



Red Hat Virtualization 4.4

Installing Red Hat Virtualization as a self-hosted engine using the command line

Using the command line to install the Red Hat Virtualization Manager as a virtual machine running on the same hosts it manages

Red Hat Virtualization 4.4 Installing Red Hat Virtualization as a self-hosted engine using the command line

Using the command line to install the Red Hat Virtualization Manager as a virtual machine running on the same hosts it manages

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

Legal Notice

Copyright © 2024 Red Hat, Inc.

The text of and illustrations in this document are licensed by Red Hat under a Creative Commons Attribution–Share Alike 3.0 Unported license ("CC-BY-SA"). An explanation of CC-BY-SA is available at

<http://creativecommons.org/licenses/by-sa/3.0/>

. In accordance with CC-BY-SA, if you distribute this document or an adaptation of it, you must provide the URL for the original version.

Red Hat, as the licensor of this document, waives the right to enforce, and agrees not to assert, Section 4d of CC-BY-SA to the fullest extent permitted by applicable law.

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

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

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

XFS[®] is a trademark of Silicon Graphics International Corp. or its subsidiaries in the United States and/or other countries.

MySQL[®] is a registered trademark of MySQL AB in the United States, the European Union and other countries.

Node.js[®] is an official trademark of Joyent. Red Hat is not formally related to or endorsed by the official Joyent Node.js open source or commercial project.

The OpenStack[®] Word Mark and OpenStack logo are either registered trademarks/service marks or trademarks/service marks of the OpenStack Foundation, in the United States and other countries and are used with the OpenStack Foundation's permission. We are not affiliated with, endorsed or sponsored by the OpenStack Foundation, or the OpenStack community.

All other trademarks are the property of their respective owners.

Abstract

This document describes how to install a self-hosted engine environment – where the Red Hat Virtualization Manager (or "engine") is installed on a virtual machine that runs on specialized hosts in the same environment it manages – using the command line to configure and run an automated installation. If this is not the configuration you want to use, see the other Installation Options in the Product Guide.

Table of Contents

PREFACE	5
RED HAT VIRTUALIZATION KEY COMPONENTS	5
SELF-HOSTED ENGINE ARCHITECTURE	5
CHAPTER 1. INSTALLATION OVERVIEW	7
CHAPTER 2. REQUIREMENTS	9
2.1. RED HAT VIRTUALIZATION MANAGER REQUIREMENTS	9
2.1.1. Hardware Requirements	9
2.1.2. Browser Requirements	9
2.1.3. Client Requirements	10
2.1.4. Operating System Requirements	10
2.2. HOST REQUIREMENTS	11
2.2.1. CPU Requirements	11
2.2.1.1. Checking if a Processor Supports the Required Flags	11
2.2.2. Memory Requirements	12
2.2.3. Storage Requirements	12
2.2.4. PCI Device Requirements	13
2.2.5. Device Assignment Requirements	13
2.2.6. vGPU Requirements	14
2.3. NETWORKING REQUIREMENTS	14
2.3.1. General requirements	14
2.3.2. Network range for self-hosted engine deployment	14
2.3.3. Firewall Requirements for DNS, NTP, and IPMI Fencing	14
2.3.4. Red Hat Virtualization Manager Firewall Requirements	15
2.3.5. Host Firewall Requirements	19
2.3.6. Database Server Firewall Requirements	24
2.3.7. Maximum Transmission Unit Requirements	24
CHAPTER 3. PREPARING STORAGE FOR RED HAT VIRTUALIZATION	25
3.1. PREPARING NFS STORAGE	26
3.2. PREPARING ISCSI STORAGE	27
3.3. PREPARING FCP STORAGE	28
3.4. PREPARING RED HAT GLUSTER STORAGE	29
3.5. CUSTOMIZING MULTIPATH CONFIGURATIONS FOR SAN VENDORS	29
3.6. RECOMMENDED SETTINGS FOR MULTIPATH.CONF	30
CHAPTER 4. INSTALLING THE SELF-HOSTED ENGINE DEPLOYMENT HOST	32
4.1. INSTALLING RED HAT VIRTUALIZATION HOSTS	32
4.1.1. Enabling the Red Hat Virtualization Host Repository	34
4.2. INSTALLING RED HAT ENTERPRISE LINUX HOSTS	34
4.2.1. Enabling the Red Hat Enterprise Linux host Repositories	35
CHAPTER 5. INSTALLING THE RED HAT VIRTUALIZATION MANAGER	38
5.1. MANUALLY INSTALLING THE RHV-M APPLIANCE	38
5.2. ENABLING AND CONFIGURING THE FIREWALL	39
5.3. DEPLOYING THE SELF-HOSTED ENGINE USING THE COMMAND LINE	39
5.4. ENABLING THE RED HAT VIRTUALIZATION MANAGER REPOSITORIES	47
5.5. CONNECTING TO THE ADMINISTRATION PORTAL	49
CHAPTER 6. INSTALLING HOSTS FOR RED HAT VIRTUALIZATION	50
6.1. RED HAT VIRTUALIZATION HOSTS	50
6.1.1. Installing Red Hat Virtualization Hosts	50

6.1.2. Enabling the Red Hat Virtualization Host Repository	52
6.1.3. Advanced Installation	53
6.1.3.1. Custom Partitioning	53
6.1.3.2. Installing a DUD driver on a host without installer support	54
6.1.3.3. Automating Red Hat Virtualization Host deployment	57
6.1.3.3.1. Preparing the installation environment	58
6.1.3.3.2. Configuring the PXE server and the boot loader	58
6.1.3.3.3. Creating and running a Kickstart file	59
6.2. RED HAT ENTERPRISE LINUX HOSTS	62
6.2.1. Installing Red Hat Enterprise Linux hosts	62
6.2.2. Enabling the Red Hat Enterprise Linux host Repositories	62
6.2.3. Installing Cockpit on Red Hat Enterprise Linux hosts	63
6.3. RECOMMENDED PRACTICES FOR CONFIGURING HOST NETWORKS	64
6.4. ADDING SELF-HOSTED ENGINE NODES TO THE RED HAT VIRTUALIZATION MANAGER	65
6.5. ADDING STANDARD HOSTS TO THE RED HAT VIRTUALIZATION MANAGER	66
CHAPTER 7. ADDING STORAGE FOR RED HAT VIRTUALIZATION	68
7.1. ADDING NFS STORAGE	68
7.2. ADDING ISCSI STORAGE	69
7.3. ADDING FCP STORAGE	71
7.4. ADDING RED HAT GLUSTER STORAGE	72
APPENDIX A. TROUBLESHOOTING A SELF-HOSTED ENGINE DEPLOYMENT	73
A.1. TROUBLESHOOTING THE MANAGER VIRTUAL MACHINE	73
Engine status: "health": "good", "vm": "up" "detail": "up"	73
Engine status: "reason": "failed liveness check", "health": "bad", "vm": "up", "detail": "up"	73
Engine status: "vm": "down", "health": "bad", "detail": "unknown", "reason": "vm not running on this host"	74
Engine status: "vm": "unknown", "health": "unknown", "detail": "unknown", "reason": "failed to getVmStats"	74
Engine status: The self-hosted engine's configuration has not been retrieved from shared storage	75
Additional Troubleshooting Commands	75
A.2. CLEANING UP A FAILED SELF-HOSTED ENGINE DEPLOYMENT	75
APPENDIX B. CUSTOMIZING THE MANAGER VIRTUAL MACHINE USING AUTOMATION DURING DEPLOYMENT	77
APPENDIX C. MIGRATING DATABASES AND SERVICES TO A REMOTE SERVER	78
C.1. MIGRATING THE DATA WAREHOUSE TO A SEPARATE MACHINE	78
C.1.1. Migrating the Data Warehouse Database to a Separate Machine	78
C.1.1.1. Enabling the Red Hat Virtualization Manager Repositories	78
C.1.1.2. Migrating the Data Warehouse Database to a Separate Machine	80
C.1.2. Migrating the Data Warehouse Service to a Separate Machine	81
C.1.2.1. Setting up the New Data Warehouse Machine	81
C.1.2.2. Stopping the Data Warehouse Service on the Manager Machine	82
C.1.2.3. Configuring the New Data Warehouse Machine	83
C.1.2.4. Disabling the Data Warehouse Service on the Manager Machine	84
APPENDIX D. CONFIGURING A HOST FOR PCI PASSTHROUGH	86
APPENDIX E. PREVENTING KERNEL MODULES FROM LOADING AUTOMATICALLY	88
E.1. REMOVING A MODULE TEMPORARILY	89
APPENDIX F. SECURING RED HAT VIRTUALIZATION	91
F.1. APPLYING THE DISA STIG PROFILE IN RHEL BASED HOSTS AND THE STANDALONE MANAGER	91
F.1.1. Enabling DISA STIG in a self-hosted engine	91
F.2. APPLYING THE PCI-DSS PROFILE IN RHV HOSTS AND THE STANDALONE MANAGER	92

F.2.1. Enabling PCI-DSS in a self-hosted engine	92
APPENDIX G. DEFINING ALLOWED CPU TYPES IN SELF-HOSTED ENGINE DEPLOYMENT	93
APPENDIX H. LEGAL NOTICE	96

PREFACE

Self-hosted engine installation is automated using Ansible. The installation script (**hosted-engine --deploy**) runs on an initial deployment host, and the Red Hat Virtualization Manager (or "engine") is installed and configured on a virtual machine that is created on the deployment host. The Manager and Data Warehouse databases are installed on the Manager virtual machine, but can be migrated to a separate server post-installation if required.

Hosts that can run the Manager virtual machine are referred to as self-hosted engine nodes. At least two self-hosted engine nodes are required to support the high availability feature.

A storage domain dedicated to the Manager virtual machine is referred to as the self-hosted engine storage domain. This storage domain is created by the installation script, so the underlying storage must be prepared before beginning the installation.

See the [Planning and Prerequisites Guide](#) for information on environment options and recommended configuration. See [Self-Hosted Engine Recommendations](#) for configuration specific to a self-hosted engine environment.

RED HAT VIRTUALIZATION KEY COMPONENTS

Component Name	Description
Red Hat Virtualization Manager	A service that provides a graphical user interface and a REST API to manage the resources in the environment. The Manager is installed on a physical or virtual machine running Red Hat Enterprise Linux.
Hosts	Red Hat Enterprise Linux hosts (RHEL hosts) and Red Hat Virtualization Hosts (image-based hypervisors) are the two supported types of host. Hosts use Kernel-based Virtual Machine (KVM) technology and provide resources used to run virtual machines.
Shared Storage	A storage service is used to store the data associated with virtual machines.
Data Warehouse	A service that collects configuration information and statistical data from the Manager.

