (IPA) and Microsoft Active Directory.

For more information on configuring directory services see [Appendix A, Directory Services](#).

2.2.3. Firewall Configuration

This section documents the firewall requirements of the oVirt environment. The ports that need to be opened, the type of traffic the port is used for, and the source of traffic which will be received on the port will be covered for the:

- oVirt Engine,
- virtualization hosts, and
- directory server.

While specific configuration instructions for additional network infrastructure which may exist between these systems will not be covered it is intended that the information provided will assist with this task.

2.2.3.1. oVirt Engine Firewall Requirements

The oVirt Engine requires that a number of ports be opened to allow network traffic through the system's firewall. The **engine-setup** script is able to set the required firewall rules automatically. Where an existing firewall configuration exists this step is able to be skipped. This allows the required changes to be manually integrated with the existing firewall script(s).

The firewall configuration documented within this chapter assumes a default configuration. Where you choose alternative values during installation, such as specifying a different HTTP, or HTTPS, port adjust the firewall rules to allow the selected port - not the default listed here.

Table 2.1. oVirt Engine Firewall Requirements

Port(s)	Protocol	Source	Destination	Purpose
22	TCP	<ul style="list-style-type: none"> • System(s) used for maintenance of the engine including backend configuration, and software upgrades. 	<ul style="list-style-type: none"> • oVirt Engine 	SSH (optional)

Port(s)	Protocol	Source	Destination	Purpose
8080, 8443	TCP	<ul style="list-style-type: none"> Administration Portal clients User Portal clients oVirt Node(s) Red Hat Enterprise Linux host(s) REST API clients 	<ul style="list-style-type: none"> oVirt Engine 	Provides HTTP and HTTPS access to the engine.
8006 - 8009	TCP	<ul style="list-style-type: none"> Administration Portal clients 	<ul style="list-style-type: none"> oVirt Engine 	



Important — Additional Ports Required to Export Storage

Where the oVirt Engine is also to export NFS storage, such as an ISO Domain, then additional ports must be allowed through the firewall. The ports used for NFS, which need to be exposed to the Red Hat Enterprise Linux Hosts and oVirt Nodes, are listed in the `/etc/sysconfig/nfs` file:

```
$ cat /etc/sysconfig/nfs
LOCKD_TCPPORT=32803
LOCKD_UDPPORT=32769
MOUNTD_PORT=892
RQUOTAD_PORT=875
STATD_PORT=662
STATD_OUTGOING_PORT=2020
```

2.2.3.2. Virtualization Host Firewall Requirements

It is required that virtualization hosts have a number of ports be opened in their firewalls to allow network traffic. This allows the oVirt Engine to successfully interact with the hosts. In the case of the oVirt Node these firewall rules are configured automatically. For other Linux hosts however it is necessary to manually configure the firewall.

Table 2.2. Virtualization Host Firewall Requirements

Port(s)	Protocol	Source	Destination	Purpose
22	TCP	<ul style="list-style-type: none"> oVirt Engine 	<ul style="list-style-type: none"> Other Hosts 	Secure Shell (SSH) access.
5634 - 6166	TCP	<ul style="list-style-type: none"> Administration Portal clients User Portal clients 	<ul style="list-style-type: none"> oVirt Node(s) Other Host(s) 	Remote guest console access via VNC and Spice. These ports must be open to facilitate

Port(s)	Protocol	Source	Destination	Purpose
				client access to virtual machines.
16514	TCP	<ul style="list-style-type: none"> oVirt Node(s) Other Host(s) 	<ul style="list-style-type: none"> oVirt Node(s) Other Host(s) 	Virtual machine migration using libvirt.
49152 - 49216	TCP	<ul style="list-style-type: none"> oVirt Node(s) Other Host(s) 	Virtual machine migration and fencing using VDSM. These ports must be open facilitate both automated and manually initiated migration of virtual machines.	
54321	TCP	<ul style="list-style-type: none"> oVirt Engine oVirt Node(s) Other Host(s) 	<ul style="list-style-type: none"> oVirt Node(s) Other Host(s) 	VDSM communications with the Manager and other virtualization hosts.

2.2.3.3. Directory Firewall Requirements

oVirt requires a directory server to support user authentication. Currently the supported directory servers are IPA and Active Directory. Both require a number of ports to be opened in the directory server's firewall to support GSS-API authentication as used by the oVirt Engine.

Table 2.3. Host Firewall Requirements

Port(s)	Protocol	Source	Destination	Purpose
88, 464	TCP, UDP	<ul style="list-style-type: none"> oVirt Engine 	<ul style="list-style-type: none"> Directory server 	Kerberos authentication.
389, 636	TCP	<ul style="list-style-type: none"> oVirt Engine 	<ul style="list-style-type: none"> Directory server 	Lightweight Directory Access Protocol (LDAP) and LDAP over SSL.

Part II. Installing oVirt Engine

Engine Installation

Before proceeding with oVirt Engine installation you must ensure that all prerequisites, as listed in [Chapter 2, System Requirements](#), are met. Once you have confirmed that all prerequisites are met you are ready to proceed with installation.

To perform initial installation and configuration of the oVirt Engine follow the steps in [Section 3.1, "Installation"](#). Once you have followed this procedure the oVirt Engine and dependencies, including JBoss (<http://www.jboss.com>), will be installed and ready for your first login. Further action must be undertaken to complete configuration of the environment to the point that you can create virtual machines. These tasks will be described in the following chapters.

To complete installation of the oVirt Engine successfully you must be able to determine:

1. The ports to be used for HTTP, and HTTPS, communication. The defaults are **8080** and **8443** respectively.
2. The Fully Qualified Domain Name (FQDN) of the system the engine is to be installed on.
3. The password you will use to secure the oVirt administration account.
4. The password you will use to secure the database.
5. The Organization Name to use when creating the engine's security certificates.
6. The storage type to be used for the initial data center attached to the engine. The default is NFS.
7. The path to use for the ISO share, if the engine is being configured to provide one. The display name, which will be used to label the domain in the oVirt Engine also needs to be provided.
8. The firewall rules, if any, present on the system that need to be integrated with the rules required for the engine to function.

Before installation is completed the values selected are displayed for confirmation. Once the values have been confirmed they are applied and the oVirt Engine is ready for use.

Example 3.1. Completed Installation

```
oVirt Manager will be installed using the following configuration:
=====
http-port:                8080
https-port:               8443
host-fqdn:                engine.demo.ovirt.org
auth-pass:                *****
db-pass:                  *****
org-name:                 oVirt
default-dc-type:          NFS
nfs-mp:                   /isodomain
iso-domain-name:          ISODomain
override-iptables:       yes
Proceed with the configuration listed above? (yes|no):
```



Note — Automated Installation

Automated installations are created by providing **engine-setup** with an answer file. An answer file contains answers to the questions asked by the setup command.

- To create an answer file, use the `--gen-answer-file` parameter to set the location to which the answer file must be saved. The **engine-setup** command will record your answers to the file.

```
# engine-setup --gen-answer-file=ANSWER_FILE
```

- To use an answer file for a new installation, use the `--answer-file` parameter to set the location of the answer file that must be used. The command **engine-setup** command will use the answers stored in the file to complete installation.

```
# engine-setup --answer-file=ANSWER_FILE
```

Run **engine-setup --help** for further information.

3.1. Installation

In this section installation of the oVirt Engine packages, and their dependencies, will be performed using **yum**. The packages provided via this mechanism are expected to work for users of Fedora, Red Hat Enterprise Linux, and other Enterprise Linux derivatives. Users of other distributions may wish to consult the following resources:

Debian

http://www.ovirt.org/wiki/Ovirt_build_on_debian/ubuntu

Gentoo

<http://wiki.gentoo.org/wiki/OVirt>

Ubuntu

http://www.ovirt.org/wiki/Ovirt_build_on_debian/ubuntu

Note that once the Engine itself is installed, it is to be configured as documented in this guide regardless of distribution.

1. Use **wget** to retrieve the oVirt **yum** repository configuration.

```
# wget http://ovirt.org/releases/stable/ovirt-engine.repo -O /etc/yum.repos.d/ovirt-engine.repo
```

2. If installed, the `classpathx-jaf` package must be removed. It conflicts with some of the components installed to support JBoss.

```
# yum remove classpathx-jaf
```

3. Use **yum** to ensure that the most up to date versions of all installed packages are in use.

```
# yum upgrade
```

4. Use **yum** to initiate installation of the *ovirt-engine* package and all dependencies. You must run this command as the root user.

```
# yum install ovirt-engine
```

Result:

All required packages and dependencies are installed. You must now configure the system.

3.2. Configuration

Once package installation is complete the oVirt Engine must be configured. The **engine-setup** command is provided to assist with this task. The script asks you a series of questions, the answers to which form the basis for system configuration. Once all required values have been provided the updated configuration is applied and the oVirt Engine services are started.

1. Start Setup Script

To begin configuring the system run **engine-setup** as the root user.

```
# engine-setup
```

2. Set Port for HTTP

The script prompts for entry of the port to use for HTTP communication. To use the default value, **8080**, press **Enter**. To use an alternative value enter it in the field, and then press **Enter**.

```
HTTP Port [8080] :
```

The port you select also appears in the URL that must be used to access the oVirt Engine over HTTP.

Example 3.2. Access Using HTTP

For a machine with hostname **engine.demo.ovirt.org** using the default HTTP port value, **8080**, the URL to access the entry page over HTTP is **http://engine.demo.ovirt.org:8080/**.

3. Set Port for HTTPS

The script prompts for entry of the port to use for HTTPS communication. To use the default value, **8443**, press **Enter**. To use an alternative value enter it in the field, and then press **Enter**.

```
HTTPS Port [8443] :
```

Where a port other than **8443** is selected it changes the URL that must be used to access the oVirt Engine over HTTPS.

Example 3.3. Access Using HTTPS

For a machine with hostname **engine.demo.ovirt.org** using the default HTTPS port value, **8443**, the URL to access the entry page over HTTPS is **https://engine.demo.ovirt.org:8443/**.

4. Set Fully Qualified Domain Name (FQDN)

The script prompts for entry of the system's fully qualified domain name. This name should be resolvable via both forward and reverse DNS lookup. The script attempts to determine the fully qualified domain name automatically. The value identified is listed in square brackets as the default setting for the fully qualified domain name, based on your setup.

```
Host fully qualified domain name, note that this name should be fully resolvable  
[engine.demo.ovirt.org] :
```

Where the automatically determined fully qualified domain name is correct, press **Enter** to use the value and continue. Where the automatically determined fully qualified domain name is not correct, enter the correct value and press **Enter** to use the new value and continue.

5. Set Administrator Password

The script creates an authentication domain internal to the oVirt Engine for the default administrative account. The domain is named `internal`, the administrative user is called `admin`. External authentication domains are added as a post-installation step using the **`ovirt-manage-domains`** command.

You must choose a password for the `admin` user. You will be asked to enter it a second time to confirm your selection.

```
Password for Administrator (admin@internal) :
```

6. Set Database Password

The script prompts for entry of a password to use for the oVirt Engine database. You must enter a strong password. Strong passwords consist of a mix of uppercase, lowercase, numeric, and punctuation characters. They are six or more characters long and do not contain dictionary words. Enter the desired password and press **Enter**. You will be asked to enter the password again to confirm it.

```
Database password (required for secure authentication with the locally created  
database) :
```

7. Set Organization Name

The script prompts for entry of the Organization Name. The Organization Name appears in the **Subject** field of the certificate used to secure communications with the oVirt Engine.

```
Organization Name for the Certificate :
```

8. Configure Default Storage Type

The script prompts for selection of the default storage type. This is the storage type that is used for the **Default** data center. You are able to add further data centers that use different storage types from the Administration Portal at any time.

```
The default storage type you will be using ['NFS' | 'FC' | 'ISCSI'] [NFS] :
```

The default selection is Network File System (**NFS**). The other available values are:

- Fibre Channel (**FC**),
- Internet Small Computer System Interface (**ISCSI**), and

The **NFS**, **FC**, and **ISCSI** options are used to connect to remote storage. oVirt also supports The **LOCALFS** storage type which allows the use of local storage attached to the virtualization hosts, but this storage type is not supported for the **Default** data center.

To use the default selection, **NFS**, press **Enter**. To select **FC**, or **ISCSI** then enter the value and then press **Enter**.

9. Configure NFS ISO Domain

The script asks whether or not an NFS share should be configured on the server and used as an ISO storage domain.

```
Should the installer configure NFS share on this server to be used as an ISO Domain?
['yes' | 'no'] [yes] :
```

An ISO storage domain is used to store copies of removable media for use when provisioning and using virtual machines. The oVirt Engine is able to use either an ISO storage domain on the system it is installed to or one that exists on remote storage. In either case the ISO storage domain must be accessible via NFS. The ability to configure an ISO storage domain is also available from the Administration Portal after installation.

To take the default action, which is to configure an NFS share for use as an ISO storage domain, press **Enter**. To skip this step type **no** and press **Enter**.

If you chose to configure an NFS share then you will also need to provide both a path and a display name for it. The path is the location on the local file system where the NFS share must be created. The directory must not already exist.

```
Mount point path:
```

The display name is the name the storage domain will appear under in the oVirt Engine. The display name must not contain non-alphanumeric characters other than the underscore (`_`) and the hyphen (`-`).

```
Display name for the ISO domain:
```

The ISO domain will be created and exported as an NFS share. It will be shown as active in the oVirt Engine once the first active virtualization host has been added to the **Default** data center.

10. Configure Firewall

The oVirt Engine requires that network traffic on a number of ports be allowed through the system's firewall. The **engine-setup** script is able to configure this automatically, but selecting this option overrides any existing firewall configuration. Where there is an existing firewall configuration that needs to be maintained you must manually configure the firewall to include the additional rules required by the oVirt Engine.

```
Firewall ports need to be opened.
You can let the installer configure iptables automatically overriding the current
configuration. The old configuration will be backed up.
Alternately you can configure the firewall later using an example iptables file found
under /usr/share/ovirt/conf/iptables.example
Should the installer configure iptables now? ['yes' | 'no'] [yes] :
```

To proceed with automatic firewall configuration type **yes** and then press **Enter**.

To skip automatic firewall configuration type **no** and then press **Enter**. You will need to add rules equivalent to those found in `/usr/share/ovirt/conf/iptables.example` to your **iptables** configuration.

11. Confirm Configuration

You have now provided the script with all the information required to complete configuration of the oVirt Engine. The values which you entered are displayed for confirmation.

Example 3.4. Configuration Confirmation Screen

```
oVirt Manager will be installed using the following configuration:
=====
http-port:                8080
https-port:              8443
host-fqdn:               engine.demo.ovirt.org
auth-pass:               *****
db-pass:                 *****
org-name:                oVirt
default-dc-type:        NFS
nfs-mp:                  /isoshare
iso-domain-name:        ISODomain
override-iptables:      yes
Proceed with the configuration listed above? (yes|no):
```

To permanently apply the configuration values listed type **yes** and then press **Enter** to apply the configuration.

If one or more of the configuration values listed is incorrect type **no** and then **Enter** to revisit the configuration.

The configuration values are applied. A number of services need to be started and as a result this step takes some time. Do not terminate the installation once application of the configuration values has commenced.

Once the script has completed successfully take note of the additional information it provides. In particular note down the **SSH Certificate fingerprint**, **SSH Public key fingerprint**, and oVirt Engine URL for your records.

Example 3.5. Successful Configuration

```
Installing:
Creating JBoss Profile...          [ DONE ]
Creating CA...                    [ DONE ]
Setting Database Security...      [ DONE ]
Creating Database...              [ DONE ]
Updating the Default Data Center Storage Type... [ DONE ]
Editing JBoss Configuration...    [ DONE ]
Editing oVirt Manager Configuration... [ DONE ]
Configuring the Default ISO Domain... [ DONE ]
Configuring Firewall (iptables)... [ DONE ]
Starting JBoss Service...         [ DONE ]

**** Installation completed successfully ****

(Please allow oVirt Manager a few moments to start up....)
```

```
Additional information:
* SSL Certificate fingerprint:
4C:A4:8F:93:62:50:C1:63:C8:09:70:77:07:90:FD:65:5B:3C:E8:DD
* SSH Public key fingerprint: fa:71:38:88:58:67:ae:f0:b1:17:fe:91:31:6c:66:6e
* A default ISO share has been created on this host.
  If IP based access restrictions are required, please edit /isoshare entry in /etc/
exports
* The firewall has been updated, the old iptables configuration file was saved to /
usr/share/ovirt/conf/iptables.backup.103654-09092011_866
* The installation log file is available at: /var/log/ovirt/engine-
setup_2011_09_09_10_32_56.log
* Please use the user "admin" and password specified in order to login into oVirt
Manager
* To configure additional users, first configure authentication domains using the
'ovirt-manage-domains' utility
* To access oVirt Manager please go to the following URL: http://
engine.demo.ovirt.org:8080
```

Result:

The oVirt Engine has been installed and configured successfully. You are now able to connect to the Administration Portal for the first time, see [Section 3.3, "Connect to the Administration Portal"](#) for further information.

3.3. Connect to the Administration Portal

The Administration Portal allows you to create, configure, monitor, and maintain the oVirt environment using a graphical interface. To begin configuring your oVirt environment you must first log into the Administration Portal.

Procedure 3.1. Connect to oVirt web management portal

1. Open a web browser.
2. Browse to **`http://engine.demo.ovirt.org:8080/`**, replacing *engine.demo.ovirt.org* with the hostname of the machine you installed oVirt Engine on and *8080* with the HTTP port specified during installation.
3. Click the **Administration Portal** link.
4. Log in with username `admin`, domain `internal`, and the password that you provided when running **engine-setup**.