SELF-HOSTED ENGINE ARCHITECTURE

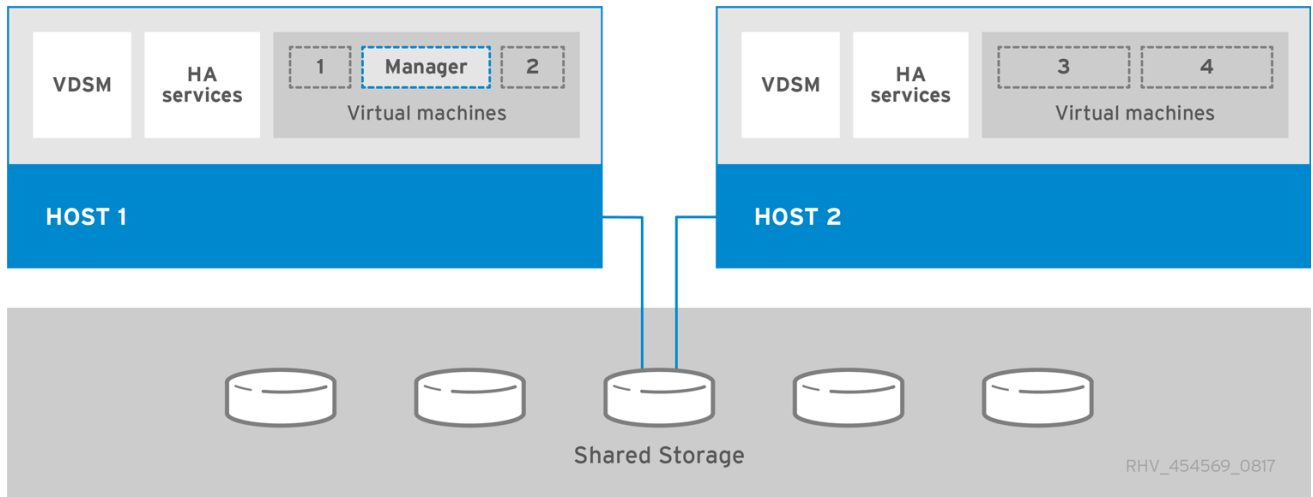
The Red Hat Virtualization Manager runs as a virtual machine on self-hosted engine nodes (specialized hosts) in the same environment it manages. A self-hosted engine environment requires one less physical server, but requires more administrative overhead to deploy and manage. The Manager is highly available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

The minimum setup of a self-hosted engine environment includes:

- One Red Hat Virtualization Manager virtual machine that is hosted on the self-hosted engine nodes. The RHV-M Appliance is used to automate the installation of a Red Hat Enterprise Linux 8 virtual machine, and the Manager on that virtual machine.

- A minimum of two self-hosted engine nodes for virtual machine high availability. You can use Red Hat Enterprise Linux hosts or Red Hat Virtualization Hosts (RHVH). VDSM (the host agent) runs on all hosts to facilitate communication with the Red Hat Virtualization Manager. The HA services run on all self-hosted engine nodes to manage the high availability of the Manager virtual machine.
- One storage service, which can be hosted locally or on a remote server, depending on the storage type used. The storage service must be accessible to all hosts.

Figure 1. Self-Hosted Engine Red Hat Virtualization Architecture



CHAPTER 1. INSTALLATION OVERVIEW

The self-hosted engine installation uses Ansible and the RHV-M Appliance (a pre-configured Manager virtual machine image) to automate the following tasks:

- Configuring the first self-hosted engine node
- Installing a Red Hat Enterprise Linux virtual machine on that node
- Installing and configuring the Red Hat Virtualization Manager on that virtual machine
- Configuring the self-hosted engine storage domain



NOTE

The RHV-M Appliance is only used during installation. It is not used to upgrade the Manager.

Installing a self-hosted engine environment involves the following steps:

1. [Prepare storage to use for the self-hosted engine storage domain and for standard storage domains.](#) You can use one of the following storage types:
 - [NFS](#)
 - [iSCSI](#)
 - [Fibre Channel \(FCP\)](#)
 - [Red Hat Gluster Storage](#)
2. [Install a deployment host to run the installation on.](#) This host will become the first self-hosted engine node. You can use either host type:
 - [Red Hat Virtualization Host](#)
 - [Red Hat Enterprise Linux](#)
3. [Install and configure the Red Hat Virtualization Manager:](#)
 - a. [Enabling and configuring the firewall](#)
 - b. [Install the self-hosted engine using the `hosted-engine --deploy` command on the deployment host.](#)
 - c. [Register the Manager with the Content Delivery Network and enable the Red Hat Virtualization Manager repositories.](#)
 - d. [Connect to the Administration Portal to add hosts and storage domains.](#)
4. [Add more self-hosted engine nodes and standard hosts to the Manager.](#) Self-hosted engine nodes can run the Manager virtual machine and other virtual machines. Standard hosts can run all other virtual machines, but not the Manager virtual machine.
 - a. Use either host type, or both:
 - [Red Hat Virtualization Host](#)

- [Red Hat Enterprise Linux](#)
 - b. [Add hosts to the Manager as self-hosted engine nodes.](#)
 - c. [Add hosts to the Manager as standard hosts.](#)
5. [Add more storage domains to the Manager.](#) The self-hosted engine storage domain is not recommended for use by anything other than the Manager virtual machine.
 6. If you want to host any databases or services on a server separate from the Manager, [you can migrate them after the installation is complete.](#)



IMPORTANT

Keep the environment up to date. See [How do I update my Red Hat Virtualization system?](#) for more information. Since bug fixes for known issues are frequently released, use scheduled tasks to update the hosts and the Manager.

CHAPTER 2. REQUIREMENTS

2.1. RED HAT VIRTUALIZATION MANAGER REQUIREMENTS

2.1.1. Hardware 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.

Hardware certification for Red Hat Virtualization is covered by the hardware certification for Red Hat Enterprise Linux. For more information, see [Does Red Hat Virtualization also have hardware certification?](#). To confirm whether specific hardware items are certified for use with Red Hat Enterprise Linux, see [Red Hat certified hardware](#).

Table 2.1. Red Hat Virtualization Manager Hardware Requirements

Resource	Minimum	Recommended
CPU	A dual core x86_64 CPU.	A quad core x86_64 CPU or multiple dual core x86_64 CPUs.
Memory	4 GB of available system RAM if Data Warehouse is not installed and if memory is not being consumed by existing processes.	16 GB of system RAM.
Hard Disk	25 GB of locally accessible, writable disk space.	50 GB of locally accessible, writable disk space. You can use the RHV Manager History Database Size Calculator to calculate the appropriate disk space for the Manager history database size.
Network Interface	1 Network Interface Card (NIC) with bandwidth of at least 1 Gbps.	1 Network Interface Card (NIC) with bandwidth of at least 1 Gbps.

2.1.2. Browser Requirements

The following browser versions and operating systems can be used to access the Administration Portal and the VM Portal.

Browser support is divided into tiers:

- Tier 1: Browser and operating system combinations that are fully tested and fully supported. Red Hat Engineering is committed to fixing issues with browsers on this tier.
- Tier 2: Browser and operating system combinations that are partially tested, and are likely to work. Limited support is provided for this tier. Red Hat Engineering will attempt to fix issues with browsers on this tier.

- Tier 3: Browser and operating system combinations that are not tested, but may work. Minimal support is provided for this tier. Red Hat Engineering will attempt to fix only minor issues with browsers on this tier.

Table 2.2. Browser Requirements

Support Tier	Operating System Family	Browser
Tier 1	Red Hat Enterprise Linux	Mozilla Firefox Extended Support Release (ESR) version
	Any	Most recent version of Google Chrome, Mozilla Firefox, or Microsoft Edge
Tier 2		
Tier 3	Any	Earlier versions of Google Chrome or Mozilla Firefox
	Any	Other browsers

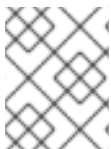
2.1.3. Client Requirements

Virtual machine consoles can only be accessed using supported Remote Viewer (**virt-viewer**) clients on Red Hat Enterprise Linux and Windows. To install **virt-viewer**, see [Installing Supporting Components on Client Machines](#) in the *Virtual Machine Management Guide*. Installing **virt-viewer** requires Administrator privileges.

You can access virtual machine consoles using the SPICE, VNC, or RDP (Windows only) protocols. You can install the QXLDDOD graphical driver in the guest operating system to improve the functionality of SPICE. SPICE currently supports a maximum resolution of 2560x1600 pixels.

Client Operating System SPICE Support

Supported QXLDDOD drivers are available on Red Hat Enterprise Linux 7.2 and later, and Windows 10.



NOTE

SPICE may work with Windows 8 or 8.1 using QXLDDOD drivers, but it is neither certified nor tested.

2.1.4. Operating System Requirements

The Red Hat Virtualization Manager must be installed on a base installation of Red Hat Enterprise Linux 8.6.

Do not install any additional packages after the base installation, as they may cause dependency issues when attempting to install the packages required by the Manager.

Do not enable additional repositories other than those required for the Manager installation.

2.2. HOST REQUIREMENTS

Hardware certification for Red Hat Virtualization is covered by the hardware certification for Red Hat Enterprise Linux. For more information, see [Does Red Hat Virtualization also have hardware certification?](#). To confirm whether specific hardware items are certified for use with Red Hat Enterprise Linux, see [Find a certified solution](#).

For more information on the requirements and limitations that apply to guests see [Red Hat Enterprise Linux Technology Capabilities and Limits](#) and [Supported Limits for Red Hat Virtualization](#).

2.2.1. CPU Requirements

All CPUs must have support for the Intel® 64 or AMD64 CPU extensions, and the AMD-V™ or Intel VT® hardware virtualization extensions enabled. Support for the No eXecute flag (NX) is also required.

The following CPU models are supported:

- AMD
 - Opteron G4
 - Opteron G5
 - EPYC
- Intel
 - Nehalem
 - Westmere
 - SandyBridge
 - IvyBridge
 - Haswell
 - Broadwell
 - Skylake Client
 - Skylake Server
 - Cascadelake Server

For each CPU model with security updates, the **CPU Type** lists a basic type and a secure type. For example:

- **Intel Cascadelake Server Family**
- **Secure Intel Cascadelake Server Family**

The Secure CPU type contains the latest updates. For details, see [BZ#1731395](#)

2.2.1.1. Checking if a Processor Supports the Required Flags

You must enable virtualization in the BIOS. Power off and reboot the host after this change to ensure that the change is applied.

Procedure

1. At the Red Hat Enterprise Linux or Red Hat Virtualization Host boot screen, press any key and select the **Boot** or **Boot with serial console** entry from the list.
2. Press **Tab** to edit the kernel parameters for the selected option.
3. Ensure there is a space after the last kernel parameter listed, and append the parameter **rescue**.
4. Press **Enter** to boot into rescue mode.
5. At the prompt, determine that your processor has the required extensions and that they are enabled by running this command:

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

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

2.2.2. Memory Requirements

The minimum required RAM is 2 GB. For cluster levels 4.2 to 4.5, the maximum supported RAM per VM in Red Hat Virtualization Host is 6 TB. For cluster levels 4.6 to 4.7, the maximum supported RAM per VM in Red Hat Virtualization Host is 16 TB.

However, the amount of RAM required varies depending on guest operating system requirements, guest application requirements, and guest memory activity and usage. KVM can also overcommit physical RAM for virtualized guests, allowing you to provision guests with RAM requirements greater than what is physically present, on the assumption that the guests are not all working concurrently at peak load. KVM does this by only allocating RAM for guests as required and shifting underutilized guests into swap.

2.2.3. Storage Requirements

Hosts require storage to store configuration, logs, kernel dumps, and for use as swap space. Storage can be local or network-based. Red Hat Virtualization Host (RHVH) can boot with one, some, or all of its default allocations in network storage. Booting from network storage can result in a freeze if there is a network disconnect. Adding a drop-in multipath configuration file can help address losses in network connectivity. If RHVH boots from SAN storage and loses connectivity, the files become read-only until network connectivity restores. Using network storage might result in a performance downgrade.

The minimum storage requirements of RHVH are documented in this section. The storage requirements for Red Hat Enterprise Linux hosts vary based on the amount of disk space used by their existing configuration but are expected to be greater than those of RHVH.

The minimum storage requirements for host installation are listed below. However, use the default allocations, which use more storage space.

- / (root) - 6 GB
- /home - 1 GB

- /tmp - 1 GB
- /boot - 1 GB
- /var - 5 GB
- /var/crash - 10 GB
- /var/log - 8 GB
- /var/log/audit - 2 GB
- /var/tmp - 10 GB
- swap - 1 GB. See [What is the recommended swap size for Red Hat platforms?](#) for details.
- Anaconda reserves 20% of the thin pool size within the volume group for future metadata expansion. This is to prevent an out-of-the-box configuration from running out of space under normal usage conditions. Overprovisioning of thin pools during installation is also not supported.
- **Minimum Total - 64 GiB**

If you are also installing the RHV-M Appliance for self-hosted engine installation, **/var/tmp** must be at least 10 GB.

If you plan to use memory overcommitment, add enough swap space to provide virtual memory for all of virtual machines. See [Memory Optimization](#).

2.2.4. PCI Device Requirements

Hosts must have at least one network interface with a minimum bandwidth of 1 Gbps. Each host should have two network interfaces, with one dedicated to supporting network-intensive activities, such as virtual machine migration. The performance of such operations is limited by the bandwidth available.

For information about how to use PCI Express and conventional PCI devices with Intel Q35-based virtual machines, see [Using PCI Express and Conventional PCI Devices with the Q35 Virtual Machine](#).

2.2.5. Device Assignment Requirements

If you plan to implement device assignment and PCI passthrough so that a virtual machine can use a specific PCIe device from a host, ensure the following requirements are met:

- CPU must support IOMMU (for example, VT-d or AMD-Vi). IBM POWER8 supports IOMMU by default.
- Firmware must support IOMMU.
- CPU root ports used must support ACS or ACS-equivalent capability.
- PCIe devices must support ACS or ACS-equivalent capability.
- All PCIe switches and bridges between the PCIe device and the root port should support ACS. For example, if a switch does not support ACS, all devices behind that switch share the same IOMMU group, and can only be assigned to the same virtual machine.
- For GPU support, Red Hat Enterprise Linux 8 supports PCI device assignment of PCIe-based NVIDIA K-Series Quadro (model 2000 series or higher), GRID, and Tesla as non-VGA graphics

devices. Currently up to two GPUs may be attached to a virtual machine in addition to one of the standard, emulated VGA interfaces. The emulated VGA is used for pre-boot and installation and the NVIDIA GPU takes over when the NVIDIA graphics drivers are loaded. Note that the NVIDIA Quadro 2000 is not supported, nor is the Quadro K420 card.

Check vendor specification and datasheets to confirm that your hardware meets these requirements. The **lspci -v** command can be used to print information for PCI devices already installed on a system.

2.2.6. vGPU Requirements

A host must meet the following requirements in order for virtual machines on that host to use a vGPU:

- vGPU-compatible GPU
- GPU-enabled host kernel
- Installed GPU with correct drivers
- Select a vGPU type and the number of instances that you would like to use with this virtual machine using the **Manage vGPU** dialog in the **Administration Portal Host Devices** tab of the virtual machine.
- vGPU-capable drivers installed on each host in the cluster
- vGPU-supported virtual machine operating system with vGPU drivers installed

2.3. NETWORKING REQUIREMENTS

2.3.1. General requirements

Red Hat Virtualization requires IPv6 to remain enabled on the physical or virtual machine running the Manager. [Do not disable IPv6](#) on the Manager machine, even if your systems do not use it.

2.3.2. Network range for self-hosted engine deployment

The self-hosted engine deployment process temporarily uses a **/24** network address under **192.168**. It defaults to **192.168.222.0/24**, and if this address is in use, it tries other **/24** addresses under **192.168** until it finds one that is not in use. If it does not find an unused network address in this range, deployment fails.

When installing the self-hosted engine using the command line, you can set the deployment script to use an alternate **/24** network range with the option **--ansible-extra-vars=he_ipv4_subnet_prefix=PREFIX**, where **PREFIX** is the prefix for the default range. For example:

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



NOTE

You can only set another range by installing Red Hat Virtualization as a self-hosted engine using the command line.

2.3.3. Firewall Requirements for DNS, NTP, and IPMI Fencing

The firewall requirements for all of the following topics are special cases that require individual consideration.

DNS and NTP

Red Hat Virtualization does not create a DNS or NTP server, so the firewall does not need to have open ports for incoming traffic.

By default, Red Hat Enterprise Linux allows outbound traffic to DNS and NTP on any destination address. If you disable outgoing traffic, define exceptions for requests that are sent to DNS and NTP servers.



IMPORTANT

- The Red Hat Virtualization Manager and all hosts (Red Hat Virtualization Host and Red Hat Enterprise Linux host) must have a fully qualified domain name and full, perfectly-aligned forward and reverse name resolution.
- Running a DNS service as a virtual machine in the Red Hat Virtualization environment is not supported. All DNS services the Red Hat Virtualization environment uses must be hosted outside of the environment.
- Use DNS instead of the `/etc/hosts` file for name resolution. Using a hosts file typically requires more work and has a greater chance for errors.

IPMI and Other Fencing Mechanisms (optional)

For IPMI (Intelligent Platform Management Interface) and other fencing mechanisms, the firewall does not need to have open ports for incoming traffic.

By default, Red Hat Enterprise Linux allows outbound IPMI traffic to ports on any destination address. If you disable outgoing traffic, make exceptions for requests being sent to your IPMI or fencing servers.

Each Red Hat Virtualization Host and Red Hat Enterprise Linux host in the cluster must be able to connect to the fencing devices of all other hosts in the cluster. If the cluster hosts are experiencing an error (network error, storage error...) and cannot function as hosts, they must be able to connect to other hosts in the data center.

The specific port number depends on the type of the fence agent you are using and how it is configured.

The firewall requirement tables in the following sections do not represent this option.

2.3.4. Red Hat Virtualization Manager Firewall Requirements

The Red Hat Virtualization Manager requires that a number of ports be opened to allow network traffic through the system's firewall.

The `engine-setup` script can configure the firewall automatically.

The firewall configuration documented here assumes a default configuration.



NOTE

A diagram of these firewall requirements is available at <https://access.redhat.com/articles/3932211>. You can use the IDs in the table to look up connections in the diagram.

Table 2.3. Red Hat Virtualization Manager Firewall Requirements

ID	Port(s)	Protocol	Source	Destination	Purpose	Encrypted by default
M1	-	ICMP	Red Hat Virtualization Hosts Red Hat Enterprise Linux hosts	Red Hat Virtualization Manager	Optional. May help in diagnosis.	No
M2	22	TCP	System(s) used for maintenance of the Manager including backend configuration, and software upgrades.	Red Hat Virtualization Manager	Secure Shell (SSH) access. Optional.	Yes
M3	2222	TCP	Clients accessing virtual machine serial consoles.	Red Hat Virtualization Manager	Secure Shell (SSH) access to enable connection to virtual machine serial consoles.	Yes
M4	80, 443	TCP	Administration Portal clients VM Portal clients Red Hat Virtualization Hosts Red Hat Enterprise Linux hosts REST API clients	Red Hat Virtualization Manager	Provides HTTP (port 80, not encrypted) and HTTPS (port 443, encrypted) access to the Manager. HTTP redirects connections to HTTPS.	Yes

ID	Port(s)	Protocol	Source	Destination	Purpose	Encrypted by default
M5	6100	TCP	Administration Portal clients VM Portal clients	Red Hat Virtualization Manager	Provides websocket proxy access for a web-based console client, noVNC , when the websocket proxy is running on the Manager.	No
M6	7410	UDP	Red Hat Virtualization Hosts Red Hat Enterprise Linux hosts	Red Hat Virtualization Manager	If Kdump is enabled on the hosts, open this port for the fence_kdump listener on the Manager. See fence_kdump Advanced Configuration . fence_kdump doesn't provide a way to encrypt the connection. However, you can manually configure this port to block access from hosts that are not eligible.	No
M7	54323	TCP	Administration Portal clients	Red Hat Virtualization Manager (ovirt-imageio service)	Required for communication with the ovirt-imageio service.	Yes

ID	Port(s)	Protocol	Source	Destination	Purpose	Encrypted by default
M8	6642	TCP	Red Hat Virtualization Hosts Red Hat Enterprise Linux hosts	Open Virtual Network (OVN) southbound database	Connect to Open Virtual Network (OVN) database	Yes
M9	9696	TCP	Clients of external network provider for OVN	External network provider for OVN	OpenStack Networking API	Yes, with configuration generated by engine-setup.
M10	35357	TCP	Clients of external network provider for OVN	External network provider for OVN	OpenStack Identity API	Yes, with configuration generated by engine-setup.
M11	53	TCP, UDP	Red Hat Virtualization Manager	DNS Server	DNS lookup requests from ports above 1023 to port 53, and responses. Open by default.	No
M12	123	UDP	Red Hat Virtualization Manager	NTP Server	NTP requests from ports above 1023 to port 123, and responses. Open by default.	No



NOTE

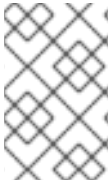
- A port for the OVN northbound database (6641) is not listed because, in the default configuration, the only client for the OVN northbound database (6641) is **ovirt-provider-ovn**. Because they both run on the same host, their communication is not visible to the network.
- By default, Red Hat Enterprise Linux allows outbound traffic to DNS and NTP on any destination address. If you disable outgoing traffic, make exceptions for the Manager to send requests to DNS and NTP servers. Other nodes may also require DNS and NTP. In that case, consult the requirements for those nodes and configure the firewall accordingly.