Result:

You have now successfully logged into the oVirt web Administration Portal. You can now begin installing virtualization hosts - see [Part III, "Installing Virtualization Hosts"](#).

Part III. Installing Virtualization Hosts

Introduction to Virtualization Hosts

oVirt supports both virtualization hosts which run the oVirt Node, and those which run a pre-configured Linux installation. Both types of virtualization host are able to coexist in the same oVirt environment.

Prior to installing virtualization hosts you should ensure that:

- all virtualization hosts meet the hardware requirements outlined in [Section 2.1.2, “Virtualization Host Requirements”](#), and
- you have successfully completed installation of the oVirt Engine as outlined in [Chapter 3, Engine Installation](#) .
- To install oVirt Nodes, see [Chapter 5, Installing the oVirt Node](#).



Important — Attach at Least Two Virtualization Hosts

It is recommended that you install at least two virtualization hosts and attach them to the oVirt environment. Where you attach only one virtualization host you will be unable to access features such as migration which require redundant hosts..

Installing the oVirt Node

This chapter covers installing and integrating oVirt Nodes with a oVirt Engine.

- The oVirt Node *must* be installed on a physical server and cannot be installed on a virtual machine.
- The installation process will reconfigure the selected storage device and destroy all data. Therefore, ensure that any data to be retained is successfully backed up before proceeding.
- The following method can be used when installing multiple servers. However, ensure that unique hostnames and IP addresses are used for each node installation, in order to avoid network conflicts.
- The following procedure provides installation instructions for using a CD-ROM created using the oVirt Node ISO image available from the oVirt website.
- oVirt Nodes can use Storage Attached Networks (SANs) and other network storage for storing virtualized guest images. However, a local storage device is required for installing and booting the node.



Important — DNS Configuration

The oVirt Node must exist in the same DNS domain as the oVirt Engine.



Note — Automated Installations

oVirt Node installations can be automated or conducted without interaction. This type of installation is only recommended for advanced users.

5.1. Installation Media

This section covers creating installation media and preparing your systems before installing a oVirt Node.

This section covers installing oVirt Nodes on a local storage device. This storage device is a removable USB storage device, an internal hard disk drive or solid state drive. Once the node is installed, the system will boot the node and all configuration data is preserved on the system.

5.1.1. Preparation Instructions

The oVirt Node ISO image is available from the oVirt website at <http://ovirt.org/releases/stable/binary/>. The oVirt Node ISO image's filename is of the form **ovirt-node-image-version-release.iso** in this directory.

oVirt also provides tools to assist with provisioning Nodes. These are provided by the *ovirt-node* and *ovirt-node-tools* packages. These packages are found in the same **yum** repository used to install oVirt Engine.

BIOS Settings and Boot Process Troubleshooting

Before installing oVirt Nodes it is necessary to verify the BIOS is correctly configured for the chosen installation method. Many motherboard and PC manufacturers disable different booting methods in the BIOS. Most BIOS chips boot from the following devices in order:

1. 3.5 inch diskette
2. CD-ROM or DVD device
3. Local hard disk

Many BIOS chips have disabled one or more of the following boot methods: USB storage devices, CD-ROMs, DVDs or network boot. To boot from your chosen method, enable the method or device and set that device as the first boot device in BIOS.

Most but not all motherboards support the boot methods described in this chapter. Consult the documentation for your motherboard or system to determine whether it is possible to use a particular boot method.



Warning — BIOS Settings Vary Between Manufacturers

BIOS settings vary between manufacturers. Any specific examples of BIOS settings may be inaccurate for some systems. Due to this inconsistency, it is necessary to review the motherboard or system manufacturer's documentation.

Confirm Hardware Virtualization Support

Verify that your system is capable of running the oVirt Node. Nodes require that virtualization extensions are present and enabled in the BIOS before installation proceeds.

1. Boot the node from removable media. For example, a USB stick or CD-ROM.
2. When the message **Automatic boot in 30 seconds...** is displayed, and begins counting down from thirty, press any key to skip the automatic boot process.
3. Ensure the **Install or Upgrade** option is selected and press **Tab** to edit the boot parameters.
4. Add the **rescue** parameter to the list of boot parameters shown on the screen, then press **Enter**. This action will boot the node in rescue mode.
5. Once the node boots, verify your CPU contains the virtualization extensions with the following command:

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

Output displays if the processor has the hardware virtualization extensions.

6. Verify that the KVM modules load by default:

```
# lsmod | grep kvm
```

Result:

If the output includes `kvm_intel` or `kvm_amd` then the kvm hardware virtualization modules are loaded and the system meets the requirements. If the output does not include the required modules then you must check that your hardware supports the virtualization extensions and that they are enabled in the system's BIOS.

5.1.2. Preparing a Node USB Storage Device

The Node is able to install itself onto USB storage devices or solid state disks. However, the initial boot/install USB device must be a separate device from the installation target. Network booting with PXE and tftp provides the greatest flexibility and scalability. For environments where network restrictions prevent network booting, or for systems without PXE capable network interface cards, a local media installation such as CD-ROM or USB is necessary. Booting from USB storage devices is a useful alternative to booting from CD, for systems without CD-ROM drives.



Note — USB Boot Support

Not all systems support booting from a USB storage device. Ensure that your system's BIOS supports booting from USB storage devices before proceeding.

5.1.2.1. Making a USB Storage Device into a Node Boot Device

This section covers making USB storage devices which are able to be used to boot nodes.

5.1.2.1.1. Using `livecd-iso-to-disk` to Create USB Install Media

The `livecd-iso-to-disk` command will install a node onto a USB storage device. The `livecd-iso-to-disk` command is part of the `rhev-node` package. Devices created with this command are able to boot the nodes on systems which support booting via USB.

The basic `livecd-iso-to-disk` command usage follows this structure:

```
# livecd-iso-to-disk image device
```

Where the *device* parameter is the partition name of the USB storage device to install to. The *image* parameter is a ISO image of the node. The default node image location is `tmp/ovirt-node-image-2.2.2-2.2.fc16.iso`. The `livecd-iso-to-disk` command requires devices to be formatted with the FAT or EXT3 file system.



Note — Partitions and `livecd-iso-to-disk`

`livecd-iso-to-disk` uses a FAT or EXT3 formatted partition or block device.

USB storage devices are sometimes formatted without a partition table, use `/dev/sdb` or similar device name.

When a USB storage device is formatted with a partition table, use `/dev/sdb1` or similar device name.

1. Download the oVirt Node ISO image file from the location specified in [Section 5.1.1, “Preparation Instructions”](#).
2. Use the **livecd-iso-to-disk** command to copy the oVirt Node ISO image to the disk. The `--format` parameter formats the disk. The `--reset-mbr` initializes the Master Boot Record (MBR). The example uses a USB storage device named `/dev/sdc`.

Example 5.1. Use of **livecd-iso-to-disk**

```
# livecd-iso-to-disk --format --reset-mbr /tmp/ovirt-node-image-2.2.2-2.2.fc16.iso /
dev/sdc
Verifying image...
/tmp/ovirt-node-image-2.2.2-2.2.fc16.iso:  eccc12a0530b9f22e5ba62b848922309
Fragment sums: 8688f5473e9c176a73f7a37499358557e6c397c9ce2dafb5eca5498fb586
Fragment count: 20
Checking: 100.0%

The media check is complete, the result is: PASS.

It is OK to use this media.
Copying live image to USB stick
Updating boot config file
Installing boot loader
syslinux: only 512-byte sectors are supported
USB stick set up as live image!
```

Result:

The USB storage device (`/dev/sdc`) is ready to be used to boot a system and install the node on it.

5.1.2.1.2. Using **dd** to Create USB Install Media

The **dd** command can also be used to install a node onto a USB storage device. Media created with the command can boot the node on systems which support booting via USB. Red Hat Enterprise Linux provides **dd** as part of the *coreutils* package. Versions of **dd** are also available on a wide variety of Linux and Unix operating systems.

Windows users are able to obtain the **dd** command through installation of **oVirt Cygwin**, a free Linux-like environment for Windows. Refer to [Procedure 5.2, “Using **dd** to Create USB Install Media on Systems Running Windows”](#) for instruction on the installation and use of **oVirt Cygwin** to install the node to a USB storage device.

The basic **dd** command usage follows this structure:

```
# dd if=image of=device
```

Where the *device* parameter is the device name of the USB storage device to install to. The *image* parameter is a ISO image of the node. The default node image location is **tmp/ovirt-node-image-2.2.2-2.2.fc16.iso**. The **dd** command does not make assumptions as to the format of the device as it performs a low-level copy of the raw data in the selected image.

Procedure 5.1. Using **dd** to Create USB Install Media

1. Download the oVirt Node ISO image file from the location specified in [Section 5.1.1, “Preparation Instructions”](#).
2. Use the **dd** command to copy the oVirt Node ISO image file to the disk. The example uses a USB storage device named `/dev/sdc`.

Example 5.2. Use of `dd`

```
# dd if=tmp/ovirt-node-image-2.2.2-2.2.fc16.iso of=/dev/sdc
243712+0 records in
243712+0 records out
124780544 bytes (125 MB) copied, 56.3009 s, 2.2 MB/s
```

**Warning — All Data on the Device Specified Will be Overwritten**

The **dd** command will overwrite all data on the device specified for the *of* parameter. Any existing data on the device will be destroyed. Ensure that the correct device is specified and that it contains no valuable data before invocation of the **dd** command.

Result:

The USB storage device (`/dev/sdc`) is ready to boot a node.

Procedure 5.2. Using `dd` to Create USB Install Media on Systems Running Windows

1. Access <http://www.redhat.com/services/custom/cygwin/> and click the **oVirt Cygwin official installation utility** link. The **rhsetup.exe** executable will download.
2. As the Administrator user run the downloaded **rhsetup.exe** executable. The **oVirt Cygwin** installer will display.
3. Follow the prompts to complete a standard installation of **oVirt Cygwin**. The *Coreutils* package within the *Base* package group provides the **dd** utility. This is automatically selected for installation.
4. Copy the **rhev-node.iso** file downloaded from **Red Hat Network** to **C:\rhev-node.iso**.
5. As the Administrator user run **oVirt Cygwin** from the desktop. A terminal window will appear.

**Important — Run oVirt Cygwin as Administrator**

On the **Windows 7** and **Windows Server 2008** platforms it is necessary to right click the **oVirt Cygwin** icon and select the **Run as Administrator...** option to ensure the application runs with the correct permissions.

6. In the terminal run **cat /proc/partitions** to see the drives and partitions currently visible to the system.

Example 5.3. View of Disk Partitions Attached to System

```
Administrator@test /
```

```
$ cat /proc/partitions
major minor #blocks name
 8      0 15728640 sda
 8      1  102400 sda1
 8      2 15624192 sda2
```

7. Plug the USB storage device which is to be used as the media for the node installation into the system. Re-run the **cat /proc/partitions** command and compare the output to that of the previous run. A new entry will appear which designates the USB storage device.

Example 5.4. View of Disk Partitions Attached to System

```
Administrator@test /
$ cat /proc/partitions
major minor #blocks name
 8      0 15728640 sda
 8      1  102400 sda1
 8      2 15624192 sda2
 8     16   524288 sdb
```

8. Use the **dd** command to copy the **rhev-node.iso** file to the disk. The example uses a USB storage device named `/dev/sdb`. Replace `sdb` with the correct device name for the USB storage device to be used.

Example 5.5. Use of **dd** Command Under oVirt Cygwin

```
Administrator@test /
$ dd if=/cygdrive/c/rhev-node.iso of=/dev/sdb& pid=$!
```

The provided command starts the transfer in the background and saves the process identifier so that it can be used to monitor the progress of the transfer. Refer to the next step for the command used to check the progress of the transfer.



Warning — All Data on the Device Specified will be Overwritten

The **dd** command will overwrite all data on the device specified for the *of* parameter. Any existing data on the device will be destroyed. Ensure that the correct device is specified and that it contains no valuable data before invocation of the **dd** command.

9. Transfer of the ISO file to the USB storage device with the version of **dd** included with **oVirt Cygwin** can take significantly longer than the equivalent on other platforms.

To check the progress of the transfer in the same terminal window that the process was started in send it the **USR1** signal. This can be achieved by issuing the **kill** in the terminal window as follows:

```
kill -USR1 $pid
```

10. When the transfer operation completes the final record counts will be displayed.

Example 5.6. Result of `dd` Initiated Copy

```
210944+0 records in
210944+0 records out
108003328 bytes (108 MB) copied, 2035.82 s, 53.1 kB/s

[1]+ Done dd if=/cygdrive/c/rhev-node.iso of=/dev/sdb
```

Result:

The USB storage device (`/dev/sdb`) is ready to boot a node.

5.1.2.2. Booting a Node USB Storage Device

Booting a node from a USB storage device is similar to booting other live USB operating systems. To boot from a USB storage device:

1. Enter the system's BIOS menu to enable USB storage device booting if not already enabled.
 - a. Enable USB booting if this feature is disabled.
 - b. Set booting USB storage devices to be first boot device.
 - c. Shut down the system.
2. Insert the USB storage device that contains the node boot image.
3. Restart the system.

Result:

The Node will boot automatically.

If the node is running, you must now initialize the local storage device. Refer to [Section 5.2.1.1, "Booting from the Installation Media"](#) for details.

5.1.3. Preparing a Node from a CD-ROM or DVD

It is possible to install the node with a CD-ROM or DVD.

5.1.3.1. Making a Node CD-ROM Boot Disk

Burn the node image to a CD-ROM with the `cdrecord` command. The `cdrecord` command is part of the `cdrecord` package which is installed on Red Hat Enterprise Linux by default.

1. Verify that the `cdrecord` package is installed on the system.

Example 5.7. Verify Installation of `cdrecord` Package

```
# rpm -q cdrecord
cdrecord-2.01-10.7.e15
```

If the package version is in the output the package is available.

If it is not listed, install `cdrecord`:

```
# yum install cdrecord
```

2. Insert a blank CD-ROM or DVD into your CD or DVD writer.
3. Record the ISO file to the disc. The `cdrecord` command uses the following:

```
cdrecord dev=device /iso/file/path/
```

This example uses the first CD-RW (`/dev/cdrw`) device available and the default node image location, **tmp/ovirt-node-image-2.2.2-2.2.fc16.iso**.

Example 5.8. Use of `cdrecord` Command

```
# cdrecord dev=/dev/cdrw tmp/ovirt-node-image-2.2.2-2.2.fc16.iso
```

Result:

If no errors occurred, the node is ready to boot. Errors sometimes occur during the recording process due to errors on the media itself. If this occurs insert another writable disk and repeat the command above.

The Node uses a program (**isomd5sum**) to verify the integrity of the installation media every time the node is booted. If media errors are reported in the boot sequence you have a bad CD-ROM. Follow the procedure above to create a new CD-ROM or DVD.

5.1.3.2. Booting a Node CD-ROM

For many systems, the default BIOS configuration boots from CD-ROM first. If booting from CD-ROM is disabled or is not the first boot device refer to [BIOS Settings and Boot Process Troubleshooting](#) and your manufacturers manuals for more information.

To boot from CD-ROM insert the node CD-ROM and then restart the computer.

The Node will start to boot. If the node does not start to boot your BIOS may not be configured to boot from CD-ROM first or booting from CD-ROM may be disabled.

If the node is running, you must now initialize the local storage device. Refer to [Section 5.2.1.1, “Booting from the Installation Media”](#) for details.

5.2. Installation

This chapter documents the installation of the oVirt node. oVirt Nodes are able to use Storage Area Networks (SANs) and other network storage for storing virtualized guest images. Nodes can be installed on SANs, provided that the Host Bus Adapter (HBA) permits configuration as a boot device in BIOS.

Nodes are able to use multipath devices for installation. Multipath is often used for SANs or other networked storage. Multipath is enabled by default at install time. Any block device which responds to **scsi_id** functions with multipath. Devices where this is not the case include USB storage and some older ATA disks.

There are two methods for installing oVirt Nodes:

- Interactive Installation (see [Section 5.2.1, “Interactive Installation”](#)).
- Automated Installation with Kernel Parameters (see the *Red Hat Enterprise Linux — Node Deployment Guide*).

5.2.1. Interactive Installation

oVirt Nodes must be installed on physical servers, not virtual machines.

The instructions in this section cover installation on a single system. When deploying on multiple systems always remember to use unique hostnames and IP addresses to avoid networking conflicts.

5.2.1.1. Booting from the Installation Media

There are several methods for booting nodes, refer to [Section 5.1, “Installation Media”](#) for detailed instructions on preparing boot media for oVirt Node installation.

Procedure 5.3. Booting from the Installation Media

1. Insert the oVirt Node installation media.
2. Power on the system and ensure the system boots from the installation media.
3. The boot splash screen appears. If no input is provided, the node installation will commence in 30 seconds, using default kernel parameters.
4. To modify the boot options, press any key. The boot menu will display.

The following boot options are available:

- **Boot**
Boot the node installer.
- **Boot with Serial Console**
Boot the node installer, with the console redirected to a serial device attached to `/dev/ttyS0`.
- **Boot from Local Drive**
Boot the operating system installed on the first local drive.

Select the appropriate boot option from the boot menu.

5. Where required additional kernel parameters should be appended to the default parameters displayed. A press of the **Enter** key boots the node installation with the default kernel parameters. Alternatively press the **Tab** key to edit kernel parameters for the selected boot option.

Result:

The Node boots with the provided boot options.



Important — Kernel Parameters

In edit mode you are able to add or remove kernel parameters from the list. Kernel parameters must be separated from each other by a space. Once the desired kernel parameters have been set press **Enter** to boot the system. Alternatively pressing **Esc** reverts any changes that you have made to the kernel parameters.

For more information on the kernel parameters, see the *Red Hat Enterprise Linux — Node Deployment Guide*.



Note — Upgrading Existing Nodes

To upgrade an existing node installation, the kernel must be booted with the *upgrade* parameter. This will automatically upgrade and reboot the system, rather than displaying the interactive configuration menu. For more information, see the *Red Hat Enterprise Linux — Node Deployment Guide*.

5.2.1.2. Installation Procedure

When the node is first booted the interactive installation script starts. This script facilitates installation of the oVirt Node using graphical prompts. The following keys are used to manipulate the screens which support node installation.

Menu Actions

- The directional keys (**Up**, **Down**, **Left**, **Right**) are used to select different controls on the screen. Alternatively the **Tab** key cycles through the controls on the screen which are enabled.
- Text fields are represented by a series of underscores (_). To enter data in a text field select it and begin entering data.
- Buttons are represented by labels which are enclosed within a pair of angle brackets (< and >). To activate a button ensure it is selected and press **Enter** or **Space**.
- Boolean options are represented by an asterisk (*) or a space character enclosed within a pair of square brackets ([and]). When the value contained within the brackets is an asterisk then the option is set, otherwise it is not. To toggle a Boolean option on or off press **Space** while it is selected.

Procedure 5.4. Node Installation

1. To commence Node installation select **Install Node** and press **Enter**.

2. Disk Configuration

The installation script automatically detects all disks attached to the system. This information is used to assist with selection of the boot and installation disks that the node should use. Each entry displayed on these screens indicates the **Location**, **Device Name**, and **Size (GB)** of the relevant disk.

a. **Boot disk**

The first disk selection screen is used to select the disk from which the node will boot. The Node's boot loader will be installed to the Master Boot Record (MBR) of the disk that is selected on this screen. The Node attempts to automatically detect the disks attached to the system and presents the list from which you choose the boot device. Alternatively you are able to manually select a device, by specifying a block device name, by enabling the **Other Device** option.



Important — Boot Order

The disk selected must be identified as a boot device and appear in the boot order either in the system's BIOS or in a pre-existing boot loader.

Automatically Detected Device Selection

- i. Select the entry for the disk the node is to boot from in the list.
- ii. Select the **<Continue>** button and press **Enter**. This action will save the boot device selection and start the next step of installation.

Manual Device Selection

- i. Select the **Other Device** entry from the list.
- ii. Select the **<Continue>** button and press **Enter**.
- iii. When prompted to **Please enter the disk to use for booting oVirt Node** enter the name of the block device from which the node should boot.

Example 5.9. Other Device Selection

```
Please enter the disk to use for booting oVirt Node
/dev/sda
```

- iv. Select the **<Continue>** button and press **Enter**. This action will save the boot device selection and start the next step of installation.

Once a disk has been selected it is necessary to select the **<Continue>** button and press **Enter** to save the selection and continue with node installation.

b. **Installation Disk(s)**

The disk(s) selected for installation will be those to which the node itself is installed. The Node attempts to automatically detect the disks attached to the system and presents the list from which installation devices are chosen.



Warning — Data Loss

All data on the selected storage device(s) will be destroyed.

- i. Select each disk which the node is to use for installation and press **Space** to toggle it to enabled. Repeat this step for all disks you want the node to use. Where other devices are to be used for installation, either solely or in addition to those which are listed automatically, enable the **Other Device** option.
- ii. Select the **<Continue>** button and press **Enter** to continue.
- iii. Where the **Other Device** option was specified a further prompt will appear. Enter the name of each additional block device to use for node installation separated by a comma. Once all required disks have been selected then select the **<Continue>** button and press **Enter**.

Example 5.10. Other Device Selection

```
Please select the disk(s) to use for installation of oVirt Node
Enter multiple entries separated by commas
/dev/mmcb1k0,/dev/mmcb1k1_____
```

Once the installation disk, or disks, have been selected the next stage of the installation starts.

3. Password

The Node requires that a password be set to protect local console access by the `admin` user. The installation script prompts you to enter the desired password in both the **Password** and **Confirm Password** fields.

A strong password must be used. Strong passwords consist of a mix of uppercase, lowercase, numeric, and punctuation characters. They are six or more characters long and do not contain dictionary words.

Once a strong password has been entered select **<Install>** and press **Enter** to install the node to disk.

Result:

Once installation is complete the message **oVirt Node Installation Finished Successfully** will be displayed. Select the **<Restart>** button and press **Enter** to reboot the system.

Further post installation configuration is required to connect the node to the oVirt Engine. See [Section 5.3, "Configuration"](#) for further details.



Note — Remove Boot Media

The boot media should be removed and the boot device order changed to prevent the installation sequence restarting after the system reboots.

5.3. Configuration

5.3.1. Logging In

The Node allows local console logins to facilitate post-installation configuration. The login prompt used is displayed once the node has booted:


```
Please login as 'admin' to configure the node
localhost login:
```

Type `admin` at the prompt and press **Enter**. When prompted enter the password which was set during the installation process and press **Enter** again to log in.

The Node configuration menu will then be displayed. The menu facilitates interactive configuration of the node. Throughout the remainder of this chapter it will be referred to as the main menu. The main menu provides access to multiple screens which report on the status and configuration of the node. They also provide the ability to change the node configuration.

The configuration interface is similar to that of the installation script. The same keys are used to navigate the menu and associated screens. Refer to [Menu Actions](#) to review the list of possible actions.

5.3.2. Status

The status screen displays a brief overview of the current state of the node. The information displayed consists of:

- the hostname,
- the current status of the network connection,
- the destination(s) of logs and reports, and
- the number of active virtual machines.

The status screen also provides a number of buttons to change the state of the node. They are:

- **<Lock>**: Locks the node. The username and password must be entered to unlock the node.
- **<Restart>**: Restarts the node.
- **<Power Off>**: Turns the node off.

5.3.3. Network

The **Network** screen is used to configure:

- the node's hostname,
- the DNS server(s) to use,
- the NTP server(s) to use, and
- the network interface to use.

Procedure 5.5. Hostname Configuration

1. To set or change the hostname select the **Hostname** field and enter the new hostname.
2. Select **<Apply>**, and press **Enter** to save changes to the hostname.

Result:

The hostname is updated.

Procedure 5.6. DNS Configuration

The Node supports the specification of one or more Domain Name System (DNS) servers to use when resolving host and domain names.

1. To set or change the primary DNS server select the **DNS Server 1** field and enter the IP address of the new primary DNS server to use.
2. To set or change the secondary DNS server select the **DNS Server 2** field and enter the IP address of the new secondary DNS server to use.
3. Select **<Apply>**, and press **Enter** to save changes to the DNS configuration.

Result:

The primary and secondary DNS servers queried by the node are updated.

Procedure 5.7. NTP Configuration

The Node supports the specification of one or more Network Time Protocol (NTP) servers with which the node should synchronize the system clock. It is important that the node is synchronized with the same time source as the oVirt Engine. This ensures accurate time keeping across the oVirt environment.

1. To set or change the primary NTP server select the **NTP Server 1** field and enter the IP address or hostname of the new primary NTP server to use.
2. To set or change the secondary NTP server select the **NTP Server 2** field and enter the IP address or hostname of the new secondary NTP server to use.
3. Select **<Apply>**, and press **Enter** to save changes to the NTP configuration.

Result:

The primary and secondary NTP servers queried by the node are updated.

Procedure 5.8. Network Interface Configuration

For each network interface detected the node will display the:

- **Device**,
- **Status**,
- **Model**, and
- **MAC Address**.

At least one network interface must be configured before the node is able to connect with the oVirt Engine.

1. Device Identification

Select the network interface to be configured from the list and press **Enter**.

In some cases it may be unclear which physical device an entry in the list refers to. Where this is the case the node is able to blink the physical device's network traffic lights to assist with identification. To make use of this facility select the entry from the list and, then select the **<Flash Lights to Identify>** button. Press **Enter** and, take note of which physical device's lights start blinking. The configuration screen for the selected device will be displayed.

2. IPv4 Settings

The Node supports both dynamic (DHCP), and static IPv4 network configuration.

Dynamic (DHCP) Network Configuration

Dynamic network configuration allows the node to be dynamically assigned an IP address via **DHCP**. To enable dynamic IPv4 network configuration select the **DHCP** option under **IPv4 Settings** and press **Space** to toggle it to enabled.

Static Network Configuration

Static network configuration allows the node to be manually assigned an IP address. To enable static IPv4 network configuration select the **Static** option under **IPv4 Settings** and press **Space** to toggle it to enabled.

Selection of the **Static** option enables the **IP Address**, **Netmask**, and **Gateway** fields. The **IP Address**, **Netmask**, and **Gateway** fields must be populated to complete static network configuration.

In particular it is necessary that:

- the **IP Address** is not already in use on the network,
- the **Netmask** matches that used by other machines on the network, and
- the **Gateway** matches that used by other machines on the network.

Where it is not clear what value should be used for the **IP Address**, **Netmask**, or **Gateway** field consult the network's administrator or consider a dynamic configuration.

Example 5.11. Static IPv4 Networking Configuration

```
IPv4 Settings
[ ] Disabled [ ] DHCP [*] Static
IP Address: 192.168.122.100_ Netmask: 255.255.255.0__
Gateway      192.168.1.1_____
```

3. IPv6 Settings

The oVirt Engine does not currently support IPv6 networking. IPv6 networking must remain set to **Disabled**.

4. VLAN Configuration

If VLAN support is required then populate the **VLAN ID** field with the VLAN identifier for the selected device.

5. Save Network Configuration

Once all networking options for the selected device have been set the configuration must be saved.

- Select the **<Apply>** button and press **Enter** to save the network configuration.
- The **Confirm Network Settings** dialog box will appear. Ensure that the **Ok** button is selected and press **Enter** to confirm.

Result:

The **Network** screen is displayed. The device is listed as **Configured**.

5.3.4. Security

The **Security** screen is used to change the `admin` password for both local and remote access. SSH password authentication is also enabled or disabled via this screen.

Procedure 5.9. Change Security Configuration

1. Enable SSH Password Authentication

To enable SSH password authentication for remote access select the **Enable ssh password authentication** option and press **Space** to toggle it to enabled.

2. Change admin Password

- a. Enter the desired `admin` password in the **Password** field. You should use a strong password.

Strong passwords contain a mix of uppercase, lowercase, numeric and punctuation characters. They are six or more characters long and do not contain dictionary words.

- b. Enter the desired `admin` password in the **Confirm Password** field. Ensure that the value entered in the **Confirm Password** field matches the value entered in the **Password** field exactly. Where this is not the case an error message will be displayed to indicate that the two values are different.

3. Select **<Apply>** and press **Enter** to save the security configuration.

Result:

The security configuration has been updated.

5.3.5. Logging

The Node creates and updates a number of log files. The **Logging** screen allows configuration of a daemon to automatically export these log files to a remote server.

Procedure 5.10. Change Logging Configuration

1. Logrotate Configuration

The **logrotate** utility simplifies the administration of log files. The Node uses **logrotate** to rotate logs when they reach a certain file size.

Log rotation involves renaming the current log(s) and starting new ones in their place. The **Logrotate Max Log Size** value set on the **Logging** screen is used to determine when a log should be rotated.

Enter the **Logrotate Max Log Size** in kilobytes. The default maximum log size is 1024 kilobytes.

2. Rsyslog Configuration

The **rsyslog** utility is a multithreaded syslog daemon. The Node is able to use **rsyslog** to transmit log files over the network to a remote syslog daemon. For information on setting up the remote syslog daemon consult the *Red Hat Enterprise Linux — Deployment Guide*.

- a. Enter the remote **Rsyslog** server address in the **Server Address** field.
- b. Enter the remote **Rsyslog** server port in the **Server Port** field. The default port is **514**.

3. netconsole Configuration

The **netconsole** module allows kernel messages to be sent to a remote machine. The Node uses **netconsole** to transmit kernel messages over the network.

- a. Enter the **Server Address**.
 - b. Enter the **Server Port**. The default port is **6666**.
4. **Save Configuration**
To save the logging configuration select **<Apply>** and press **Enter**.

Result:

The logging configuration has been updated and logs will be exported to the remote **Rsyslog** server specified.

5.3.6. Kernel Dump

oVirt node hosts generate a kernel dump (a **kdump** file) in the event of a system failure. These **kdump** files are essential for debugging and support.

The Node supports the export of kernel dumps by **kdump** using NFS or SSH so that they may be analyzed at a later date. The **Kernel Dump** screen provides for configuration of this facility.

Procedure 5.11. **kdump** Configuration

1. **Kernel Configuration**

Crash dumps generated by **kdump** are exported over NFS or SSH. Select the desired transfer method and press **Space** to enable it.

For the export method chosen a location to which the **kdump** files should be exported must also be specified.

a. **NFS location**

Set the NFS location to which crash logs should be exported in the **NFS Location** field. The **NFS Location** should be the full NFS path which includes fully qualified domain name and directory path.

Example 5.12. NFS Location

```
demo.ovirt.org:/var/crash
```

b. **SSH location**

Set the SSH location to which crash logs should be exported in the **SSH Location** field. The **SSH Location** should be the full SSH login which includes the fully qualified domain name and username.

Example 5.13. SSH Location

```
root@demo.ovirt.org
```

2. **Save Configuration**

To save the configuration the user must select **<Apply>** and press **Enter**.

Result:

The **Kernel Dump** configuration has been updated and kernel dumps will be exported to the remote server(s) specified.

5.3.7. Remote Storage

The Node supports the use of a remote iSCSI initiator for storage. The iSCSI initiator to use is set from the **Remote Storage** screen.

Procedure 5.12. Remote Storage Configuration

1. **iSCSI Initiator Name**

Enter the initiator name in the **iSCSI Initiator Name** field.

Example 5.14. iSCSI Initiator Name

```
iqn.1994-05.com.redhat:5189835eeb40
```

2. **Save Configuration**

To save the configuration the user must select **<Apply>** and press **Enter**.

Result:

The **Remote Storage** configuration has been updated.

5.3.8. oVirt-M

The Node is able to attach itself to the oVirt Engine immediately if the address of the engine is available. Where the engine has not yet been installed you must instead set a password. This allows the node to be added from the Administration Portal once the engine has been installed. Both modes of configuration are supported from the **oVirt-M** screen.



Important — Security Considerations

Setting a password on the **oVirt-M** configuration screen sets the node's root password and enables SSH password authentication. Once the node has successfully been added to the engine it is recommended SSH password authentication is disabled.

Procedure 5.13. oVirt-M Configuration

1. **Configuration Using a Management Server Address**

- a. Enter the IP address or fully qualified domain name of the engine in the **Management Server** field.
- b. Enter the management server port in the **Management Server Port** field. The default value is **8443**. Where a different port was selected during oVirt Engine installation then it should be specified here, replacing the default value.
- c. Enable the **Verify oVirtM Certificate** option if you wish to verify that the finger print of the certificate retrieved from the management server you specified is correct. The value that the certificate finger print should be compared against is returned at the end of oVirt Engine installation.
- d. Leave the **Password** and **Confirm Password** fields blank, these fields are not required if the address of the management server is known.

Configuration Using a Password

- a. Enter a password in the **Password** field. It is recommended that you use a strong password. Strong passwords contain a mix of uppercase, lowercase, numeric and

punctuation characters. They are six or more characters long and do not contain dictionary words.

- b. Re-enter the password in the **Confirm Password** field.
 - c. Leave the **Management Server** and **Management Server Port** fields blank. As long as a password is set, allowing the node to be added to the engine later, these fields are not required.
2. **Save Configuration**
To save the configuration select **<Apply>** and press **Enter**.

Result:

The **oVirt** configuration has been updated.

5.4. Using the Node

Where the node was configured with the address of the oVirt Engine it reboots and is automatically registered with it. The oVirt Engine interface displays the node under the **Hosts** tab. To prepare the node for use, it must be approved using oVirt Engine.

Procedure 5.14. Approving a oVirt Node

1. Login to the oVirt Engine Administration Portal.
2. From the **Hosts** tab, click on the host to be approved. The host should currently be listed with the status of **Pending Approval**.
3. Click the **Approve** button. The **Edit and Approve Hosts** dialog displays. You can use the dialog to set a name for the host and configure power management, where the host has a supported power management card. For information on power management configuration, see the *Power Management* chapter of the *oVirt — Administration Guide*.
4. Click **OK**. If you have not configured power management you will be prompted to confirm that you wish to proceed without doing so, click **OK**.

Result:

The status in the **Hosts** tab changes to **Installing**, after a brief delay the host status changes to **Up**.

Where the node was configured without the address of the oVirt Engine it needs to be added manually. To add the node manually you must have both the IP address of the machine upon which it was installed and the password that was set on the **oVirt-M** screen during configuration.

Procedure 5.15. Adding a oVirt Node with a Password

1. Login to the oVirt Engine Administration Portal.
2. From the **Hosts** tab, click **New**. The **New Host** dialog displays.
3. Enter the details of the new host.
 - **Name**: a descriptive name for the host.
 - **Address**: the IP address, or resolvable hostname of the host (provided during installation).
 - **Port**: the port used for internal communication control between the hosts. A default port is displayed; change the default only if you are sure that another port can be used.

- **Host Cluster:** the cluster to which the host belongs (select from the drop-down list).
 - **Root password:** the password of the designated host; used during installation of the host.
4. Optionally, configure power management, where the host has a supported power management card. For information on power management configuration, see the *Power Management* chapter of the *oVirt — Administration Guide*.
 5. Click **OK**.