2.3.5. Host Firewall Requirements

Red Hat Enterprise Linux hosts and Red Hat Virtualization Hosts (RHVH) require a number of ports to be opened to allow network traffic through the system's firewall. The firewall rules are automatically configured by default when adding a new host to the Manager, overwriting any pre-existing firewall configuration.

To disable automatic firewall configuration when adding a new host, clear the **Automatically configure host firewall** check box under **Advanced Parameters**.

To customize the host firewall rules, see [RHV: How to customize the Host's firewall rules?](#) .



NOTE

A diagram of these firewall requirements is available at [Red Hat Virtualization: Firewall Requirements Diagram](#). You can use the IDs in the table to look up connections in the diagram.

Table 2.4. Virtualization Host Firewall Requirements

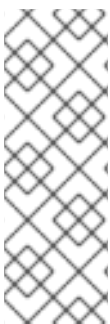
ID	Port(s)	Protocol	Source	Destination	Purpose	Encrypted by default
H1	22	TCP	Red Hat Virtualization Manager	Red Hat Virtualization Hosts Red Hat Enterprise Linux hosts	Secure Shell (SSH) access. Optional.	Yes
H2	2223	TCP	Red Hat Virtualization Manager	Red Hat Virtualization Hosts Red Hat Enterprise Linux hosts	Secure Shell (SSH) access to enable connection to virtual machine serial consoles.	Yes

ID	Port(s)	Protocol	Source	Destination	Purpose	Encrypted by default
H3	161	UDP	Red Hat Virtualization Hosts Red Hat Enterprise Linux hosts	Red Hat Virtualization Manager	Simple network management protocol (SNMP). Only required if you want Simple Network Management Protocol traps sent from the host to one or more external SNMP managers. Optional.	No
H4	111	TCP	NFS storage server	Red Hat Virtualization Hosts Red Hat Enterprise Linux hosts	NFS connections. Optional.	No
H5	5900 - 6923	TCP	Administration Portal clients VM Portal clients	Red Hat Virtualization Hosts Red Hat Enterprise Linux hosts	Remote guest console access via VNC and SPICE. These ports must be open to facilitate client access to virtual machines.	Yes (optional)

ID	Port(s)	Protocol	Source	Destination	Purpose	Encrypted by default
H6	5989	TCP, UDP	Common Information Model Object Manager (CIMOM)	Red Hat Virtualization Hosts Red Hat Enterprise Linux hosts	Used by Common Information Model Object Managers (CIMOM) to monitor virtual machines running on the host. Only required if you want to use a CIMOM to monitor the virtual machines in your virtualization environment. Optional.	No
H7	9090	TCP	Red Hat Virtualization Manager Client machines	Red Hat Virtualization Hosts Red Hat Enterprise Linux hosts	Required to access the Cockpit web interface, if installed.	Yes
H8	16514	TCP	Red Hat Virtualization Hosts Red Hat Enterprise Linux hosts	Red Hat Virtualization Hosts Red Hat Enterprise Linux hosts	Virtual machine migration using libvirt .	Yes

ID	Port(s)	Protocol	Source	Destination	Purpose	Encrypted by default
H9	49152 - 49215	TCP	Red Hat Virtualization Hosts Red Hat Enterprise Linux hosts	Red Hat Virtualization Hosts Red Hat Enterprise Linux hosts	Virtual machine migration and fencing using VDSM. These ports must be open to facilitate both automated and manual migration of virtual machines.	Yes. Depending on agent for fencing, migration is done through libvirt.
H10	54321	TCP	Red Hat Virtualization Manager Red Hat Virtualization Hosts Red Hat Enterprise Linux hosts	Red Hat Virtualization Hosts Red Hat Enterprise Linux hosts	VDSM communications with the Manager and other virtualization hosts.	Yes
H11	54322	TCP	Red Hat Virtualization Manager ovirt-imageio service	Red Hat Virtualization Hosts Red Hat Enterprise Linux hosts	Required for communication with the ovirt-imageio service.	Yes
H12	6081	UDP	Red Hat Virtualization Hosts Red Hat Enterprise Linux hosts	Red Hat Virtualization Hosts Red Hat Enterprise Linux hosts	Required, when Open Virtual Network (OVN) is used as a network provider, to allow OVN to create tunnels between hosts.	No

ID	Port(s)	Protocol	Source	Destination	Purpose	Encrypted by default
H13	53	TCP, UDP	Red Hat Virtualization Hosts Red Hat Enterprise Linux hosts	DNS Server	DNS lookup requests from ports above 1023 to port 53, and responses. This port is required and open by default.	No
H14	123	UDP	Red Hat Virtualization Hosts Red Hat Enterprise Linux hosts	NTP Server	NTP requests from ports above 1023 to port 123, and responses. This port is required and open by default.	
H15	4500	TCP, UDP	Red Hat Virtualization Hosts	Red Hat Virtualization Hosts	Internet Security Protocol (IPSec)	Yes
H16	500	UDP	Red Hat Virtualization Hosts	Red Hat Virtualization Hosts	Internet Security Protocol (IPSec)	Yes
H17	-	AH, ESP	Red Hat Virtualization Hosts	Red Hat Virtualization Hosts	Internet Security Protocol (IPSec)	Yes



NOTE

By default, Red Hat Enterprise Linux allows outbound traffic to DNS and NTP on any destination address. If you disable outgoing traffic, make exceptions for the Red Hat Virtualization Hosts

Red Hat Enterprise Linux hosts to send requests to DNS and NTP servers. Other nodes may also require DNS and NTP. In that case, consult the requirements for those nodes and configure the firewall accordingly.

2.3.6. Database Server Firewall Requirements

Red Hat Virtualization supports the use of a remote database server for the Manager database (**engine**) and the Data Warehouse database (**ovirt-engine-history**). If you plan to use a remote database server, it must allow connections from the Manager and the Data Warehouse service (which can be separate from the Manager).

Similarly, if you plan to access a local or remote Data Warehouse database from an external system, the database must allow connections from that system.



IMPORTANT

Accessing the Manager database from external systems is not supported.



NOTE

A diagram of these firewall requirements is available at <https://access.redhat.com/articles/3932211>. You can use the IDs in the table to look up connections in the diagram.

Table 2.5. Database Server Firewall Requirements

ID	Port(s)	Protocol	Source	Destination	Purpose	Encrypted by default
D1	5432	TCP, UDP	Red Hat Virtualization Manager Data Warehouse service	Manager (engine) database server Data Warehouse (ovirt-engine-history) database server	Default port for PostgreSQL database connections.	No, but can be enabled.
D2	5432	TCP, UDP	External systems	Data Warehouse (ovirt-engine-history) database server	Default port for PostgreSQL database connections.	Disabled by default. No, but can be enabled.

2.3.7. Maximum Transmission Unit Requirements

The recommended Maximum Transmission Units (MTU) setting for Hosts during deployment is 1500. It is possible to update this setting after the environment is set up to a different MTU. For more information on changing the MTU setting, see [How to change the Hosted Engine VM network MTU](#) .

CHAPTER 3. PREPARING STORAGE FOR RED HAT VIRTUALIZATION

You need to prepare storage to be used for storage domains in the new environment. A Red Hat Virtualization environment must have at least one data storage domain, but adding more is recommended.



WARNING

When installing or reinstalling the host's operating system, Red Hat strongly recommends that you first detach any existing non-OS storage that is attached to the host to avoid accidental initialization of these disks, and with that, potential data loss.

A data domain holds the virtual hard disks and OVF files of all the virtual machines and templates in a data center, and cannot be shared across data centers while active (but can be migrated between data centers). Data domains of multiple storage types can be added to the same data center, provided they are all shared, rather than local, domains.

You can use one of the following storage types:

- [NFS](#)
- [iSCSI](#)
- [Fibre Channel \(FCP\)](#)
- [Red Hat Gluster Storage](#)

Prerequisites

- Self-hosted engines must have an additional data domain with at least 74 GiB dedicated to the Manager virtual machine. The self-hosted engine installer creates this domain. Prepare the storage for this domain before installation.



WARNING

Extending or otherwise changing the self-hosted engine storage domain after deployment of the self-hosted engine is not supported. Any such change might prevent the self-hosted engine from booting.

- When using a block storage domain, either FCP or iSCSI, a single target LUN is the only supported setup for a self-hosted engine.

- If you use iSCSI storage, the self-hosted engine storage domain must use a dedicated iSCSI target. Any additional storage domains must use a different iSCSI target.
- It is strongly recommended to create additional data storage domains in the same data center as the self-hosted engine storage domain. If you deploy the self-hosted engine in a data center with only one active data storage domain, and that storage domain is corrupted, you cannot add new storage domains or remove the corrupted storage domain. You must redeploy the self-hosted engine.

3.1. PREPARING NFS STORAGE

Set up NFS shares on your file storage or remote server to serve as storage domains on Red Hat Enterprise Virtualization Host systems. After exporting the shares on the remote storage and configuring them in the Red Hat Virtualization Manager, the shares will be automatically imported on the Red Hat Virtualization hosts.

For information on setting up, configuring, mounting and exporting NFS, see [Managing file systems](#) for Red Hat Enterprise Linux 8.

Specific system user accounts and system user groups are required by Red Hat Virtualization so the Manager can store data in the storage domains represented by the exported directories. The following procedure sets the permissions for one directory. You must repeat the **chown** and **chmod** steps for all of the directories you intend to use as storage domains in Red Hat Virtualization.