Result:

The added node displays in the list of hosts. Once the host is successfully connect its status changes to **Up**.

Installing VDSM

The oVirt Engine supports the use of Linux hosts other than the oVirt Node. To use such a host with oVirt Engine it is currently necessary to install and configure VDSM, which provides the mechanism via which the Engine will communicate with the host. This chapter currently documents the installation of VDSM using **yum**. In future revisions alternative installation methods will also be documented, in the meantime for installation instructions specific to other platforms see the community wiki at <http://www.ovirt.org/wiki/>.



Warning — DNS Configuration

All hosts must have a fully resolvable network address. Valid forward and reverse lookups for the address must be available in DNS. Virtual machine migration will not work in environments where this is not the case.



Important

Only the AMD64/Intel 64 version systems are compatible for use as oVirt Hosts.

6.1. Installing VDSM From RPMs

1. Configure Repository

Configure the host to enable installation of software from the oVirt Project repository.

```
# wget http://ovirt.org/releases/stable/ovirt-engine.repo -O /etc/yum.repos.d/ovirt-engine.repo
```

2. Install VDSM Packages

Install the VDSM packages.

```
# yum install -y vds vds-cli
```

3. Open firewall ports

oVirt platform uses a number of network ports for management and other virtualization features.

The following steps configure **iptables** to open the required ports. These steps replace any existing firewall configuration with that required for oVirt Engine. If you have existing firewall rules with which this configuration must be merged then you must manually edit the rules defined in the **iptables** configuration file, **/etc/sysconfig/iptables**.

- a. Remove any existing firewall rules.

```
# iptables --flush
```

- b. Add the ports required by oVirt Engine to the **iptables** rules.

```
# iptables --append INPUT -m state --state ESTABLISHED,RELATED -j ACCEPT
# iptables --append INPUT -p icmp -j ACCEPT
# iptables --append INPUT -i lo -j ACCEPT
# iptables --append INPUT -p tcp --dport 22 -j ACCEPT
# iptables --append INPUT -p tcp --dport 16514 -j ACCEPT
# iptables --append INPUT -p tcp --dport 54321 -j ACCEPT
# iptables --append INPUT -p tcp -m multiport --dports 5634:6166 -j ACCEPT
# iptables --append INPUT -p tcp -m multiport --dports 49152:49216 -j ACCEPT
# iptables --append INPUT -j REJECT --reject-with icmp-host-prohibited
# iptables --append FORWARD -m physdev ! --physdev-is-bridged -j REJECT --reject-with icmp-host-prohibited
```



Note

The provided **iptables** commands add firewall rules to accept network traffic on a number of ports. These include:

- port 22 for **SSH**,
- ports 5634 to 6166 for guest console connections,
- port 16514 for **libvirt** virtual machine migration traffic,
- ports 49152 to 49216 for VDSM virtual machine migration traffic, and
- port 54321 for the oVirt Engine.

- c. Save the modified rules.

```
# service iptables save
```

- d. Ensure that the **iptables** service is configured to start on boot and has been restarted, or started for the first time if it wasn't already running.

```
# chkconfig iptables on
# service iptables restart
```

4. Configure sudo access

The oVirt Engine makes use of **sudo** to perform operations as root on the host. The default configuration stored in **/etc/sudoers** contains values to allow this. If this file has been modified since Red Hat Enterprise Linux installation these values may have been removed. As root run **visudo** to ensure that the **/etc/sudoers** contains the default configuration values. Where it does not they must be added.

```
# Allow root to run any commands anywhere
root    ALL=(ALL)  ALL
```

5. Enable SSH access for root

The oVirt management daemon accesses host machines via SSH. To do this it logs in as `root` with an encrypted key for authentication. To ensure that SSH is configured and `root` is able to use it to access the system follow these additional steps.



Warning

The first time the oVirt Engine is connected to the host it will install an authentication key. In the process it will overwrite any existing keys which exist in `/root/.ssh/authorized_keys`.

- a. These steps assume that the `openssh-server` package is installed on the system. Where the package is not present use **yum** to install it.

```
# yum install openssh-server
```

- b. Use **chkconfig** to verify which run-levels SSH is enabled at.

```
# chkconfig --list sshd
sshd 0:off 1:off 2:on 3:on 4:on 5:on 6:off
```

It is expected that the SSH daemon shows as **on** for run-levels **3**, **4**, and **5**. This is the default configuration.

If the configuration on the host differs use **chkconfig** to enable it for the required run-levels. The `/etc/init.d/sshd` script can then be used to ensure the service is currently started.

```
# chkconfig --level 345 sshd on
# /etc/init.d/sshd start
```

To verify this operation as successful run **chkconfig --list sshd** again and check the output. It should now show the daemon as **on** at run-level **3**, **4**, and **5**.

- c. oVirt Engine requires that the SSH daemon configuration on the host allows remote login by the `root` user.

To check whether or not this is the case search the `/etc/ssh/sshd_config` for the value **PermitRootLogin**. This must be done while logged in as `root`.

```
# grep PermitRootLogin /etc/ssh/sshd_config
PermitRootLogin no
```

Where `PermitRootLogin` is set to **no** the value must be changed to **yes**. To do this edit the configuration file.

```
# vi /etc/ssh/sshd_config
```

Once the updated configuration file has been saved the SSH daemon must be told to reload it.

```
# /etc/init.d/sshd reload
Reloading sshd: [ OK ]
```

The root user should now be able to access the system via SSH.

Result:

The VDSM packages are installed, it is now time to add the host from the oVirt Engine, see [Section 6.2, “To Add a Host”](#).

6.2. To Add a Host

In the process of adding a host, you will need to provide the IP and password of the host. The engine then logs into the host to perform virtualization checks, install packages, create a network bridge and reboot the host. The process of adding a new host can take some time, the state of the process can be followed in the Details pane.

1. Click the **Hosts** tab. The Hosts tab displays a list of all hosts in the system.

2. Click the **New** button. The New Host dialog displays.

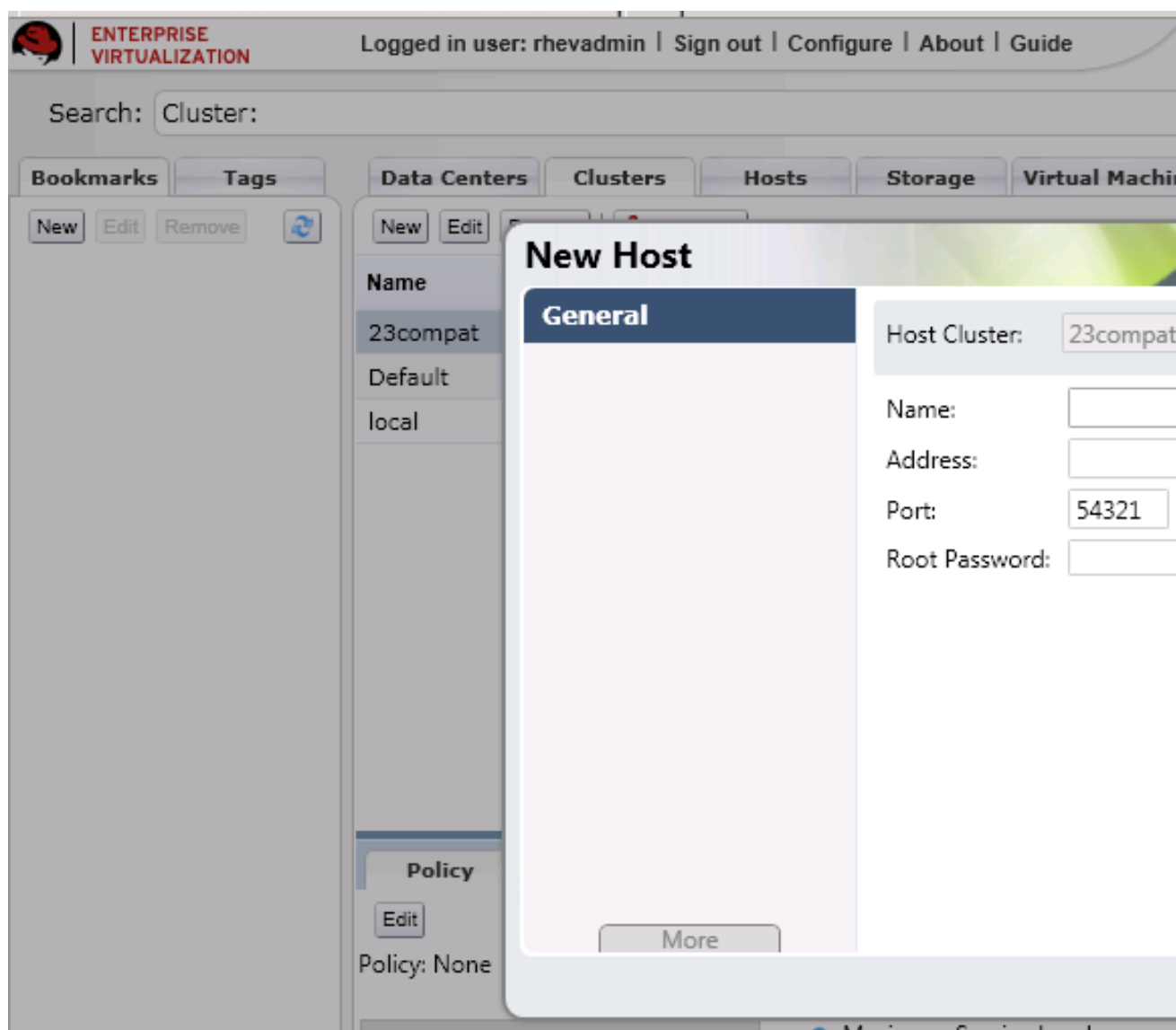


Figure 6.1. New Host Dialog

Enter the details of the new host.

- **Name:** a descriptive name for the host.
- **Address:** the IP address, or resolvable hostname of the host (provided during installation).
- **Port:** the port used for internal communication control between the hosts. A default port is displayed; change the default only if you are sure that another port can be used.
- **Host Cluster:** the cluster to which the host belongs (select from the drop-down list).
- **Root password:** the password of the designated host; used during installation of the host.
- **Enable Power Management:** Select this checkbox to turn out-of-band (OOB) power management on. If selected, the information for the following fields must also be provided.
 - The **Address** of the host. This is usually the address of the remote access card (RAC) on the host.

- A valid **User Name** for the OOB management.
- A valid, robust **Password** for the OOB management.
- The **Type** of the OOB management device. Select the appropriate device from the drop down list.

alom	Sun Integrated Lights Out Manager (ILOM)
apc	APC Master MasterSwitch network power switch
bladecenter	IBM Bladecentre Remote Supervisor Adapter
drac5	Dell Remote Access Controller for Dell computers
eps	ePowerSwitch 8M+ network power switch
ilo	HP Integrated Lights Out standard
ipmilan	Intelligent Platform Management Interface
rsa	IBM Remote Supervisor Adaptor
rsb	Fujitsu-Siemens RSB management interface
wti	WTI Network PowerSwitch

- The **Port** to connect to OOB management.
- **Slot**: The slot number in the blade chassis. This option is for blade systems only.
- **Options**: Extra command line options for the fence agent. Detailed documentation of the options available is provided in the man page for each fence agent.
- **Secure**: Some fence agents support both encrypted and unencrypted communications. Select this option to enable encrypted communications.
- Click the **Test** button to test the operation of the OOB management solution.

oVirt recommends power management. Power management enables the system to fence a troublesome host using an additional interface.



Note

If the host is required to be Highly Available, power management must be enabled and configured.

3. Click **OK**.

Result:

The new host displays in the list of hosts with a status of **Installing**. Once installation is complete, the status will update to **Reboot** and then **Awaiting**. The host must be activated for the status to change to **Up**.

**Note**

View the process of the host installation on the Details pane.

6.3. Activating a Host

After a host has been added, or an existing host has been taken down for maintenance, it needs to be activated before it can be used.

To activate a host:

1. In the Hosts tab, select the host to be activated.
2. Click the **Activate** button.

Result:

The host status changes to **Up**. Virtual machines can now run on the host.

Part IV. Environment Configuration

Planning your Data Center

Successful planning is essential for a highly available, scalable oVirt environment.

Although it is assumed that your solution architect has defined the environment before installation, the following considerations must be made when designing the system.

CPU

Virtual Machines must be distributed across hosts so that enough capacity is available to handle higher than average loads during peak processing. Average target utilization will be 50% of available CPU.

Memory

The oVirt page sharing process overcommits up to 150% of physical memory for virtual machines. Therefore, allow for an approximately 30% overcommit.

Networking

When designing the network, it is important to ensure that the volume of traffic produced by storage, remote connections and virtual machines is taken into account. As a general rule, allow approximately 50 MBps per virtual machine.

It is best practice to separate disk I/O traffic from end-user traffic, as this reduces the load on the Ethernet connection and reduces security vulnerabilities by isolating data from the visual stream. For Ethernet networks, it is suggested that bonds (802.3ad) are utilized to aggregate server traffic types.



Note

It is possible to connect both the storage and Nodes via a single high performance switch. For this configuration to be effective, the switch must be able to provide 30 GBps on the backplane.

High Availability

The system requires at least two hosts to achieve high availability. This redundancy is useful when performing maintenance or repairs.

7.1. Data Centers

Following oVirt Engine installation it is necessary to define the data centers, and clusters, that you will use to organize your virtualization hosts. The installation process creates a data center, and associated cluster, called **Default**. The storage type of the data center is set based on the one selected during installation. If you wish to make use of other storage types then you will need to add additional data centers and clusters to support them.

7.1.1. Prerequisites for Setting up a Data Center

Before you create a new data center, prepare the following resources. The following tasks must be done at the host level, not from the oVirt platform.

1. Setup and configure hosts. A host can be a oVirt Node or a Red Hat Enterprise Linux 6 host. A cluster needs a minimum of one host, and at least one active host is required to connect the system to a storage pool.
2. Setup, configure and define storage. It is recommended that data centers have a minimum of two storage domains, one to store disk images of the virtual machines and one to store ISO images. Set up the storage domains of the type required for the data center; NFS, iSCSI, FCP or Local. For example, for an NFS data center, create and mount the export directories.
3. Set up logical networks for the data center, cluster and the hosts. It is recommended that you have the IP addresses/domain names available for reference.

7.1.2. Working with Data Centers

This section describes how to configure, create and manage data centers. The data center is the highest level container for all physical and logical resources within a managed virtual environment. The data center is a collection of clusters of hosts. It owns the logical network (that is, the defined subnets for management, guest network traffic and storage network traffic) and the storage pool.

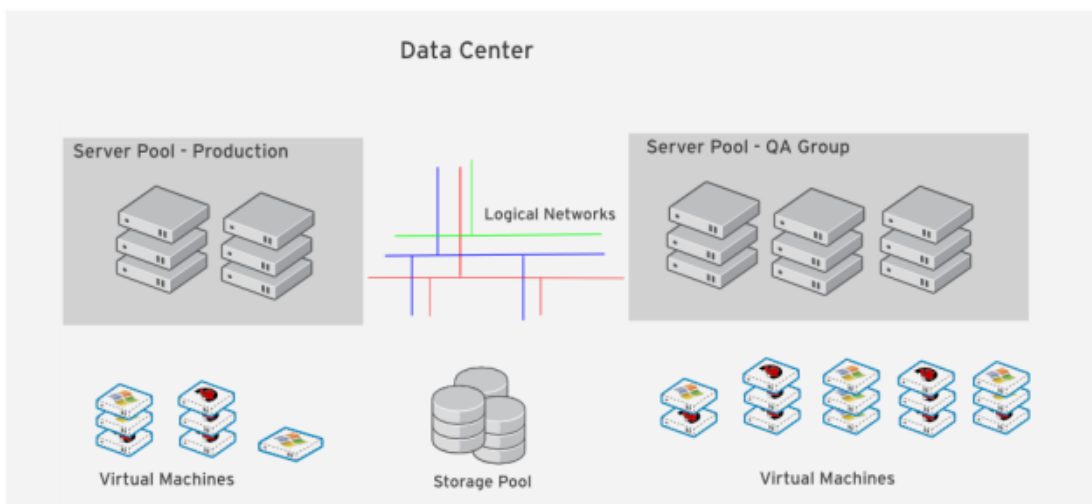


Figure 7.1. Data Centers

oVirt contains a default data center at installation. Enterprises can set up new and additional data centers that can all be managed from the single Administration Portal. For example, an organization may have different data centers for different physical locations, business units, or for reasons of security. It is recommended that you do not remove the default data center, instead set up new appropriately named data centers.

The system administrator, as the superuser, can manage all aspects of the platform, that is, data centers, storage pools, users, roles and permissions by default; however more specific administrative roles and permissions can be assigned to other users. For example, the enterprise may need a data center administrator for a specific data center, or a particular cluster may need an administrator. All system administration roles for physical resources have a hierarchical permission system. For example, a data center administrator will automatically have permission to manage all the objects in that data center, storage, cluster and hosts; while a cluster administrator can manage all objects in the particular cluster.

7.1.3. Creating a New Data Center

A data center is a logical grouping of clusters of hosts. If you wish to create an additional data center, use the instructions in this section; if you wish to use the existing default data center then you may skip this process and begin creating clusters, see [Section 7.2, “Clusters”](#).

The **Data Centers** tab displays a list of data centers.