Prerequisites

1. Install the NFS **utils** package.

```
# dnf install nfs-utils -y
```

2. To check the enabled versions:

```
# cat /proc/fs/nfsd/versions
```

3. Enable the following services:

```
# systemctl enable nfs-server  
# systemctl enable rpcbind
```

Procedure

1. Create the group **kvm**:

```
# groupadd kvm -g 36
```

2. Create the user **vdsm** in the group **kvm**:

```
# useradd vdsm -u 36 -g kvm
```

3. Create the **storage** directory and modify the access rights.

```
# mkdir /storage
# chmod 0755 /storage
# chown 36:36 /storage/
```

4. Add the **storage** directory to **/etc/exports** with the relevant permissions.

```
# vi /etc/exports
# cat /etc/exports
/storage *(rw)
```

5. Restart the following services:

```
# systemctl restart rpcbind
# systemctl restart nfs-server
```

6. To see which export are available for a specific IP address:

```
# exportfs
/nfs_server/srv
      10.46.11.3/24
/nfs_server <world>
```



NOTE

If changes in **/etc/exports** have been made after starting the services, the **exportfs -ra** command can be used to reload the changes. After performing all the above stages, the exports directory should be ready and can be tested on a different host to check that it is usable.

3.2. PREPARING ISCSI STORAGE

Red Hat Virtualization supports iSCSI storage, which is a storage domain created from a volume group made up of LUNs. Volume groups and LUNs cannot be attached to more than one storage domain at a time.

For information on setting up and configuring iSCSI storage, see [Configuring an iSCSI target](#) in *Managing storage devices* for Red Hat Enterprise Linux 8.



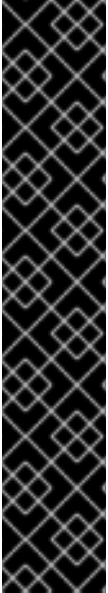
IMPORTANT

If you are using block storage and intend to deploy virtual machines on raw devices or direct LUNs and manage them with the Logical Volume Manager (LVM), you must create a filter to hide guest logical volumes. This will prevent guest logical volumes from being activated when the host is booted, a situation that could lead to stale logical volumes and cause data corruption. Use the **vdsm-tool config-lvm-filter** command to create filters for the LVM. See [Creating an LVM filter](#)



IMPORTANT

Red Hat Virtualization currently does not support block storage with a block size of 4K. You must configure block storage in legacy (512b block) mode.



IMPORTANT

If your host is booting from SAN storage and loses connectivity to the storage, the storage file systems become read-only and remain in this state after connectivity is restored.

To prevent this situation, add a drop-in multipath configuration file on the root file system of the SAN for the boot LUN to ensure that it is queued when there is a connection:

```
# cat /etc/multipath/conf.d/host.conf
multipaths {
  multipath {
    wwid boot_LUN_wwid
    no_path_retry queue
  }
}
```

3.3. PREPARING FCP STORAGE

Red Hat Virtualization supports SAN storage by creating a storage domain from a volume group made of pre-existing LUNs. Neither volume groups nor LUNs can be attached to more than one storage domain at a time.

Red Hat Virtualization 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 on setting up and configuring FCP or multipathing on Red Hat Enterprise Linux, see the [Storage Administration Guide](#) and [DM Multipath Guide](#).



IMPORTANT

If you are using block storage and intend to deploy virtual machines on raw devices or direct LUNs and manage them with the Logical Volume Manager (LVM), you must create a filter to hide guest logical volumes. This will prevent guest logical volumes from being activated when the host is booted, a situation that could lead to stale logical volumes and cause data corruption. Use the **vdsm-tool config-lvm-filter** command to create filters for the LVM. See [Creating an LVM filter](#)



IMPORTANT

Red Hat Virtualization currently does not support block storage with a block size of 4K. You must configure block storage in legacy (512b block) mode.

IMPORTANT

If your host is booting from SAN storage and loses connectivity to the storage, the storage file systems become read-only and remain in this state after connectivity is restored.

To prevent this situation, add a drop-in multipath configuration file on the root file system of the SAN for the boot LUN to ensure that it is queued when there is a connection:

```
# cat /etc/multipath/conf.d/host.conf
multipaths {
  multipath {
    wwid boot_LUN_wwid
    no_path_retry queue
  }
}
```

3.4. PREPARING RED HAT GLUSTER STORAGE

For information on setting up and configuring Red Hat Gluster Storage, see the [Red Hat Gluster Storage Installation Guide](#).

For the Red Hat Gluster Storage versions that are supported with Red Hat Virtualization, see [Red Hat Gluster Storage Version Compatibility and Support](#).

3.5. CUSTOMIZING MULTIPATH CONFIGURATIONS FOR SAN VENDORS

If your RHV environment is configured to use multipath connections with SANs, you can customize the multipath configuration settings to meet requirements specified by your storage vendor. These customizations can override both the default settings and settings that are specified in **/etc/multipath.conf**.

To override the multipath settings, do not customize **/etc/multipath.conf**. Because VDSM owns **/etc/multipath.conf**, installing or upgrading VDSM or Red Hat Virtualization can overwrite this file including any customizations it contains. This overwriting can cause severe storage failures.

Instead, you create a file in the **/etc/multipath/conf.d** directory that contains the settings you want to customize or override.

VDSM executes the files in **/etc/multipath/conf.d** in alphabetical order. So, to control the order of execution, you begin the filename with a number that makes it come last. For example, **/etc/multipath/conf.d/90-myfile.conf**.

To avoid causing severe storage failures, follow these guidelines:

- Do not modify **/etc/multipath.conf**. If the file contains user modifications, and the file is overwritten, it can cause unexpected storage problems.
- Do not override the **user_friendly_names** and **find_multipaths** settings. For details, see [Recommended Settings for Multipath.conf](#).
- Avoid overriding the **no_path_retry** and **polling_interval** settings unless a storage vendor specifically requires you to do so. For details, see [Recommended Settings for Multipath.conf](#).

**WARNING**

Not following these guidelines can cause catastrophic storage errors.

Prerequisites

- VDSM is configured to use the multipath module. To verify this, enter:

```
# vdsmd --is-configured --module multipath
```

Procedure

1. Create a new configuration file in the **/etc/multipath/conf.d** directory.
2. Copy the individual setting you want to override from **/etc/multipath.conf** to the new configuration file in **/etc/multipath/conf.d/<my_device>.conf**. Remove any comment marks, edit the setting values, and save your changes.
3. Apply the new configuration settings by entering:

```
# systemctl reload multipathd
```

**NOTE**

Do not restart the multipathd service. Doing so generates errors in the VDSM logs.

Verification steps

1. Test that the new configuration performs as expected on a non-production cluster in a variety of failure scenarios. For example, disable all of the storage connections.
2. Enable one connection at a time and verify that doing so makes the storage domain reachable.

Additional resources

- [Recommended Settings for Multipath.conf](#)
- [Red Hat Enterprise Linux DM Multipath](#)
- [Configuring iSCSI Multipathing](#)
- [How do I customize /etc/multipath.conf on my RHVH hypervisors? What values must not change and why?](#)

3.6. RECOMMENDED SETTINGS FOR MULTIPATH.CONF

Do not override the following settings:

user_friendly_names no

Device names must be consistent across all hypervisors. For example, `/dev/mapper/{WWID}`. The default value of this setting, **no**, prevents the assignment of arbitrary and inconsistent device names such as `/dev/mapper/mpath{N}` on various hypervisors, which can lead to unpredictable system behavior.

**WARNING**

Do not change this setting to **user_friendly_names yes**. User-friendly names are likely to cause unpredictable system behavior or failures, and are not supported.

find_multipaths no

This setting controls whether RHVH tries to access devices through multipath only if more than one path is available. The current value, **no**, allows RHV to access devices through multipath even if only one path is available.

**WARNING**

Do not override this setting.

Avoid overriding the following settings unless required by the storage system vendor:

no_path_retry 4

This setting controls the number of polling attempts to retry when no paths are available. Before RHV version 4.2, the value of **no_path_retry** was **fail** because QEMU had trouble with the I/O queuing when no paths were available. The **fail** value made it fail quickly and paused the virtual machine. RHV version 4.2 changed this value to **4** so when multipathd detects the last path has failed, it checks all of the paths four more times. Assuming the default 5-second polling interval, checking the paths takes 20 seconds. If no path is up, multipathd tells the kernel to stop queuing and fails all outstanding and future I/O until a path is restored. When a path is restored, the 20-second delay is reset for the next time all paths fail. For more details, see [the commit that changed this setting](#).

polling_interval 5

This setting determines the number of seconds between polling attempts to detect whether a path is open or has failed. Unless the vendor provides a clear reason for increasing the value, keep the VDSM-generated default so the system responds to path failures sooner.

CHAPTER 4. INSTALLING THE SELF-HOSTED ENGINE DEPLOYMENT HOST

A self-hosted engine can be deployed from a [Red Hat Virtualization Host](#) or a [Red Hat Enterprise Linux host](#).



IMPORTANT

If you plan to use bonded interfaces for high availability or VLANs to separate different types of traffic (for example, for storage or management connections), you should configure them on the host before beginning the self-hosted engine deployment. See [Networking Recommendations](#) in the *Planning and Prerequisites Guide*.

4.1. INSTALLING RED HAT VIRTUALIZATION HOSTS

Red Hat Virtualization Host (RHVH) is a minimal operating system based on Red Hat Enterprise Linux that is designed to provide a simple method for setting up a physical machine to act as a hypervisor in a Red Hat Virtualization environment. The minimal operating system contains only the packages required for the machine to act as a hypervisor, and features a Cockpit web interface for monitoring the host and performing administrative tasks. See [Running Cockpit](#) for the minimum browser requirements.

RHVH supports NIST 800-53 partitioning requirements to improve security. RHVH uses a NIST 800-53 partition layout by default.

The host must meet the minimum [host requirements](#).



WARNING

When installing or reinstalling the host's operating system, Red Hat strongly recommends that you first detach any existing non-OS storage that is attached to the host to avoid accidental initialization of these disks, and with that, potential data loss.

Procedure

1. Go to the [Get Started with Red Hat Virtualization](#) on the Red Hat Customer Portal and log in.
2. Click **Download Latest** to access the product download page.
3. Choose the appropriate **Hypervisor Image for RHV** from the list and click **Download Now**.
4. Start the machine on which you are installing RHVH, booting from the prepared installation media.
5. From the boot menu, select **Install RHVH 4.4** and press **Enter**.