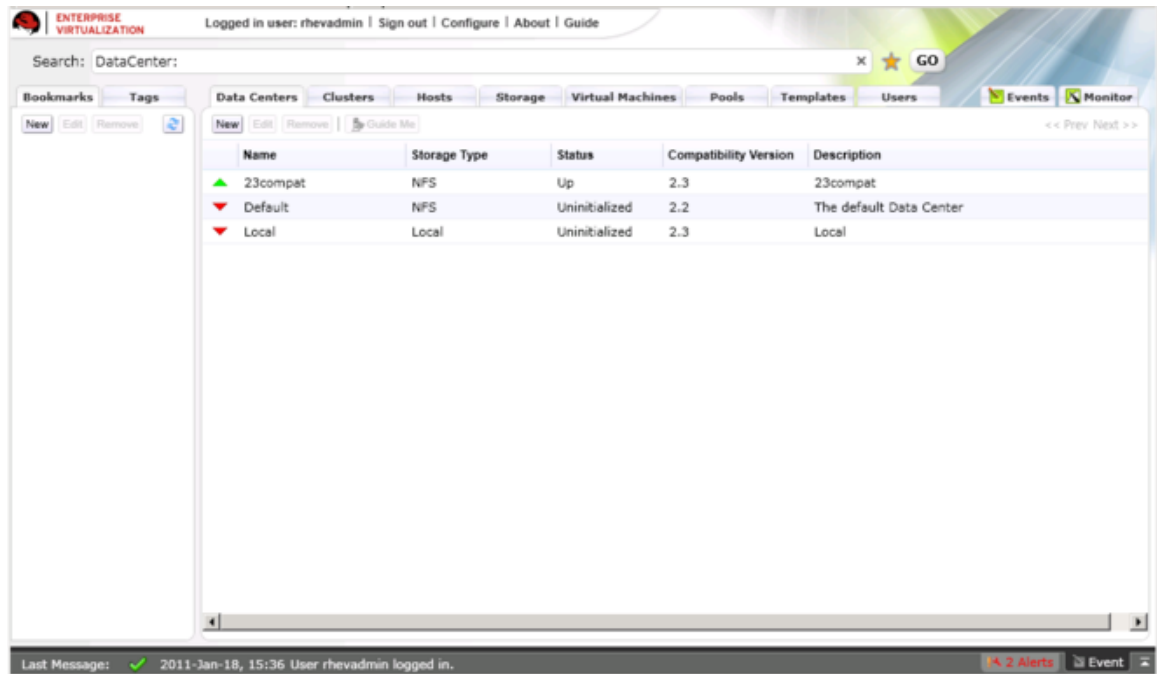


Figure 7.2. Data Centers Tab

1. Click the **New** button on the **Data Centers** tab. The **New Data Center** dialog displays.
2. Enter the **Name** and **Description** of the data center.
3. Select the storage **Type** of the data center. Select the storage appropriate to your data center; one of the following:
 - NFS
 - iSCSI
 - FCP
 - Local Storage
4. Select the **Compatibility Level** of the data center, 2.2 or 3.0.
5. Click **OK**.

6. The **Guide Me** dialog displays a list of configuration tasks that must be completed before the data center can be activated. The data center configuration tasks can be done immediately or later.



Figure 7.3. New Data Center Guide Me Dialog

Click **Configure Later** to close the dialog.

Result:

The new data center is added, and appears in appropriate searches or lists of data centers, with a status of **Uninitialized**. An uninitialized data center typically requires further configuration, for example, storage domains must be attached to it. Either click the **Configure Storage** button on the **Guide Me** dialog or select the new data center in the list, and click the **Storage** tab in the Details pane. You can define existing storage for the data center, or attach existing storage domains to the data center.

7.2. Clusters

A cluster is a collection of physical hosts that share the same storage domains and have the same type of CPU. Because virtual machines can be migrated across hosts in the same cluster, the cluster is the highest level at which power and load-sharing policies can be defined. The oVirt Enterprise Virtualization platform contains a default cluster in the default data center at installation.

Every cluster in the system must belong to a data center, and every host in the system must belong to a cluster. This enables the system to dynamically allocate a virtual machine to any host in the applicable cluster, according to policies defined on the **Cluster** tab and in the Configuration tool during runtime, thus maximizing memory and disk space, as well as virtual machine uptime.

At any given time, after a virtual machine runs on a specific host in the cluster, the virtual machine can be migrated to another host in the cluster using **Migrate**. This can be very useful when a host is shut down for maintenance. The migration to another host in the cluster is transparent to the user, and the user continues working as usual. Note that a virtual machine cannot be migrated to a host outside the cluster.

7.2.1. Creating a Cluster

Before creating a new cluster, ensure that there is at least one host available to be assigned to it. The hosts in a cluster all run the same type of CPU. That is, a cluster must not contain a mix of both Intel and AMD CPUs.



Important

The default **ovirt** network cannot be modified once a cluster has been attached to a data center. Any configuration required for the **ovirt** network, such as enabling VLAN tagging, must be performed before a cluster is attached, and the data center is still in the **Uninitialized** state.

Procedure 7.1. To create a new host cluster:

1. Click the **Clusters** tab. A list of clusters displays.
2. Click the **New** button on the **Clusters** tab.

The **New Cluster** dialog displays.

3. On the **General** tab, Select an existing **Data Center** from the list, the cluster **Name** and **Description**. The name should not include spaces.

Select the **CPU Name** for hosts in this cluster. All hosts must run the same type of CPU. The **CPU Name** list displays all the CPU types supported by oVirt. Finally on the **General** tab, select the **Compatibility Level** of the data center, from either 2.2 or 3.0.

4. Use the **Memory Optimization** tab to define how much of the host's memory can be used in excess of the permitted memory for a virtual machine in the cluster. For example, all virtual machines will not be using the full amount of allocated memory all the time. Memory sharing allows virtual machines that require additional memory at a certain time to use memory that is not being used at that time by other virtual machines. This feature allows you to fine tune how you wish to optimize the memory page sharing and large pages implementation on the hosts in the cluster.

Select from **None**, that disables memory page sharing, **Optimized for Server Load**, that sets the memory page sharing threshold to 150% of the system memory on each host, or **Optimized for Desktop Load**, that sets the memory page sharing threshold to 200% of the system memory on each host.

5. Select the **Resilience Policy** tab to define if high availability is to be implemented for the virtual machines in the cluster. If a host shuts down unexpectedly or is put into maintenance, the virtual machines running on the host can be re-run on another host in the same cluster. This field allows you to configure the migration settings for virtual machines.

Select from **Migrate Virtual Machines**, (migrates all machines); **Migrate only Highly Available Virtual Machines** or **Do Not Migrate Virtual Machines**.

6. Click **OK** to create the cluster. The new host cluster is added to the data center and displays on the Cluster tab.

The **New Cluster - Guide Me** dialog displays.

7. The **Guide Me** tab prompts you to add hosts to the new cluster. Click the **Configure Hosts** button, the **New Host** dialog displays.

Enter the details of the host to assign to the cluster. Click **OK** to close the **New Host** dialog. Now click **Configure Later** to close the **New Cluster Guide Me** dialog and return to the **Clusters** tab.

Result:

The **Hosts** tab on the Details pane displays the newly added hosts.

Network Setup

This chapter provides instruction on configuring networking for the oVirt environment. For information about managing networking, including maintenance, refer to the *oVirt Administration Guide*.

oVirt uses networking to support almost every aspect of operations. Storage, host management, user connections, and virtual machine connectivity, for example, all rely on a well planned and configured network to deliver optimal performance. Setting up networking is a vital prerequisite for a oVirt environment because it is much simpler to plan for your projected networking requirements and implement your network accordingly than it is to discover your networking requirements through use and attempt to alter your network configuration retroactively.

A familiarity with the network concepts and their use is highly recommended when planning and setting up networking in a oVirt environment. This document does not describe the concepts, protocols, requirements or general usage of bonds, bridges and logical networks. It is recommended that you read your network hardware vendor's guides for more information on managing networking.

8.1. Determine Network Requirements

It is possible to deploy a oVirt environment with no consideration given to networking at all. Simply ensuring that each physical machine in the environment has at least one *Network Interface Controller*(NIC) cabled to a switch and assigned an IP address by DHCP is enough to begin using a oVirt environment. While it is true that this approach to networking will provide a functional environment, it will not provide an optimal environment. Because network usage varies by task or action, grouping related tasks or functions into specialized networks can improve performance while simplifying the troubleshooting of network issues. By default, the oVirt Engine creates one logical network called `ovirt` and uses this logical network for all traffic.

The `ovirt` network is created and labeled as the **Management** logical network. The `ovirt` logical network is intended for management traffic between the oVirt Engine and virtualization hosts. Other types of traffic that are common to all oVirt environments are:

- Display related traffic.
- General virtual machine networking traffic.
- Storage related traffic.

For the oVirt environment to perform optimally, these types of traffic should be separated. This is easily accomplished by assigning each type of network traffic to a different logical network. Each logical network must be associated with a cabled, active network device. The network device that supports each logical network can be physical, a NIC or logical, like a bond device or virtual NIC(VNIC).

The number of logical networks that can be defined and implemented in a oVirt environment is limited by the number of network interfaces present on the virtualization hosts in the environment. Each logical network needs at least one physical device to support it. Because a logical network must be implemented for each host in a cluster for the logical network to be operational, the number of logical networks that can be implemented in a given cluster is limited by the host in a cluster with the fewest NICs.

8.2. Logical Networks

By default the management network, called **ovirt** is defined for a data center. New logical networks, for example for data, storage or display can be defined by the administrator. In general, logical

networks are created to isolate network traffic by functionality or virtualize a physical topology. In addition, other networks can be used to segregate virtual machine traffic from the management networks, or isolate traffic between groups of virtual machines in the same cluster. In oVirt Engine, network definition, type and function are encapsulated in a logical entity called a Logical Network. For example, a data center may have the following networks,

- Guest data network
- Storage network access
- Management network
- Display network (for SPICE or VNC)



Warning:

Do not change networking in a data center or a cluster if any hosts are running. This may make the host unreachable.

8.2.1. Adding Logical Networks

A logical network is assigned by an administrator as a required resource of a cluster in a data center. By extension all hosts in a cluster must have the same set of logical networks implemented. The implementation itself may vary from host to host (IP and bonding properties). Therefore, to configure a network, you need to first define the network for a data center and then apply this network to each host in a cluster. By default the management network (`ovirt`) is defined for a data center.

This section describes how to define a Storage logical network for a data center, apply the Storage logical network to a cluster, and implement the Storage logical network for hosts. This process can be repeated for each logical network being added to a data center.

To define logical networks in a cluster

1. Navigate to the **Tree** pane and click the **Expand All** button. Under System, click select the data center to which a new logical network will be added. On the results list, the selected data center displays.
2. On the details pane, select the **Logical Networks** subtab. This displays the existing logical networks. At the minimum, the default `ovirt` network is listed.
3. Click **New**. The **New Logical Network** dialog displays.

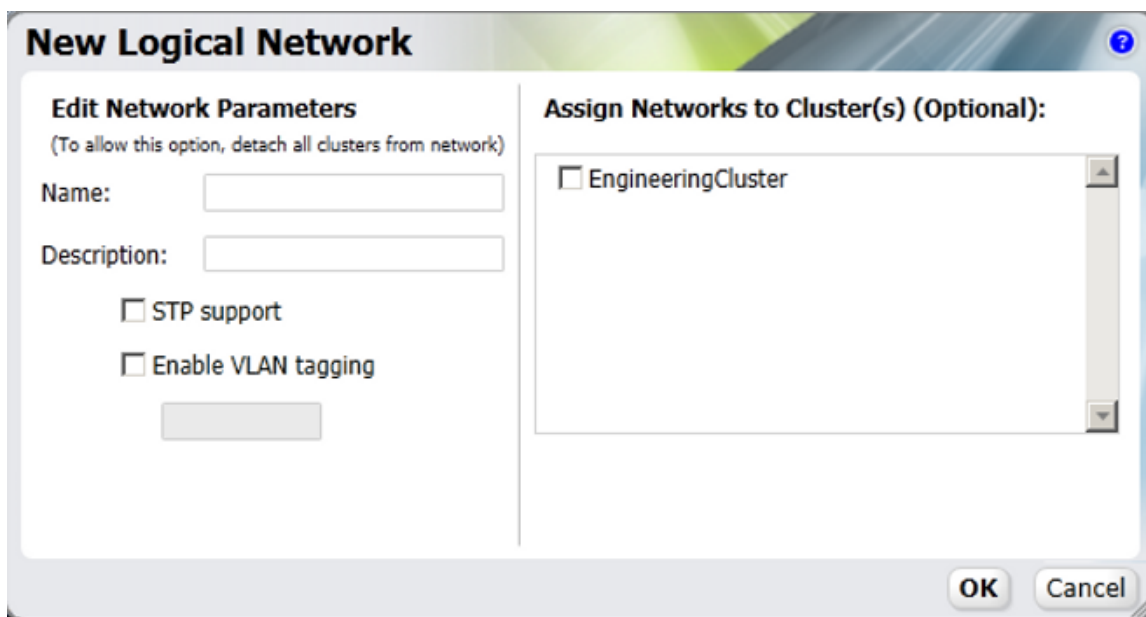


Figure 8.1. Add a logical network.

Fill in the **Name** and **Description** fields, and select the desired cluster from the **Assign Networks to Cluster(s)** section, to automatically add the Storage network to the cluster.

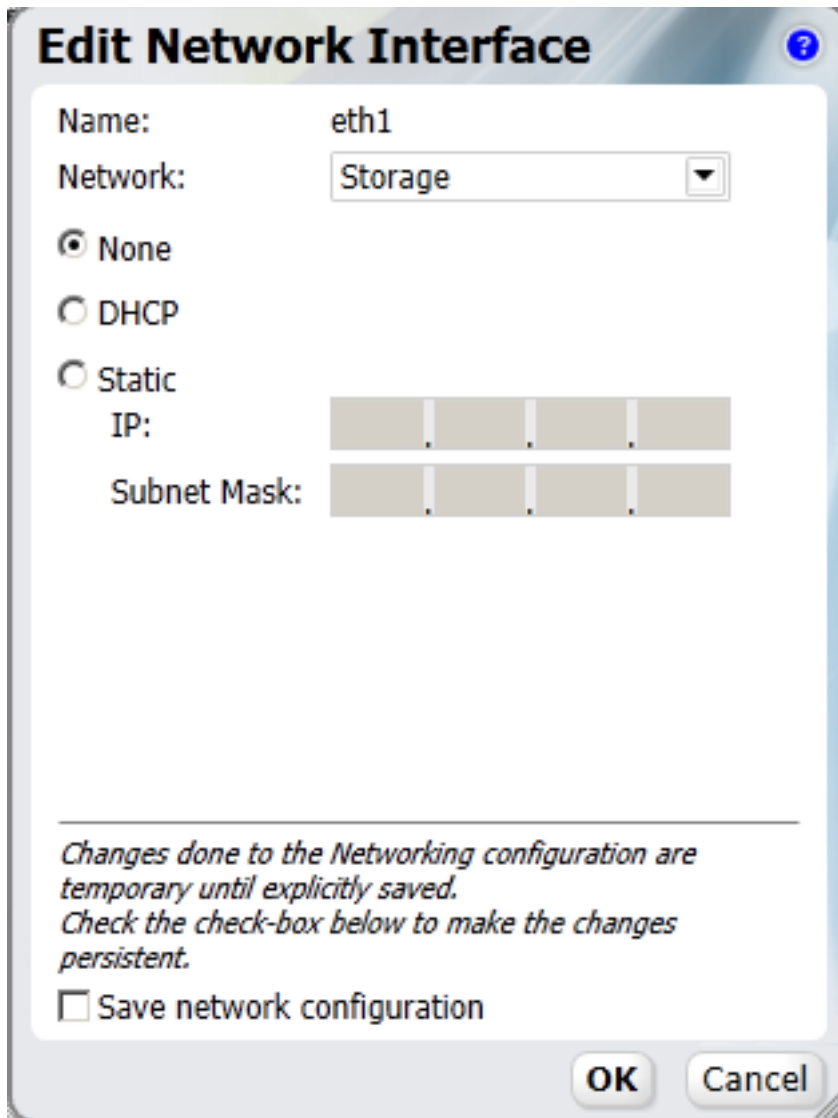
4. Click **OK** to create the new logical network.

Result:

You have defined this network as a resource required by the a cluster in the data center. You can now add this resource to the hosts in the cluster.

To add network to hosts

1. Back on the **Tree** pane, click **Default** **Clusters** **Default** **Hosts**. The **Hosts** tab displays a list of available hosts.
2. For each of your installed hosts, perform the following tasks:
 - a. Click on the host. On the **details** pane, select the **Network Interfaces** tab.
 - b. A list of network interfaces available for this host displays. One of them will already have the management network (ovirt) configured.
 - c. Select the interface on which to configure the newly added network and click the **Add/Edit** button. The **Edit Network Interface** dialog displays.



Edit Network Interface

Name: eth1

Network: Storage

None

DHCP

Static

IP: [][][][]

Subnet Mask: [][][][]

Changes done to the Networking configuration are temporary until explicitly saved. Check the check-box below to make the changes persistent.

Save network configuration

OK Cancel

Figure 8.2. Edit network interface

Configure the following options:

- Under **Network**, select your newly created storage network.
- Select the **Static** or **DHCP** radio button. If static was selected, enter the **IP** and **Subnet Mask** you have prepared as part of the prerequisites to this procedure.
- Select the **Save network configuration** checkbox.

d. Click **OK**.