**NOTE**

You can also press the **Tab** key to edit the kernel parameters. Kernel parameters must be separated by a space, and you can boot the system using the specified kernel parameters by pressing the **Enter** key. Press the **Esc** key to clear any changes to the kernel parameters and return to the boot menu.

6. Select a language, and click **Continue**.
7. Select a keyboard layout from the **Keyboard Layout** screen and click **Done**.
8. Select the device on which to install RHVH from the **Installation Destination** screen. Optionally, enable encryption. Click **Done**.

**IMPORTANT**

Use the **Automatically configure partitioning** option.

9. Select a time zone from the **Time & Date** screen and click **Done**.
10. Select a network from the **Network & Host Name** screen and click **Configure...** to configure the connection details.

**NOTE**

To use the connection every time the system boots, select the **Connect automatically with priority** check box. For more information, see [Configuring network and host name options](#) in the *Red Hat Enterprise Linux 8 Installation Guide*.

Enter a host name in the **Host Name** field, and click **Done**.

11. Optional: Configure **Security Policy** and **Kdump**. See [Customizing your RHEL installation using the GUI](#) in *Performing a standard RHEL installation* for Red Hat Enterprise Linux 8 for more information on each of the sections in the **Installation Summary** screen.
12. Click **Begin Installation**.
13. Set a root password and, optionally, create an additional user while RHVH installs.

**WARNING**

Do not create untrusted users on RHVH, as this can lead to exploitation of local security vulnerabilities.

14. Click **Reboot** to complete the installation.

**NOTE**

When RHVH restarts, **nodectl check** performs a health check on the host and displays the result when you log in on the command line. The message **node status: OK** or **node status: DEGRADED** indicates the health status. Run **nodectl check** to get more information.

**NOTE**

If necessary, you can [prevent kernel modules from loading automatically](#).

4.1.1. Enabling the Red Hat Virtualization Host Repository

Register the system to receive updates. Red Hat Virtualization Host only requires one repository. This section provides instructions for registering RHVH with the [Content Delivery Network](#), or with [Red Hat Satellite 6](#).

Registering RHVH with the Content Delivery Network

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

```
# subscription-manager register
```

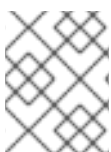
2. Enable the **Red Hat Virtualization Host 8** repository to allow later updates to the Red Hat Virtualization Host:

```
# subscription-manager repos --enable=rhvh-4-for-rhel-8-x86_64-rpms
```

Registering RHVH with Red Hat Satellite 6

1. Log in to the Cockpit web interface at **https://*HostFQDN*:9090**.
2. Click **Terminal**.
3. Register RHVH with Red Hat Satellite 6:

```
# rpm -Uvh http://satellite.example.com/pub/katello-ca-consumer-latest.noarch.rpm
# subscription-manager register --org="org_id"
# subscription-manager list --available
# subscription-manager attach --pool=pool_id
# subscription-manager repos \
--disable='*' \
--enable=rhvh-4-for-rhel-8-x86_64-rpms
```

**NOTE**

You can also configure virtual machine subscriptions in Red Hat Satellite using `virt-who`. See [Using virt-who to manage host-based subscriptions](#).

4.2. INSTALLING RED HAT ENTERPRISE LINUX HOSTS

A Red Hat Enterprise Linux host is based on a standard basic installation of Red Hat Enterprise Linux 8 on a physical server, with the **Red Hat Enterprise Linux Server** and **Red Hat Virtualization** subscriptions attached.

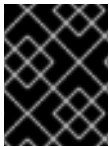
For detailed installation instructions, see the [Performing a standard RHEL installation](#).

The host must meet the minimum [host requirements](#).



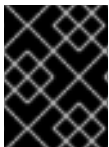
WARNING

When installing or reinstalling the host's operating system, Red Hat strongly recommends that you first detach any existing non-OS storage that is attached to the host to avoid accidental initialization of these disks, and with that, potential data loss.



IMPORTANT

Virtualization must be enabled in your host's BIOS settings. For information on changing your host's BIOS settings, refer to your host's hardware documentation.



IMPORTANT

Do not install third-party watchdogs on Red Hat Enterprise Linux hosts. They can interfere with the watchdog daemon provided by VDSM.

4.2.1. Enabling the Red Hat Enterprise Linux host Repositories

To use a Red Hat Enterprise Linux machine as a host, you must register the system with the Content Delivery Network, attach the **Red Hat Enterprise Linux Server** and **Red Hat Virtualization** subscriptions, and enable the host repositories.

Procedure

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

```
# subscription-manager register
```

2. Find the **Red Hat Enterprise Linux Server** and **Red Hat Virtualization** subscription pools and record the pool IDs:

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

3. Use the pool IDs to attach the subscriptions to the system:

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

**NOTE**

To view currently attached subscriptions:

```
# subscription-manager list --consumed
```

To list all enabled repositories:

```
# dnf repolist
```

4. Configure the repositories:

```
# subscription-manager repos \
  --disable='*' \
  --enable=rhel-8-for-x86_64-baseos-eus-rpms \
  --enable=rhel-8-for-x86_64-appstream-eus-rpms \
  --enable=rhv-4-mgmt-agent-for-rhel-8-x86_64-rpms \
  --enable=fast-datapath-for-rhel-8-x86_64-rpms \
  --enable=advanced-virt-for-rhel-8-x86_64-rpms \
  --enable=openstack-16.2-cinderlib-for-rhel-8-x86_64-rpms \
  --enable=rhceph-4-tools-for-rhel-8-x86_64-rpms \
  --enable=rhel-8-for-x86_64-appstream-tus-rpms \
  --enable=rhel-8-for-x86_64-baseos-tus-rpms
```

5. Set the RHEL version to 8.6:

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

6. Reset the **virt** module:

```
# dnf module reset virt
```

**NOTE**

If this module is already enabled in the Advanced Virtualization stream, this step is not necessary, but it has no negative impact.

You can see the value of the stream by entering:

```
# dnf module list virt
```

7. Enable the **virt** module in the Advanced Virtualization stream with the following command:

- For RHV 4.4.2:

```
# dnf module enable virt:8.2
```

- For RHV 4.4.3 to 4.4.5:

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


- For RHV 4.4.6 to 4.4.10:

```
# dnf module enable virt:av
```

- For RHV 4.4 and later:

```
# dnf module enable virt:rhel
```



NOTE

Starting with RHEL 8.6 the Advanced virtualization packages will use the standard **virt:rhel** module. For RHEL 8.4 and 8.5, only one Advanced Virtualization stream is used, **rhel:av**.

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

```
# dnf upgrade --nobest
```

2. Reboot the machine.



NOTE

If necessary, you can [prevent kernel modules from loading automatically](#).

CHAPTER 5. INSTALLING THE RED HAT VIRTUALIZATION MANAGER

5.1. MANUALLY INSTALLING THE RHV-M APPLIANCE

When you deploy the self-hosted engine, the following sequence of events takes place:

1. The installer installs the RHV-M Appliance to the deployment host.
2. The appliance installs the Manager virtual machine.
3. The appliance installs the Manager on the Manager virtual machine.

However, you can install the appliance manually on the deployment host beforehand if you need to. The appliance is large and network connectivity issues might cause the appliance installation to take a long time, or possibly fail.

Procedure

1. On Red Hat Enterprise Linux hosts:

- a. Reset the **virt** module:

```
# dnf module reset virt
```



NOTE

If this module is already enabled in the Advanced Virtualization stream, this step is not necessary, but it has no negative impact.

You can see the value of the stream by entering:

```
# dnf module list virt
```

- b. Enable the **virt** module in the Advanced Virtualization stream with the following command:

- For RHV 4.4.2:

```
# dnf module enable virt:8.2
```

- For RHV 4.4.3 to 4.4.5:

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

- For RHV 4.4.6 to 4.4.10:

```
# dnf module enable virt:av
```

- For RHV 4.4 and later:

```
# dnf module enable virt:rhel
```

**NOTE**

Starting with RHEL 8.6 the Advanced virtualization packages will use the standard **virt:rhel** module. For RHEL 8.4 and 8.5, only one Advanced Virtualization stream is used, **rhel:av**.

1. Synchronize installed packages to update them to the latest available versions:

```
# dnf distro-sync --nobest
```

2. Install the RHV-M Appliance to the host manually:

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

Now, when you deploy the self-hosted engine, the installer detects that the appliance is already installed.

5.2. ENABLING AND CONFIGURING THE FIREWALL

firewalld must be installed and running before you run the self-hosted deployment script. You must also have an active zone with an interface configured.

Prerequisites

- **firewalld** is installed. **hosted-engine-setup** requires the **firewalld** package, so you do not need to do any additional steps.

Procedure

1. Start **firewalld**:

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

To ensure **firewalld** starts automatically at system start, enter the following command as root:

```
# systemctl enable firewalld
```

2. Ensure that **firewalld** is running:

```
# systemctl status firewalld
```

3. Ensure that your management interface is in a firewall zone via

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

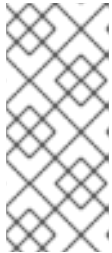
Now you are ready to deploy the self-hosted engine.

5.3. DEPLOYING THE SELF-HOSTED ENGINE USING THE COMMAND LINE

You can deploy a self-hosted engine from the command line. After installing the setup package, you run the command **hosted-engine --deploy**, and a script collects the details of your environment and uses them to configure the host and the Manager.

You can customize the Manager virtual machine during deployment, either manually, by pausing the deployment, or using automation.

- Setting the variable **he_pause_host** to **true** pauses deployment after installing the Manager and adding the deployment host to the Manager.
- Setting the variable **he_pause_before_engine_setup** to **true** pauses the deployment before installing the Manager and before restoring the Manager when using **he_restore_from_file**.



NOTE

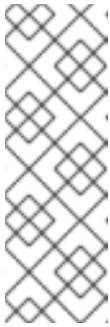
When the **he_pause_host** or **he_pause_before_engine_setup** variables are set to true a lock file is created at **/tmp** with the suffix **_he_setup_lock** on the deployment host. You can then manually customize the virtual machine as needed. The deployment continues after you delete the lock file, or after 24 hours, whichever comes first.

- Adding an Ansible playbook to any of the following directories on the deployment host automatically runs the playbook. Add the playbook under one of the following directories under **/usr/share/ansible/collections/ansible_collections/redhat/rhv/roles/hosted_engine_setup/hooks/**:
 - **enginevm_before_engine_setup**
 - **enginevm_after_engine_setup**
 - **after_add_host**
 - **after_setup**

Prerequisites

- Upgrade the appliance content to the latest product version before performing **engine-setup**.
 - To do this manually, pause the deployment using **he_pause_before_engine_setup** and perform a **dnf update**.
 - To do this automatically, apply the **enginevm_before_engine_setup** hook.
- FQDNs prepared for your Manager and the host. Forward and reverse lookup records must both be set in the DNS.
- When using a block storage domain, either FCP or iSCSI, a single target LUN is the only supported setup for a self-hosted engine.
- Optional: If you want to customize the Manager virtual machine during deployment using automation, an Ansible playbook must be added. See [Customizing the Engine virtual machine using automation during deployment](#).
- The self-hosted engine setup script requires ssh public key access using 2048-bit RSA keys from the engine virtual machine to the root account of its bare metal host. In **/etc/ssh/sshd_config**, these values must be set as follows:

- **PubkeyAcceptedKeyTypes** must allow 2048-bit RSA keys or stronger. By default, this setting uses system-wide crypto policies. For more information, see the manual page **crypto-policies(7)**.



NOTE

RHVH hosts that are registered with the Manager in versions earlier than 4.4.5.5 require RSA 2048 for backward compatibility until all the keys are migrated.

RHVH hosts registered for 4.4.5.5 and later use the strongest algorithm that is supported by both the Manager and RHVH. The **PubkeyAcceptedKeyTypes** setting helps determine which algorithm is used.

- **PermitRootLogin** is set to **without-password** or **yes**
- **PubkeyAuthentication** is set to **yes**

Procedure

1. Install the deployment tool:

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

2. Use the **tmux** window manager to run the script to avoid losing the session in case of network or terminal disruption.

Install and run **tmux**:

```
# dnf -y install tmux
# tmux
```

3. Start the deployment script:

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

Alternatively, to pause the deployment after adding the deployment host to the Manager, use the command line option **--ansible-extra-vars=he_pause_host=true**:

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



NOTE

To escape the script at any time, use the **Ctrl+D** keyboard combination to abort deployment. In the event of session timeout or connection disruption, run **tmux attach** to recover the deployment session.

4. When prompted, enter **Yes** to begin the deployment:

```
Continuing will configure this host for serving as hypervisor and will create a local VM with a running engine.
The locally running engine will be used to configure a new storage domain and create a VM
```

there.

At the end the disk of the local VM will be moved to the shared storage.

Are you sure you want to continue? (Yes, No)[Yes]:

5. Configure the network. Check that the gateway shown is correct and press **Enter**. Enter a pingable address on the same subnet so the script can check the host's connectivity.

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

6. The script detects possible NICs to use as a management bridge for the environment. Enter one of them or press **Enter** to accept the default.

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

7. Specify how to check network connectivity. The default is **dns**.

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

ping

Attempts to ping the gateway.

dns

Checks the connection to the DNS server.

tcp

Creates a TCP connection to a host and port combination. You need to specify a destination IP address and port. Once the connection is successfully created, the network is considered to be alive. Ensure that the given host is able to accept incoming TCP connections on the given port.

none

The network is always considered connected.

8. Enter a name for the data center in which to deploy the host for the self-hosted engine. The default name is **Default**.

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

9. Enter a name for the cluster in which to deploy the host for the self-hosted engine. The default name is **Default**.

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

10. If you want to use a custom appliance for the virtual machine installation, enter the path to the OVA archive. Otherwise, leave this field empty to use the RHV-M Appliance.

11. To deploy with a custom RHV-M Appliance appliance image, specify the path to the OVA archive. Otherwise, leave this field empty to use the RHV-M Appliance.

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

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

12. Enter the CPU and memory configuration for the Manager virtual machine:

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

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

13. Specify the FQDN for the Manager virtual machine, such as **manager.example.com**:

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

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

Engine VM FQDN []:

14. Specify the domain of the Manager virtual machine. For example, if the FQDN is **manager.example.com**, then enter **example.com**.

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

Engine VM domain: [example.com]

15. Create the root password for the Manager, and reenter it to confirm:

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

Confirm appliance root password:

16. Optional: Enter an SSH public key to enable you to log in to the Manager virtual machine as the root user without entering a password, and specify whether to enable SSH access for the root user:

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

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

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

SSH public key []:

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

17. Optional: You can apply the DISA STIG security profile on the Manager virtual machine. The DISA STIG profile is the default OpenSCAP profile.

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

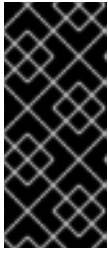
18. Enter a MAC address for the Manager virtual machine, or accept a randomly generated one. If you want to provide the Manager virtual machine with an IP address via DHCP, ensure that you have a valid DHCP reservation for this MAC address. The deployment script will not configure the DHCP server for you.

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

19. Enter the Manager virtual machine's networking details:

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

If you specified **Static**, enter the IP address of the Manager virtual machine:



IMPORTANT

- The static IP address must belong to the same subnet as the host. For example, if the host is in 10.1.1.0/24, the Manager virtual machine's IP must be in the same subnet range (10.1.1.1-254/24).
- For IPv6, Red Hat Virtualization supports only static addressing.

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

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

Engine VM DNS (leave it empty to skip):

20. Specify whether to add entries for the Manager virtual machine and the base host to the virtual machine's **/etc/hosts** file. You must ensure that the host names are resolvable.

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

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

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

21. Provide the name and TCP port number of the SMTP server, the email address used to send email notifications, and a comma-separated list of email addresses to receive these notifications. Alternatively, press **Enter** to accept the defaults:

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

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

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

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

22. Create a password for the **admin@internal** user to access the Administration Portal and reenter it to confirm:

Enter engine admin password:

Confirm engine admin password:

23. Specify the hostname of the deployment host:

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

The script creates the virtual machine. By default, the script first downloads and installs the RHV-M Appliance, which increases the installation time.

24. Optional: If you set the variable **he_pause_host: true**, the deployment pauses after adding the deployment host to the Manager. You can now log in from the deployment host to the Manager

virtual machine to customize it. You can log in with either the FQDN or the IP address of the Manager. For example, if the FQDN of the Manager is **manager.example.com**:

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

TIP

In the installation log, the IP address is in **local_vm_ip**. The installation log is the most recent instance of **/var/log/ovirt-hosted-engine-setup/ovirt-hosted-engine-setup-ansible-bootstrap_local_vm***.

- a. Customize the Manager virtual machine as needed.
- b. When you are done, log in to the Administration Portal using a browser with the Manager FQDN and make sure that the host's state is **Up**.
- c. Delete the lock file and the deployment script automatically continues, configuring the Manager virtual machine.

25. Select the type of storage to use:

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

- For NFS, enter the version, full address and path to the storage, and any mount options:

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

- For iSCSI, enter the portal details and select a target and LUN from the auto-detected lists. You can only select one iSCSI target during the deployment, but multipathing is supported to connect all portals of the same portal group.



NOTE

To specify more than one iSCSI target, you must enable multipathing before deploying the self-hosted engine. See [Red Hat Enterprise Linux DM Multipath](#) for details. There is also a [Multipath Helper](#) tool that generates a script to install and configure multipath with different options.

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

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

```
192.168.2.xxx:3260
192.168.3.xxx:3260
```

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

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

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

- For Gluster storage, enter the full address and path to the storage, and any mount options:



IMPORTANT

Only replica 1 and replica 3 Gluster storage are supported. Ensure you configure the volume as follows:

```
gluster volume set VOLUME_NAME group virt
gluster volume set VOLUME_NAME performance.strict-o-direct on
gluster volume set VOLUME_NAME network.remote-dio off
gluster volume set VOLUME_NAME storage.owner-uid 36
gluster volume set VOLUME_NAME storage.owner-gid 36
gluster volume set VOLUME_NAME network.ping-timeout 30
```

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

- For Fibre Channel, select a LUN from the auto-detected list. The host bus adapters must be configured and connected, and the LUN must not contain any existing data. To reuse an existing LUN, see [Reusing LUNs](#) in the *Administration Guide*.

The following luns have been found on the requested target:
 [1] 3514f0c5447600351 30GiB XtremIO XtremApp
 status: used, paths: 2 active

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

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

26. Enter the disk size of the Manager virtual machine:

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

When the deployment completes successfully, one data center, cluster, host, storage domain, and the Manager virtual machine are already running. You can log in to the Administration Portal to add any other resources.

27. Optional: Install and configure Red Hat Single Sign On so that you can add additional users to the environment. For more information, see [Installing and Configuring Red Hat Single Sign-On](#) in the *Administration Guide*.
28. Optional: Deploy Grafana so you can monitor and display reports from your RHV environment. For more information, see [Configuring Grafana](#) in the *Administration Guide*.

The Manager virtual machine, the host running it, and the self-hosted engine storage domain are flagged with a gold crown in the Administration Portal.



NOTE

Both the Manager's I/O scheduler and the hypervisor that hosts the Manager reorder I/O requests. This double reordering might delay I/O requests to the storage layer, impacting performance.

Depending on your data center, you might improve performance by changing the I/O scheduler to **none**. For more information, see [Available disk schedulers](#) in *Monitoring and managing system status and performance* for RHEL.

The next step is to enable the Red Hat Virtualization Manager repositories.

5.4. ENABLING THE RED HAT VIRTUALIZATION MANAGER REPOSITORIES

You need to log in and register the Manager machine with Red Hat Subscription Manager, attach the **Red Hat Virtualization Manager** subscription, and enable the Manager repositories.

Procedure

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

```
# subscription-manager register
```



NOTE

If you are using an IPv6 network, use an IPv6 transition mechanism to access the Content Delivery Network and subscription manager.

2. Find the **Red Hat Virtualization Manager** subscription pool and record the pool ID:

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

3. Use the pool ID to attach the subscription to the system:

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

**NOTE**

To view currently attached subscriptions:

```
# subscription-manager list --consumed
```

To list all enabled repositories:

```
# dnf repolist
```

4. Configure the repositories:

```
# subscription-manager repos \
  --disable='*' \
  --enable=rhel-8-for-x86_64-baseos-eus-rpms \
  --enable=rhel-8-for-x86_64-appstream-eus-rpms \
  --enable=rhv-4.4-manager-for-rhel-8-x86_64-rpms \
  --enable=fast-datapath-for-rhel-8-x86_64-rpms \
  --enable=jb-eap-7.4-for-rhel-8-x86_64-rpms \
  --enable=openstack-16.2-cinderlib-for-rhel-8-x86_64-rpms \
  --enable=rhceph-4-tools-for-rhel-8-x86_64-rpms \
  --enable=rhel-8-for-x86_64-appstream-tus-rpms \
  --enable=rhel-8-for-x86_64-baseos-tus-rpms
```

5. Set the RHEL version to 8.6:

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

6. Enable the **pki-deps** module.

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

7. Enable version 12 of the **postgresql** module.

```
# dnf module -y enable postgresql:12
```

8. Enable version 14 of the **nodejs** module:

```
# dnf module -y enable nodejs:14
```

9. Update the Self-Hosted Engine using the procedure [Updating a Self-Hosted Engine](#) in the *Upgrade Guide*.

Additional resources

For information on modules and module streams, see the following sections in *Installing, managing, and removing user-space components*

- [Module streams](#)
- [Selecting a stream before installation of packages](#)
- [Resetting module streams](#)

- [Switching to a later stream](#)

Log in to the Administration Portal, where you can add hosts and storage to the environment:

5.5. CONNECTING TO THE ADMINISTRATION PORTAL

Access the Administration Portal using a web browser.