Result:

You have now added a new Storage network to your data center. On the **Logical Networks** tab of the **Default** data center, you should have at least two networks - ovirt and Storage. You can repeat this process for each logical network being added to a data center

8.2.1.1. Designate Display Network

A logical network created to carry display traffic must be designated as the **Display Network** using the oVirt Engine administration portal.

To designate a new logical network as the display network

1. Select the **Clusters** tab, and select the cluster you wish to designate a display network for.
2. Select the **Logical Networks**.
3. Select the logical network that you wish to designate as the Display network.
4. Click the **Set as Display** button.
5. **Update Display Network** displays in the **Events** tab.

Result:

You have successfully designated a logical network as the display network.

8.3. Set Up a Bond Device

A *Bond* aggregates multiple NICs in a parallel manner to provide combined speed that is beyond single NIC speeds. Bonding provides increased fault tolerance by increasing the number of failures required for networking to fail completely. The NICs that form a bond device must be of the same make and model in order to ensure that both devices support the same options and modes.

The packet dispersal algorithm for a bond is determined by the bonding mode used.

Bonding Modes

oVirt supports the following common bonding modes:

- (Mode 1) Active-backup policy sets all interfaces to the backup state while one remains active. Upon failure on the active interface, a backup interface replaces it as the only active interface in the bond. The MAC address of the bond in mode 1 is visible on only one port (the network adapter), to prevent confusion for the switch. Mode 1 provides fault tolerance and is supported in oVirt.
- (Mode 2) XOR policy selects an interface to transmit packages to based on the result of a XOR operation on the source and destination MAC addresses multiplied by the modulo slave count. This calculation ensures that the same interface is selected for each destination MAC address used. Mode 2 provides fault tolerance and load balancing and is supported in oVirt.
- (Mode 4) IEEE 802.3ad policy creates aggregation groups for which included interfaces share the speed and duplex settings. Mode 4 uses all interfaces in the active aggregation group in accordance with the IEEE 802.3ad specification and is supported in oVirt.
- (Mode 5) Adaptive transmit load balancing policy ensures the outgoing traffic distribution is according to the load on each interface and that the current interface receives all incoming traffic. If the interface assigned to receive traffic fails, another interface is assigned the receiving role instead. Mode 5 is supported in oVirt.

To create a bond device using the oVirt Engine administration portal

1. Click the **Hosts** tab.
2. Select the host for which a bond device will be created.
3. Click the **Network Interfaces** tab in the details pane.
4. Select the devices that will be included in the bond, ensuring they are all of the same make and model.

- Bonding mode: the bonding mode that provides the desired functionality.
 - IP address assignment mechanism: either DHCP or Static. If static, then IP, Subnet Mask, and Default Gateway must all be set manually.
 - Check connectivity: ensure that the bond device is functional upon creation.
 - Save Network Configuration: make the bond device persistent through reboots.
6. Click the **OK**. The **Events** tab displays **Bond device created**.

Result:

A bond device is listed in the **Network Interfaces** tab of the details pane for the selected host

To enable bonding on a switch

Bonding must be enabled for the ports that the host uses on the switch it is connected to. The process by which bonding is enabled is slightly different for each switch, consult the manual provided by your switch vendor for detailed information on how to enable bonding.

The following is an bond example configuration for a switch. Your switch configuration may look different.

```
interface Port-channel11
switchport access vlan 153
switchport mode access
spanning-tree portfast disable
spanning-tree bpduguard disable
spanning-tree guard root

interface GigabitEthernet0/16
switchport access vlan 153
switchport mode access
channel-group 11 mode active

interface GigabitEthernet0/17
switchport access vlan 153
switchport mode access
```



Important

For every type of switch it is important to set up the switch bonding with the *Link Aggregation Control Protocol (LACP)* protocol and *not* the *Cisco Port Aggregation Protocol (PAgP)* protocol.

Storage Setup

This chapter provides instruction on configuring, and attaching storage to the oVirt environment. For information about managing storage, including maintenance and removal, refer to the *oVirt Administration Guide*.

oVirt uses a centralized storage system for virtual machine disk images, ISO files and snapshots. Storage networking can be implemented using Network File System (NFS), Internet Small Computer System Interface (iSCSI), Fibre Channel Protocol (FCP), or local storage attached directly to the virtualization hosts. This section describes how to set up and manage the variety of storage types that can be used in the oVirt platform. Setting up storage is a vital prerequisite for a new data center because a data center cannot be initialized unless storage domains are attached and activated.

A oVirt system administrator needs to create, configure, attach and maintain storage for the virtualized enterprise. A familiarity with the storage types and their use is highly recommended. This document does not describe the concepts, protocols, requirements or general usage of NFS, iSCSI, FCP, or local storage. It is recommended that you read your storage array vendor's guides, and refer to *Red Hat Enterprise Linux — Storage Administration Guide* for more information on managing storage, if necessary.

The oVirt platform enables administrators to assign and manage storage effectively and efficiently. The **Storage** tab on the oVirt platform provides an efficient graphical way to view and manage networked storage. The **Storage Results** list displays all the storage domains, and the Details pane enables access to general information about the domain.

oVirt platform has three types of storage domains:

- Data domains hold the disk images of all the virtual machines running in the system, operating system images and data disks. In addition, snapshots of the virtual machines are also stored in the data domain. The data cannot be shared across data centers, and the data domain must be of the same type as the data center. For example, a data center of a iSCSI type, must have an iSCSI data domain. A data domain cannot be shared between data centers.

Additionally you must attach a data domain to a data center before you will be allowed to attach domains of other types to it.

- ISO domains store ISO files (or logical CDs) used to install and boot operating systems and applications for the virtual machines. Because an ISO domain is a logical entity replacing a library of physical CDs or DVDs, an ISO domain removes the data center's need for physical media. An ISO domain can be shared across different data centers. ISO storage domains must be located on NFS storage.
- An export domain is a temporary storage repository that is used to copy/move images between data centers and oVirt Engine installations. In addition, the export domain can be used to backup virtual machines. An export domain can be moved between data centers, however, it can only be active in one data center at a time. Support for export storage domains backed by mechanisms other than NFS is being deprecated. New export storage domains must be created on NFS storage.

Once you have determined the storage needs of your data center(s) you must begin configuring and attaching the storage:

- To configure, and attach NFS storage, see [Section 9.1.1, “Adding NFS Storage”](#).
- To configure, and attach iSCSI storage, see [Section 9.1.2, “Adding iSCSI Storage”](#).
- To configure, and attach FCP storage, see [Section 9.1.3, “Adding FCP Storage”](#).

- To configure, and attach local storage, see [Section 9.1.4, “Adding Local Storage”](#).



Important — Before Adding Storage

Before adding storage ensure that you have a working oVirt Engine environment. You must be able to successfully access the Administration Portal, and there must be at least one host connected with a status of **Up**.



Important — Export Domain Storage Type

Support for export storage domains backed by storage on anything other than NFS is being deprecated. While existing export storage domains imported from oVirt 2.2 environments remain supported new export storage domains must be created on NFS storage.

9.1. Storage Domains Overview

9.1.1. Adding NFS Storage

An NFS storage domain is an NFS file share that is attached to a data center. Once you attach an NFS file share to the data center as a storage domain it is used to provide storage to the oVirt environment. How the storage domain is used depends on the function you select when attaching it.

This section details how to prepare NFS file shares on your storage infrastructure and attach them using the oVirt Engine. For further information on NFS itself, see the *Red Hat Enterprise Linux — Storage Administration Guide*

New Domain

Name:

Data Center: (NFS, V1)

Domain Function / Storage Type: **Format:**

Use Host:

Export path:

*Please use 'FQDN:/path' or 'IP:/path'
Example 'server.example.com:/export/VMs'*

Figure 9.1. Add NFS



Important

For best results, the network interface over which data is shared should be capable of speeds of at least 1Gb/s.

NFSv4 is not natively supported by oVirt. oVirt will always attempt to mount NFS storage using NFSv3.

Your NFS storage server must support NFSv3 to be used with oVirt. Attempts to attach NFS storage which has been exported from servers that do not support NFSv3 to the oVirt environment will fail.

Preparing NFS Storage

This section outlines how to prepare an NFS file share on a server running Red Hat Enterprise Linux 6. Once created the NFS share can be attached by the oVirt Engine.

1. Install *nfs-utils*

NFS functionality is provided by the *nfs-utils* package. Before file shares can be created, check that the package is installed by querying the RPM database for the system:

```
$ rpm -qi nfs-utils
```

If the *nfs-utils* package is installed then the package information will be displayed. If no output is displayed then the package is not currently installed. Install it using **yum** while logged in as the root user:

```
# yum install nfs-utils
```

2. Configure Boot Scripts

To ensure that NFS shares are always available when the system is operational both the *nfs* and *rpcbind* services must start at boot time. Use the **chkconfig** command while logged in as root to modify the boot scripts.

```
# chkconfig --add rpcbind
# chkconfig --add nfs
# chkconfig rpcbind on
# chkconfig nfs on
```

Once the boot script configuration has been done, start the services for the first time.

```
# service rpcbind start
# service nfs start
```

3. Create Directory

Create the directory you wish to share using NFS.

```
# mkdir /exports/iso
```

Replace `/exports/iso` with the name, and path of the directory you wish to use.

4. Export Directory

To be accessible over the network using NFS the directory must be exported. NFS exports are controlled using the `/etc/exports` configuration file. Each export path appears on a separate line followed by a tab character and any additional NFS options. Exports to be attached to the oVirt Engine must have the read, and write, options set.

To grant read, and write access to `/exports/iso` using NFS for example you add the following line to the `/etc/exports` file.

```
/exports/iso      *(rw)
```

Again, replace `/exports/iso` with the name, and path of the directory you wish to use.

5. Reload NFS Configuration

For the changes to the `/etc/exports` file to take effect the service must be told to reload the configuration. To force the service to reload the configuration run the following command as root:

```
# service nfs reload
```

6. Set Permissions

The NFS export directory must be configured for read write access and must be owned by `vdsm:kvm`. If these users do not exist on your external NFS server use the following command, assuming that `/exports/iso` is the directory to be used as an NFS share.

```
# chown -R 36:36 /exports/iso
```

The permissions on the directory must be set to allow read and write access to both the owner and the group. The owner should also have execute access to the directory. The permissions are set using the `chmod` command. The following command arguments set the required permissions on the `/exports/iso` directory.

```
# chmod 0755 /exports/iso
```

Result:

The NFS file share has been created, and is ready to be attached by the oVirt Engine.

Attaching NFS Storage

An NFS type **Storage Domain** is a mounted NFS share that is attached to a data center. It is used to provide storage for virtualized guest images and ISO boot media. Once NFS storage has been exported it must be attached to the oVirt Engine, using the Administration Portal.

To add an NFS data, or export, storage domain you must select an NFS data center. NFS storage domains for ISO storage are able to be added to data centers of any type.

1. Click the **Storage** tab. The Storage list and toolbar display.
2. Click **New Domain**.
3. The **New Storage** dialog box displays.

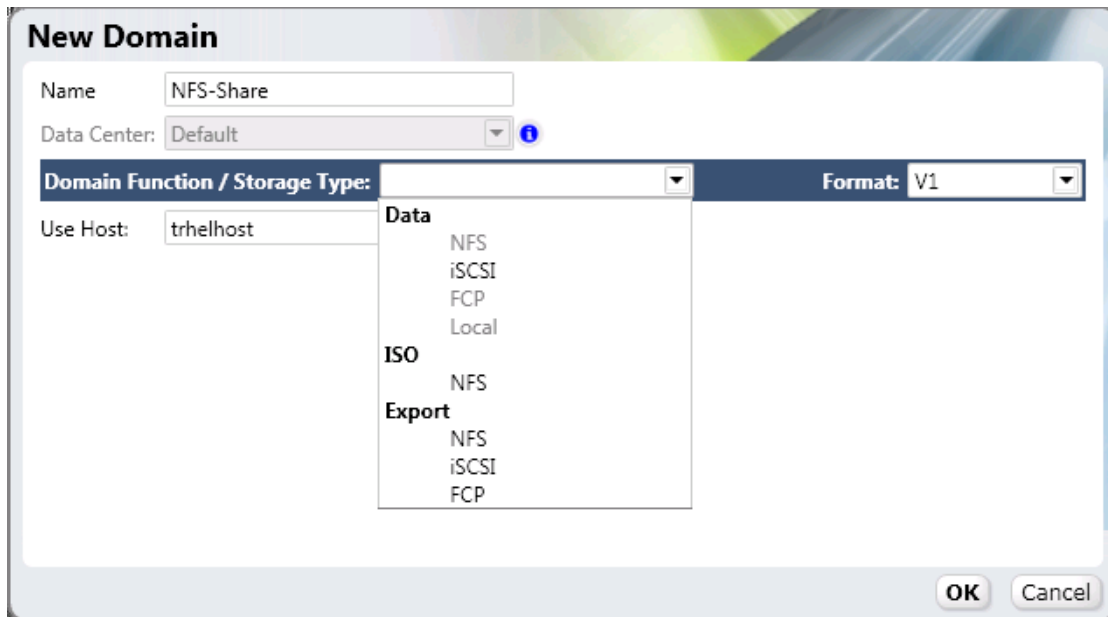


Figure 9.2. NFS Storage

- Configure the following options:

Name: Enter a suitably descriptive name.

Data Center: Select the required Data Center from the drop-down list.

Domain Function/ Storage Type: In the drop down menu, select Data → NFS. The storage domain types which are not compatible with the Default data center are grayed out. After you select your domain type, the Export Path field appears.

Export path: Enter the IP address or a resolvable hostname of the chosen host. The export path should be in the format of **192.168.0.10:/Images/ISO** or **domain.example.com:/Images/ISO**

Use Host: Select any of the hosts from the drop down menu. Only hosts which belong in the pre-selected data center will display in this list.



Active Host Required

All communication to the storage domain is via the selected host and not directly from the oVirt Engine. At least one active host must exist in the system, and be attached to the chosen data center, before the storage is configured.

- Click **OK**.

Result:

The new NFS data domain displays on the Storage tab. It will remain with a *Locked status* while it is being prepared for use. When ready, it is automatically attached to the data center.

9.1.2. Adding iSCSI Storage

oVirt platform supports iSCSI storage via the creation of a Storage Domain for a Volume Group. A Volume Group is a set of pre-defined Logical Unit Numbers (LUNs). oVirt supports creation of a

Storage Domain from a pre-existent Volume Group or a set of LUNs. Neither Volume Groups nor LUNs are able to be attached to more than one Storage Domain at a time.

For information regarding the setup and configuration of iSCSI on Red Hat Enterprise Linux, see the *Red Hat Enterprise Linux — Storage Administration Guide*.

1. On the tree pane, select the **Tree** tab. On **System**, click the **+** icon to display the available data centers.
2. Select the **Data Center** to which the domain is to be added. The storage type of the data center selected determines the type of storage domains that can be added to it. To add an iSCSI data, or export, storage domain you must select an iSCSI data center. iSCSI storage domains can not be used for ISO storage domains.
3. Click the **New Domain** button.
4. Click **New Storage**. The **New Storage** dialog box displays.
5. From the **Domain Function / Storage Type** drop-down menu, select the appropriate storage type for the storage domain. The storage domain types that are not compatible with the chosen data center are not available.
6. Select an active host in the **Use host** field. To attach a domain, the name of an active host must be selected from the list of existing hosts. Only hosts that are attached to the selected Data Center are listed.



Active Host Required

All communication to the storage domain is via the selected host and not directly from the oVirt Engine. At least one active host must exist in the system, and be attached to the chosen data center, before the storage is configured.

7. The oVirt Engine is able to map either iSCSI targets to LUNs, or LUNs to iSCSI targets. The **New Domain** dialog automatically displays known targets with unused LUNs when iSCSI is selected

as the storage type. If the target that you are adding storage from is not listed then you can use target discovery to find it, otherwise proceed to the next step.

New Domain

Name: FinanceDataDomain

Data Center: FinanceDataCenter (iSCSI)

Domain Function / Storage Type: Data / iSCSI Format: V2

Use Host: Atlantic

Discover Targets

Address: storage.demo.redhat. **User Authentication:**

Port: 3260 CHAP user name: CHAP password:

Discover Login All

Target Name	Address	Port
-------------	---------	------

OK Cancel

Figure 9.3. New Domain Dialog

iSCSI Target Discovery

- Click **Discover Targets** to enable target discovery options. The **New Domain** dialog automatically displays targets with unused LUNs when iSCSI is selected as the storage type. If the target that you are adding is not listed, click **Discover Targets** to enable target discovery options.
- Enter the fully qualified domain name or IP address of the iSCSI host in the **Address** field.
- Enter the port to connect to the host on when browsing for targets in the **Port** field. The default is **3260**.
- If the Challenge Handshake Authentication Protocol (CHAP) is being used to secure the storage, select the **User Authentication** check box. Enter the CHAP user name and password.
- Click the **Discover** button.

- Click the + button next to the desired target. This will expand the entry and display all unused LUNs attached to the target.

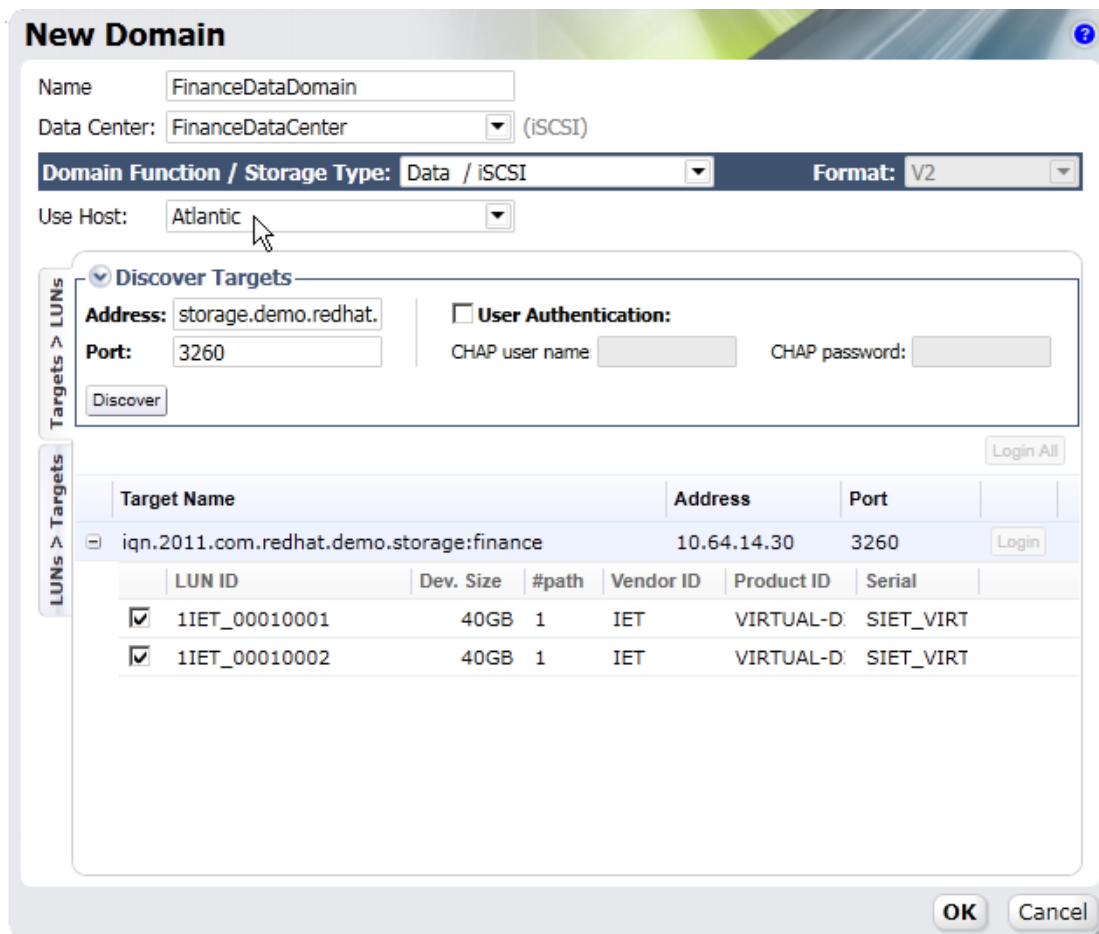


Figure 9.4. iSCSI LUN Selection

- Select the check box for each LUN that you are using to create the storage domain.
- Click **OK** to create the storage domain.

Result:

The new iSCSI storage domain displays on the storage tab. This will take some time.

9.1.2.1. Mapping iSCSI Targets to LUNs

Follow the below mentioned procedure:

- Click the + button next to the desired target.
- Select the check box for each LUN that you are using to create the storage domain.
- Click **OK**.

Result:

The new storage domain is created.

9.1.3. Adding FCP Storage

oVirt platform supports SAN storage via the creation of a Storage Domain for a Volume Group. A Volume Group is a set of pre-defined Logical Unit Numbers (LUNs). oVirt supports creation of a Storage Domain from a pre-defined Volume Group or a set of LUNs. Neither Volume Groups nor LUNs are able to be attached to more than one Storage Domain at a time.

oVirt system administrators need a working knowledge of Storage Area Networks (SAN) concepts. SAN usually uses Fibre Channel Protocol (FCP) for traffic between hosts and shared external storage. For this reason, SAN may occasionally be referred to as FCP storage.

For information regarding the setup and configuration of FCP or multipathing on Red Hat Enterprise Linux, please refer to the *Storage Administration Guide* and *DM Multipath Guide*.

Procedure 9.1. To Add FCP Storage:

1. Click the **Storage** tab. The Storage list and toolbar display.
2. Click **New Domain**.
3. The **New Domain** dialog box displays.

Figure 9.5. Adding FCP Storage

4. Configure the following options:
 1. **Name:** Enter a suitably descriptive name.
 2. **Data Center:** Select the required Data Center from the drop-down list.
 3. **Domain Function/ Storage Type:** Select **FCP**.
 4. **Use Host:** Select the IP address of either the node or Red Hat Enterprise Linux host.



Active Host Required

All communication to the storage domain is via the selected host and not directly from the oVirt Engine. At least one active host must exist in the system, and be attached to the chosen data center, before the storage is configured.

5. The list of existing LUNs display. On the selected LUN, select the Add LUN check box to use it as the FCP data domain.
5. Click **OK**.

Result:

The new FCP data domain displays on the **Storage** tab. It will remain with a Locked status while it is being prepared for use. When ready, it is automatically attached to the data center. Select either **Build New Domain** or **Use Existing Volume Group**.

9.1.4. Adding Local Storage

A local storage domain can be set up on a host, to be used as a data domain for a data center and cluster that contains only a single host. Virtual machines created in a single host cluster cannot be migrated, fenced or scheduled.

Preparing Local Storage

This section outlines how to set up a local directory with recommended settings.

- On a oVirt Node host, set up the path for the local storage as **/data/images**. This is the only path permitted for a oVirt Node. On a Red Hat Enterprise Linux host other paths are supported but the directories to support it must be manually created first.

The path must have permissions allowing read and write access to the vdsmd user and kvm group. These permissions are set automatically on the **/data/images** path for oVirt Node hosts. you must set them manually on paths to be used for local storage on Red Hat Enterprise Linux hosts.

```
# chown 36:36 /data /data/images
# chmod 0755 /data /data/images
```

- On a Red Hat Enterprise Linux host, set up the path for local storage in the **/data** directory. Any path is permitted on a Red Hat Enterprise Linux host. Follow these instructions to add local storage:
 1. On the tree pane, select the **Tree** tab. On **System**, click the **+** icon to display the available data centers.
 2. Select the **Data Center** to which the domain is to be added. The storage type of the data center selected determines the type of storage domains that can be added to it. To add a local data storage domain you must select a local data center.
 3. Click **New Domain**. The **New Domain** dialog box displays.
 4. Enter the **Name** of the domain. A descriptive name is recommended.

5. Select the **Data / Local on Host** option as the **Domain Function / Storage Type** for the storage domain.
6. Select the local host in the **Use host** field. This must be the host on which the local storage is set up.



Important — Active Host Required

All communication to the storage domain is via the selected host and not directly from the oVirt Engine. At least one active host must exist in the system, and be attached to the chosen data center, before the storage is configured.

7. Enter the Path of the storage. For example, **/data/images**.
8. Click **OK**.

Result:

The new local storage domain displays on the Storage tab. This may take a few moments.

9.2. Populate the ISO Domain

Once an ISO storage domain is defined for a data center, CD-ROM images or ISO images must be uploaded for the virtual machines to use. oVirt provides an ISO uploader tool that ensures that the images are uploaded into the correct directory path, with the correct user permissions. While an example is provided here, for full usage information see [Section B.3, “ISO Uploader”](#).

The creation of ISO images from physical media is not described in this document. It is assumed that you have access to the images required for your environment.

1. Copy the required ISO image to a temporary directory on the system running oVirt Engine.
2. Log in to the system running oVirt Engine as the `root` user.
3. Use the **ovirt-iso-uploader** command to upload the ISO image. This action will take some time, the amount of time varies depending on the size of the image being uploaded and available network bandwidth.

Example 9.1. ISO Uploader Usage

In this example the ISO image **RHEL6.iso** is uploaded to the ISO domain called **ISODomain** using NFS. The command will prompt for an administrative username and password. The username must be provided in the form *username@domain*.

```
# ovirt-iso-uploader --iso-domain=ISODomain upload RHEL6.iso
```

Result:

The ISO image is uploaded and appears in the ISO storage domain specified. It is also available in the list of available boot media when creating virtual machines in the data center which the storage domain is attached to.

9.2.1. Uploading the VirtIO and Guest Tool Image Files

When a local ISO storage domain is configured during installation the *virtio-win* ISO and Virtual Floppy Drive (VFD) images, containing the VirtIO drivers for Windows virtual machines, are automatically copied to the new storage domain. The *rhev-tools-setup* ISO, containing the oVirt Guest Tools for Windows virtual machines are also copied to the domain.

These image files provide software that can be installed on guest operating systems to improve the performance and usability of the virtual machine. The most recent *virtio-win* and *rhev-tools-setup* images are referred to by the following symbolic links on the filesystem of the oVirt Engine:

- `/usr/share/virtio-win/virtio-win.iso`
- `/usr/share/virtio-win/virtio-win.vfd`
- `/usr/share/rhev-guest-tools-iso/rhev-tools-setup.iso`

These image files must be manually uploaded to ISO storage domains that were not created locally by the installation process. You must use the **ovirt-iso-uploader** command to upload these images to your ISO storage domain. Once the image files have been uploaded they can be attached to, and used by, virtual machines.

Example 9.2. Upload of VirtIO and Guest Tool Image Files

In this example the **virtio-win.iso**, **virtio-win.vfd**, and **rhev-tools-setup.iso** image files are uploaded to the **ISODomain** ISO storage domain.

```
# ovirt-iso-uploader --iso-domain=ISODomain upload /usr/share/virtio-win/virtio-win.iso /usr/share/virtio-win/virtio-win.vfd /usr/share/rhev-guest-tools-iso/rhev-tools-setup.iso
```

Part V. Appendices

Appendix A. Directory Services

oVirt Engine is able to use both Active Directory and IPA Server for user authentication. This appendix documents the basic requirements for directory servers of either type to be added to the environment.

Information on adding or removing directory services domains to the oVirt Engine configuration, once they have been created, is available in [Section B.1, “Domain Management Tool”](#).

A.1. IPA Server

IPA is an integrated security information management solution which combines Red Hat Enterprise Linux, oVirt Directory Server, MIT Kerberos, and NTP. It provides web browser and command-line interfaces, and its numerous administration tools allow an administrator to quickly install, set up, and administer one or more servers for centralized authentication and identity management.

The latest version of IPA extends the integration of DNS, includes a Certificate System Server, an enhanced administrative framework, support for host identities, netgroups, automount by location and other features.

Installation

IPA focuses on making centralized identity and policy easy to manage in Linux and Unix environments, and includes compatibility with Windows environments. If you need assistance installing or configuring IPA, see the *Red Hat Enterprise Linux — Enterprise Identity Management Guide*.

A.1.1. Adding New Users

Because this section is devoted to getting you started with IPA quickly and easily, we have included only a limited number of examples. In this case, we have used the example of adding a new user to the system as an introduction to administering your IPA system. The methods and general approach, however, apply to nearly all IPA objects (users, groups, hosts, etc.), upon which you would perform some operation, such as add, show, find, or delete. The general syntax involved follows the same pattern: **ipa object-operation**

Use the # **ipa user-add** command to create IPA users. Numerous options are available to customize the way your IPA users are created. Use the **ipa help user** command to access the available help on operations regarding user creation. Password management can be performed as a separate operation or as part of the initial user creation process. This, and other aspects of creating IPA users, are discussed below.

Interactive Mode

In interactive mode, the user is first created and their password created separately so that they can authenticate and log in. Use the # **ipa user-add** command to create an IPA user. You can run this command with or without additional parameters. If you omit any of the required parameters, the interface will prompt you for the information.

The following example demonstrates adding a new user to IPA. In this example, the **ipa user-add** command was executed without any additional parameters; all required information was entered in interactive mode.

```
# ipa user-add
First name: Ryan
Last name: Andrews
```

```
User login [randrews]:
-----
Added user "randrews"
-----
User login: randrews
First name: Ryan
Last name: Andrews
Full name: Ryan Andrews
Display name: Ryan Andrews
Initials: RA
Home directory: /home/randrews
GECOS field: randrews
Login shell: /bin/sh
Kerberos principal: randrews@IPADOCES.ORG
UID: 1316000004
```

Type **ipa passwd <user login>** to create a password for the user. This is a temporary password, or *one-time password (OTP)*, and the user is required to change it the first time they log in. This is done intentionally, so that an administrator can reset a password for a user but they are unable to take advantage of that knowledge, because the user must change the password when they first log in.

Unattended Mode

As an integrated (or unattended) operation, you can pass the `--password` option to the **ipa user-add** command. This will force the command to prompt for an initial password. As an alternative, echo the password directly into the command:

```
# echo "secret123" | ipa user-add asmart --first=Alex --last=Smart --password
-----
Added user "asmart"
-----
User login: asmart
First name: Alex
Last name: Smart
Full name: Alex Smart
Display name: Alex Smart
Initials: AS
Home directory: /home/asmart
GECOS field: asmart
Login shell: /bin/sh
Kerberos principal: asmart@IPADOCES.ORG
UID: 1315400003
```

Performing Initial Login

You can now authenticate using the newly-created user and temporary password. Type **knit <user login>** to log in to IPA. This will prompt you for a password and then immediately request a password change.

You can browse the IPA man pages and help system to explore other IPA commands. Please take some time to become familiar with the ways other IPA objects can be created and modified.

A.2. Active Directory

This section is intended to provide a brief overview of the **Active Directory** configuration required to support the directory services requirements of oVirt. It is not to be considered as a complete guide to the use of **Active Directory**. For full configuration procedures, refer to the appropriate Microsoft documentation available at [http://technet.microsoft.com/en-us/library/dd578336\(WS.10\).aspx](http://technet.microsoft.com/en-us/library/dd578336(WS.10).aspx).

Installation

Active Directory Domain Services must be installed and available prior to commencement of the oVirt Engine installation. For **Active Directory Domain Services** installation instructions, including the creation of a new Forest and associated Domain, refer to the Microsoft documentation at: [http://technet.microsoft.com/en-au/library/cc772464\(WS.10\).aspx](http://technet.microsoft.com/en-au/library/cc772464(WS.10).aspx).

Users

The oVirt administrative user must be created prior to installation. The credentials of this user are required to complete oVirt Engine installation. It is the account used when logging into the oVirt Administration Portal to manage the system.

The oVirt administrative user must have delegated control over the Domain to:

- Join a computer to the domain.
- Modify the membership of a group.

For information on creation of user accounts refer to <http://technet.microsoft.com/en-us/library/cc732336.aspx>.

For information on delegation of control refer to <http://technet.microsoft.com/en-us/library/cc732524.aspx>.



Important — Do *not* use the Administrator user

An Active Directory user account must be created specifically for use as the oVirt administrative user. Do *not* use the Active Directory Administrator account as the oVirt Engine administrative user.

Appendix B. Additional Utilities

B.1. Domain Management Tool

oVirt Engine uses directory services to authenticate users. While during installation the engine sets up a domain named **internal** this is only used for the `admin` user. To add and remove other users from the system it is first necessary to add the directory service(s) in which they are found.

The supported directory services are Active Directory and IPA. oVirt Engine includes a domain management tool, **ovirt-manage-domains**, to add and remove domains provided by these services. In this way it is possible to grant access to the oVirt environment to users stored across multiple domains. This is true even where some users are stored in a domain managed by Active Directory and others are stored in a domain managed by IPA.

You will find the **ovirt-manage-domains** command on the machine to which oVirt Engine was installed. The **ovirt-manage-domains** command must be run as the `root` user.

B.1.1. Syntax

The usage syntax is:

```
Usage: ovirt-manage-domains -action=ACTION [options]
```

Available actions are:

add

Add a domain to the engine's directory services configuration.

edit

Edit a domain in the engine's directory services configuration.

delete

Delete a domain from the engine's directory services configuration.

validate

Validate the engine's directory services configuration. The command attempts to authenticate to each domain in the configuration using the configured username and password.

list

List the engine's current directory services configuration.

The options available to be combined with the actions on the command line are:

-domain=DOMAIN

Specifies the domain the action must be performed on. The *-domain* parameter is mandatory for *add*, *edit*, and *delete*.

-user=USER

Specifies the domain user to use. The *-user* parameter is mandatory for *add*, and optional for *edit*.

-interactive

Specifies that the domain user's password is to be provided interactively. This option, or the *-passwordFile* option, must be used to provide the password for use with the *add* action.

-passwordFile=FILE

Specifies that the domain user's password is on the first line of the provided file. This option, or the *-interactive* option, must be used to provide the password for use with the *add* action.

-configFile=FILE

Specifies an alternative configuration file that the command must load. The *-configFile* parameter is always optional.

-report

Specifies that when performing the *validate* action all validation errors encountered will be reported in full.

Common usage examples are discussed further within this guide. For full usage information consult the **ovirt-manage-domains** command's help output:

```
# ovirt-manage-domains --help
```

B.1.2. Examples

The following examples demonstrate the use of the **ovirt-manage-domains** to perform basic manipulation of the oVirt Engine domain configuration.

Example B.1. Adding Domains to Configuration

This example runs the **ovirt-manage-domains** command to add the **directory.demo.redhat.com** domain to the oVirt Engine configuration. The configuration is set to use the *admin* user when querying the domain with the password to be provided interactively.

```
# ovirt-manage-domains -action=add -domain='directory.demo.redhat.com' -user='admin' -
interactive
loaded template kr5.conf file
setting default_tkt_enctypes
setting realms
setting domain realm
success
User guid is: 80b71bae-98a1-11e0-8f20-525400866c73
Successfully added domain directory.demo.redhat.com
```

Example B.2. Edit Domain in Configuration

This example runs the **ovirt-manage-domains** command to edit the **directory.demo.redhat.com** domain in the oVirt Engine configuration. The configuration is updated to use the *admin* user when querying this domain with the password to be provided interactively.

```
# ovirt-manage-domains -action=edit -domain=directory.demo.redhat.com -user=admin -
interactive
loaded template kr5.conf file
setting default_tkt_enctypes
setting realms
setting domain realm0
success
User guid is: 80b71bae-98a1-11e0-8f20-525400866c73
Successfully edited domain directory.demo.redhat.com
```

Example B.3. Deleting Domains from Configuration

This example runs the **ovirt-manage-domains** command to remove the **directory.demo.redhat.com** domain from the oVirt Engine configuration. Users defined in the removed domain will no longer be able to authenticate with oVirt Engine. The entries for the affected users will remain defined in oVirt Engine until they are explicitly removed.

The domain being removed in this example is the last one listed in the oVirt Engine configuration. A warning is displayed highlighting this fact and that only the admin user from the **internal** domain will be able to log in until another domain is added.

```
# ovirt-manage-domains -action=delete -domain='directory.demo.redhat.com'  
WARNING: Domain directory.demo.redhat.com is the last domain in the configuration. After  
deleting it you will have to either add another domain, or to use the internal admin user  
in order to login.  
Successfully deleted domain directory.demo.redhat.com. Please remove all users and groups  
of this domain using the Administration portal or the API.
```

Example B.4. Validating Configuration

This example runs the **ovirt-manage-domains** command to validate the oVirt Engine configuration. The command attempts to log into each listed domain with the credentials provided in the configuration. If the attempt is successful then the domain is reported as valid.

```
# ovirt-manage-domains -action=validate  
User guid is: 80b71bae-98a1-11e0-8f20-525400866c73  
Domain directory.demo.redhat.com is valid.
```

Example B.5. Listing Domains in Configuration

This example runs the **ovirt-manage-domains** command to list the domains defined in the oVirt Engine configuration. For each configuration entry the command displays the domain, the username — in User Principle Name (UPN) format, and whether the domain is local or remote.

```
# ovirt-manage-domains -action=list  
Domain: directory.demo.redhat.com  
User name: admin@DIRECTORY.DEMO.REDHAT.COM  
This domain is a remote domain.
```

B.2. Configuration Tool

During installation, only a subset of oVirt Engine's configuration settings are modified from their defaults. You make further changes with the included configuration tool, **ovirt-config**.

The configuration tool does not require JBoss or oVirt Engine to be running to update the configuration. Configuration key values are stored in the database and as such it must be operational for configuration changes to be saved. Changes are only applied once JBoss is restarted.

The engine's configuration is stored as a series of key to value pair mappings. The configuration tool allows you to:

- list all available configuration keys,

- list all available configuration values,
- get the value of a specific configuration key, and
- set the value of a specific configuration key.

The configuration tool also allows you to maintain multiple versions of the engine's configuration. When getting or setting the value for a configuration key the `--cver` parameter is used to specify which configuration version is to be used. The default configuration version is **general**.

B.2.1. Syntax

You will find the configuration tool on the machine to which oVirt Engine was installed. Common usage examples are discussed within this guide. For full usage information consult the **ovirt-config** command's help output:

```
# ovirt-config --help
```

Common Tasks

List Available Configuration Keys

Use the `--list` parameter to list available configuration keys.

```
# ovirt-config --list
```

The tool lists each available configuration key by name. It also returns a description of each key's purpose.

List Available Configuration Values

Use the `--all` parameter to list available configuration values.

```
# ovirt-config --all
```

The tool lists each available configuration key by name as well as the current value of the key, and the configuration version.

Get Value of Configuration Key

Use the `--get` parameter to get the value of a specific key.

```
# ovirt-config --get KEY_NAME
```

Replace `KEY_NAME` with the name of the key to retrieve. The tool returns the key name, value, and the configuration version. Optionally the `--cver` parameter is used to specify the configuration version from which the value should be retrieved.

Set Value of Configuration Key

Use the `--set` parameter to set the value of a specific key. You must also set the configuration version to which the change is to apply using the `--cver` parameter.

```
# ovirt-config --set KEY_NAME=KEY_VALUE --cver=VERSION
```

Replace `KEY_NAME` with the name of the key to set, and replace `KEY_VALUE` with the value to assign to it. In an environment with more than one configuration version you must also take care to replace `VERSION` with the name of the configuration version in use.

B.2.2. Examples

Example B.6. Getting a Configuration Value

```
# ovirt-config --get=SearchResultsLimit --cver=general
100
```

Example B.7. Setting a Configuration Value

```
# ovirt-config --set SearchResultsLimit=50 --cver=general
```

B.3. ISO Uploader

The oVirt Engine installation includes a tool for uploading ISO images to the ISO storage domain. This tool is referred to as the ISO uploader. It provides for the listing of storage domains and uploading of ISO files to them.

The ISO uploader command is **ovirt-iso-uploader**. You must be logged in as the `root` user to run it successfully. You must provide the administration credentials for the oVirt environment on the command line. Full usage information, including a list of all valid options for the command, is available by running the **ovirt-iso-uploader -h** command.

B.3.1. Syntax

The basic syntax is of the form:

```
Usage: ovirt-iso-uploader [options] list
       ovirt-iso-uploader [options] upload [file].[file]...[file]
```

The two supported modes of operation are *list*, and *upload*.

- The *list* parameter lists the available ISO storage domains. These storage domains are the valid targets for ISO uploads. By default the list is obtained from the oVirt Engine installation on the local machine.
- The *upload* parameter uploads the selected ISO file(s) to the specified ISO storage domain. By default the transfer is performed using NFS however SSH is also available.

Basic ISO uploader usage requires that, at a minimum, the either the *list* or *upload* parameter is provided. Where *upload* is selected then the name of at least one local file to upload must also be provided.

The **ovirt-iso-uploader** command has a large number of options.

General Options

--version

Displays the version number of the command in use, and exits immediately.

-h, --help

Displays command usage information, and exits immediately.

Appendix B. Additional Utilities

`--quiet`

Sets quiet mode, reducing console output to a minimum. This is off by default.

`--log-file=PATH`

Sets *PATH* as the log file the command should use for its own log output.

`--conf-file=PATH`

Sets *PATH* as the configuration file the command should use.

`-v, --verbose`

Sets verbose mode, providing more console output. This is off by default.

`-f, --force`

Where the source file being uploaded has the same file name as an existing file at the destination, force the existing file to be overwritten automatically. This is off by default.

oVirt Engine Options

The options in the oVirt Engine configuration group are used to specify the engine authentication details and, filter log collection from one or more virtualization hosts. If no options in this group are specified, data is not collected from any virtualization host.

`-u USER, --user=USER`

Sets the user as *USER*. This must be a user that exists in directory services, and is known to the oVirt Engine. The user must be specified in the format *user@domain*, where *user* replaced by the username, and *domain* is replaced by the directory services domain in use.

`-r FQDN, --ovirt=FQDN`

Sets the oVirt Engine to connect to as *FQDN*. *FQDN* must be replaced by the fully qualified domain name of the engine. By default it is assumed that the ISO uploader is being run on the same machine as the engine. Therefore the default value for this parameter is **localhost**.

ISO Storage Domain Options

The options in this configuration group are used to specify the ISO domain to which files must be uploaded

`-i, --iso-domain=ISODOMAIN`

Sets the storage domain named *ISODOMAIN* as the destination for uploads.

`-n, --nfs-server=NFSSERVER`

Sets the NFS path of *NFSSERVER* as the destination for uploads. This option is an alternative to `--iso-domain`, the two must not be used at the same time.

Example B.8. Specifying an NFS Server

```
# ovirt-iso-uploader --nfs-server=storage.demo.redhat.com:/iso/path upload RHEL6.0.iso
```

Connection Options

By default the ISO uploader uses NFS to upload files. Use options within this configuration group to use SSH file transfer instead.

`--ssh-user=USER`

Sets *USER* as the SSH username to use for the upload.

`--ssh-port=PORT`

Sets *PORT* as the port to use when connecting to SSH.

`-k KEYFILE, --key-file=KEYFILE`

Sets *KEYFILE* as the public key to use for SSH authentication. If no key is set the program will prompt you to enter the password of the user specified instead.

B.3.2. Examples

Example B.9. Basic ISO Uploader Usage

In this example the ISO uploader is run to list the available ISO storage domains. The username is not provided on the command line so the tool instead prompts for it to be entered. Once the storage domains have been listed, an ISO file is uploaded to one of them over NFS.

```
# ovirt-iso-uploader list
Please provide the REST API username for oVirt-M (CTRL+D to
abort): admin@directory.demo.redhat.com
Please provide the REST API password for oVirt-M (CTRL+D to abort):
ISO Storage Domain List:
  ISODomain
# ovirt-iso-uploader --iso-domain=ISODomain upload RHEL6.iso
Please provide the REST API username for oVirt-M (CTRL+D to
abort): admin@directory.demo.redhat.com
Please provide the REST API password for oVirt-M (CTRL+D to abort):
```

B.4. Log Collector

The oVirt Engine installation includes a log collection tool. This allows you to easily collect relevant logs from across the oVirt environment when requesting support.

The log collection command is **ovirt-log-collector**. You must be logged in as the `root` user to run it successfully. You must provide the administration credentials for the oVirt environment on the command line. Full usage information, including a list of all valid options for the command, is available by running the **ovirt-log-collector -h** command.

B.4.1. Syntax

The basic syntax is of the form:

```
Usage: ovirt-log-collector [options] list [all, clusters, datacenters]
       ovirt-log-collector [options] collect
```

The two supported modes of operation are *list*, and *collect*.

- The *list* parameter lists either the hosts, clusters, or data centers attached to the oVirt Engine. You are then able to filter log collection based on the listed objects.
- The *collect* parameter performs log collection from the oVirt Virtualization Manager. The collected logs are placed in an archive file under the `/tmp/logcollector` directory. The **ovirt-log-collector** command outputs the specific filename that it chose to use when log collection is completed.

Appendix B. Additional Utilities

The default action taken if no parameters are provided is to list available hosts along with the data center, and cluster, to which they belong. Where necessary the log collector will prompt you to enter usernames and passwords required to retrieve logs.

The **ovirt-log-collector** command has a large number of options. You can use these options to further refine the scope of log collection.

General Options

--version

Displays the version number of the command in use, and exits immediately.

-h, --help

Displays command usage information, and exits immediately.

--conf-file=PATH

Sets *PATH* as the configuration file the tool is to use.

--local-tmp=PATH

Sets *PATH* as the directory to which retrieved logs are to be saved. Default is **/tmp/logcollector**.

--ticket-number=TICKET

Sets *TICKET* as the ticket, or case number, to associate with the SOS report.

--upload=FTP_SERVER

Sets *FTP_SERVER* as the destination for retrieved logs to be sent using FTP. Do not use this option unless advised to by a oVirt support representative.

--quiet

Sets quiet mode, reducing console output to a minimum. This is off by default.

--log-file=PATH

Sets *PATH* as the log file the command should use for its own log output. Note that this is not to be confused with the *--local-tmp* parameter.

-v, --verbose

Sets verbose mode, providing more console output. This is off by default.

oVirt Engine Options

The options in the oVirt Engine configuration group are used to specify the engine authentication details and, filter log collection from one or more virtualization hosts. Note that it is possible to combine the options used to select the virtualization hosts, for example selecting all host in clusters **A** and **B** where the name of the host matches pattern **SalesHost***.

--no-nodes

Sets the option to skip collection of logs from the virtualization hosts.

-u USER, --user=USER

Sets the username to log in as to *USER*. This must be a username that exists in directory services, and is known to the oVirt Engine. The user must be specified in the format *user@domain*, where *user* is replaced by the username, and *domain* is replaced by the directory services domain in use.

-r FQDN, --ovirt=FQDN

Sets the oVirt Engine to connect to as *FQDN*. *FQDN* must be replaced by the fully qualified domain name of the engine. By default it is assumed that the log collector is being run on the same machine as the engine. Therefore the default value for this parameter is **localhost**.

-c CLUSTER, --cluster CLUSTER

Collect all logs from the oVirt Engine, as well as virtualization hosts in the cluster named *CLUSTER*. The cluster(s) for inclusion must be specified in a comma separated list of cluster names or match patterns.

-d DATACENTER, --data-center DATACENTER

Collect all logs from the oVirt Engine, as well as virtualization hosts in the data center named *DATACENTER*. The data center(s) for inclusion must be specified as a comma separated list of data center names or match patterns.

-H HOSTS_LIST, --hosts=HOSTS_LIST

Collect all logs from the oVirt Engine, as well as virtualization hosts included in *HOSTS_LIST*. The hosts for inclusion must be specified as a comma separated list of hostnames, fully qualified domain names, or IP addresses. Match patterns for each type of value are also valid.

SOS Report Options

The JBoss SOS plugin is always executed by log collector. To activate data collection from the JMX console the *--java-home*, *--jboss-user*, and *jboss-pass* parameters must also be provided.

--jboss-home=JBASS_HOME

JBoss installation directory path. Default is **/var/lib/jbossas**.

--java-home=JAVA_HOME

Java installation directory path. Default is **/usr/lib/jvm/java**.

--jboss-profile=JBASS_PROFILE

Quoted and space separated list of server profiles. This is used to limit log collection to the specified profiles. The default is *'ovirt-slimmed'*.

--enable-jmx

Enable the collection of run-time metrics from oVirt's JBoss JMX interface.

--jboss-user=JBASS_USER

JBoss JMX invoker user to be used with twiddle. Default is *admin*.

--jboss-logsize=LOG_SIZE

Maximum size for each log file retrieved, in MB.

--jboss-stdjar=STATE

Sets collection of JAR statistics for JBoss standard JARs. Replace *STATE* with **on**, or **off**. The default is **on**.

--jboss-servjar=STATE

Sets collection of JAR statistics from any server configuration directories. Replace *STATE* with **on**, or **off**. The default is **on**.

--jboss-twiddle=STATE

Sets collection of twiddle data on, or off. Twiddle is the JBoss tool used to collect data from the JMX invoker. Replace *STATE* with **on**, or **off**. The default is **on**.

Appendix B. Additional Utilities

`--jboss-appxml=XML_LIST`

Quoted and space separated list of applications whose XML descriptions should be retrieved. Default is 'all'.

SSH Configuration

`--ssh-host=PORT`

Sets *PORT* as the port to use for SSH connections with virtualization hosts.

`-k KEYFILE, --key-file=KEYFILE`

Sets *KEYFILE* as the public SSH key to be used for accessing the virtualization hosts.

`--max-connections=MAX_CONNECTIONS`

Sets *MAX_CONNECTIONS* as the maximum concurrent SSH connections for logs from virtualization hosts. The default is **10**.

PostgreSQL Database Options

The log collector connects to the oVirt Engine database and dumps it for inclusion in the log report if *pg-pass* is specified. The database username, and database name also must be specified if they were changed from the default values during installation.

Where the database is not on the local machine set the *pg-dbhost*, and optionally supply a *pg-host-key*, to collect remote logs. The PostgreSQL SOS plugin must be installed on the database server for remote log collection to be successful.

`--no-postgresql`

Disables collection of database. Database collection is performed by default.

`--pg-user=USER`

Sets *USER* as the username to use for connections with the database server. The default is postgres.

`--pg-dbname=DBNAME`

Sets *DBNAME* as the database name to use for connections with the database server. The default is ovirt.

`--pg-dbhost=DBHOST`

Sets *DBHOST* as the hostname for the database server. The default is localhost.

`--pg-host-key=KEYFILE`

Sets *KEYFILE* as the public identity file (private key) for the database server. This value is not set by default as it is not required where the database exists on the local host.

B.4.2. Examples

Example B.10. Basic Log Collector Usage

In this example log collector is run to collect all logs from the oVirt Engine and the three attached hosts. Additionally the database and JBoss logs are also collected.

```
# ovirt-log-collector
Please provide the username for ovirt (CTRL+D to abort): admin@directory.demo.redhat.com
Please provide the password for ovirt (CTRL+D to abort):
Host list (datacenter=None, cluster=None, host=None):
Data Center      | Cluster          | Hostname/IP Address
SalesDataCenter  | SalesCluster     | 192.168.122.250
EngineeringDataCenter | EngineeringCluster | 192.168.122.251
```

```
FinanceDataCenter | FinanceCluster | 192.168.122.252
# ovirt-log-collector collect
Please provide the username for ovirt (CTRL+D to abort): admin@directory.demo.redhat.com
Please provide the password for ovirt (CTRL+D to abort):
About to collect information from 3 nodes. Continue? (Y/n): Y
INFO: Gathering information from selected nodes...
INFO: collecting information from 192.168.122.250
INFO: collecting information from 192.168.122.251
INFO: collecting information from 192.168.122.252
INFO: finished collecting information from 192.168.122.250
INFO: finished collecting information from 192.168.122.251
INFO: finished collecting information from 192.168.122.252
Please provide the password to dump the PostgreSQL database (CTRL+D to abort):
INFO: Gathering PostgreSQL the oVirt-M database and log files from localhost...
INFO: Gathering oVirt-M information...
Please provide the password for jboss (CTRL+D to abort):
INFO: Log files have been collected and placed in /tmp/logcollector/sosreport-rhn-
account-20110804121320-ce2a.tar.xz.
    The MD5 for this file is 6d741b78925998caff29020df2b2ce2a and its size is 26.7M
```

Appendix C. Revision History

Revision 1-1 **Thursday February 2 2011**

Stephen Gordon sgordon@redhat.com

Initial release for oVirt.