1. In a web browser, navigate to **https://*manager-fqdn*/ovirt-engine**, replacing *manager-fqdn* with the FQDN that you provided during installation.



NOTE

You can access the Administration Portal using alternate host names or IP addresses. To do so, you need to add a configuration file under **/etc/ovirt-engine/engine.conf.d/**. For example:

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

The list of alternate host names needs to be separated by spaces. You can also add the IP address of the Manager to the list, but using IP addresses instead of DNS-resolvable host names is not recommended.

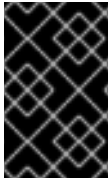
2. Click **Administration Portal**. An SSO login page displays. SSO login enables you to log in to the Administration and VM Portal at the same time.
3. Enter your **User Name** and **Password**. If you are logging in for the first time, use the user name **admin** along with the password that you specified during installation.
4. Select the **Domain** to authenticate against. If you are logging in using the internal **admin** user name, select the **internal** domain.
5. Click **Log In**.
6. You can view the Administration Portal in multiple languages. The default selection is chosen based on the locale settings of your web browser. If you want to view the Administration Portal in a language other than the default, select your preferred language from the drop-down list on the welcome page.

To log out of the Red Hat Virtualization Administration Portal, click your user name in the header bar and click **Sign Out**. You are logged out of all portals and the Manager welcome screen displays.

CHAPTER 6. INSTALLING HOSTS FOR RED HAT VIRTUALIZATION

Red Hat Virtualization supports two types of hosts: [Red Hat Virtualization Hosts \(RHVH\)](#) and [Red Hat Enterprise Linux hosts](#). Depending on your environment, you may want to use one type only, or both. At least two hosts are required for features such as migration and high availability.

See [Recommended practices for configuring host networks](#) for networking information.



IMPORTANT

SELinux is in enforcing mode upon installation. To verify, run **getenforce**. SELinux must be in enforcing mode on all hosts and Managers for your Red Hat Virtualization environment to be supported.

Table 6.1. Host Types

Host Type	Other Names	Description
Red Hat Virtualization Host	RHVH, thin host	This is a minimal operating system based on Red Hat Enterprise Linux. It is distributed as an ISO file from the Customer Portal and contains only the packages required for the machine to act as a host.
Red Hat Enterprise Linux host	RHEL host, thick host	Red Hat Enterprise Linux systems with the appropriate subscriptions attached can be used as hosts.

Host Compatibility

When you create a new data center, you can set the compatibility version. Select the compatibility version that suits all the hosts in the data center. Once set, version regression is not allowed. For a fresh Red Hat Virtualization installation, the latest compatibility version is set in the default data center and default cluster; to use an earlier compatibility version, you must create additional data centers and clusters. For more information about compatibility versions see *Red Hat Virtualization Manager Compatibility* in [Red Hat Virtualization Life Cycle](#).

6.1. RED HAT VIRTUALIZATION HOSTS

6.1.1. Installing Red Hat Virtualization Hosts

Red Hat Virtualization Host (RHVH) is a minimal operating system based on Red Hat Enterprise Linux that is designed to provide a simple method for setting up a physical machine to act as a hypervisor in a Red Hat Virtualization environment. The minimal operating system contains only the packages required for the machine to act as a hypervisor, and features a Cockpit web interface for monitoring the host and performing administrative tasks. See [Running Cockpit](#) for the minimum browser requirements.

RHVH supports NIST 800-53 partitioning requirements to improve security. RHVH uses a NIST 800-53 partition layout by default.

The host must meet the minimum [host requirements](#).



WARNING

When installing or reinstalling the host's operating system, Red Hat strongly recommends that you first detach any existing non-OS storage that is attached to the host to avoid accidental initialization of these disks, and with that, potential data loss.

Procedure

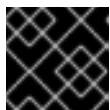
1. Go to the [Get Started with Red Hat Virtualization](#) on the Red Hat Customer Portal and log in.
2. Click **Download Latest** to access the product download page.
3. Choose the appropriate **Hypervisor Image for RHV** from the list and click **Download Now**.
4. Start the machine on which you are installing RHVH, booting from the prepared installation media.
5. From the boot menu, select **Install RHVH 4.4** and press **Enter**.



NOTE

You can also press the **Tab** key to edit the kernel parameters. Kernel parameters must be separated by a space, and you can boot the system using the specified kernel parameters by pressing the **Enter** key. Press the **Esc** key to clear any changes to the kernel parameters and return to the boot menu.

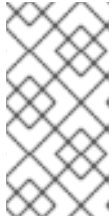
6. Select a language, and click **Continue**.
7. Select a keyboard layout from the **Keyboard Layout** screen and click **Done**.
8. Select the device on which to install RHVH from the **Installation Destination** screen. Optionally, enable encryption. Click **Done**.



IMPORTANT

Use the **Automatically configure partitioning** option.

9. Select a time zone from the **Time & Date** screen and click **Done**.
10. Select a network from the **Network & Host Name** screen and click **Configure...** to configure the connection details.

**NOTE**

To use the connection every time the system boots, select the **Connect automatically with priority** check box. For more information, see [Configuring network and host name options](#) in the *Red Hat Enterprise Linux 8 Installation Guide*.

Enter a host name in the **Host Name** field, and click **Done**.

11. Optional: Configure **Security Policy** and **Kdump**. See [Customizing your RHEL installation using the GUI](#) in *Performing a standard RHEL installation* for Red Hat Enterprise Linux 8 for more information on each of the sections in the **Installation Summary** screen.
12. Click **Begin Installation**.
13. Set a root password and, optionally, create an additional user while RHVH installs.

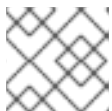
**WARNING**

Do not create untrusted users on RHVH, as this can lead to exploitation of local security vulnerabilities.

14. Click **Reboot** to complete the installation.

**NOTE**

When RHVH restarts, **nodectl check** performs a health check on the host and displays the result when you log in on the command line. The message **node status: OK** or **node status: DEGRADED** indicates the health status. Run **nodectl check** to get more information.

**NOTE**

If necessary, you can [prevent kernel modules from loading automatically](#).

6.1.2. Enabling the Red Hat Virtualization Host Repository

Register the system to receive updates. Red Hat Virtualization Host only requires one repository. This section provides instructions for registering RHVH with the [Content Delivery Network](#), or with [Red Hat Satellite 6](#).

Registering RHVH with the Content Delivery Network

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

```
# subscription-manager register
```


2. Enable the **Red Hat Virtualization Host 8** repository to allow later updates to the Red Hat Virtualization Host:

```
# subscription-manager repos --enable=rhvh-4-for-rhel-8-x86_64-rpms
```

Registering RHVH with Red Hat Satellite 6

1. Log in to the Cockpit web interface at **<https://HostFQDNorIP:9090>**.
2. Click **Terminal**.
3. Register RHVH with Red Hat Satellite 6:

```
# rpm -Uvh http://satellite.example.com/pub/katello-ca-consumer-latest.noarch.rpm
# subscription-manager register --org="org_id"
# subscription-manager list --available
# subscription-manager attach --pool=pool_id
# subscription-manager repos \
  --disable='*' \
  --enable=rhvh-4-for-rhel-8-x86_64-rpms
```



NOTE

You can also configure virtual machine subscriptions in Red Hat Satellite using virt-who. See [Using virt-who to manage host-based subscriptions](#) .

6.1.3. Advanced Installation

6.1.3.1. Custom Partitioning

Custom partitioning on Red Hat Virtualization Host (RHVH) is not recommended. Use the **Automatically configure partitioning** option in the **Installation Destination** window.

If your installation requires custom partitioning, select the **I will configure partitioning** option during the installation, and note that the following restrictions apply:

- Ensure the default **LVM Thin Provisioning** option is selected in the **Manual Partitioning** window.
- The following directories are required and must be on thin provisioned logical volumes:
 - root (/)
 - /home
 - /tmp
 - /var
 - /var/crash
 - /var/log
 - /var/log/audit



IMPORTANT

Do not create a separate partition for **/usr**. Doing so will cause the installation to fail.

/usr must be on a logical volume that is able to change versions along with RHVH, and therefore should be left on root (**/**).

For information about the required storage sizes for each partition, see [Storage Requirements](#).

- The **/boot** directory should be defined as a standard partition.
- The **/var** directory must be on a separate volume or disk.
- Only XFS or Ext4 file systems are supported.

Configuring Manual Partitioning in a Kickstart File

The following example demonstrates how to configure manual partitioning in a Kickstart file.

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



NOTE

If you use **logvol --thinpool --grow**, you must also include **volgroup --reserved-space** or **volgroup --reserved-percent** to reserve space in the volume group for the thin pool to grow.

6.1.3.2. Installing a DUD driver on a host without installer support

There are times when installing Red Hat Virtualization Host (RHVH) requires a Driver Update Disk (DUD), such as when using a hardware RAID device that is not supported by the default configuration of RHVH. In contrast with Red Hat Enterprise Linux hosts, RHVH does not fully support using a DUD. Subsequently the host fails to boot normally after installation because it does not see RAID. Instead it boots into emergency mode.

Example output:

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

In such a case you can manually add the drivers before finishing the installation.

Prerequisites

- A machine onto which you are installing RHVH.
- A DUD.
- If you are using a USB drive for the DUD and RHVH, you must have at least two available USB ports.

Procedure

1. Load the DUD on the host machine.
2. Install RHVH. See [Installing Red Hat Virtualization Hosts](#) in *Installing Red Hat Virtualization as a self-hosted engine using the command line*.



IMPORTANT

When installation completes, do not reboot the system.

TIP

If you want to access the DUD using SSH, do the following:

- Add the string **inst.sshd** to the kernel command line:

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

- Enable networking during the installation.

3. Enter the console mode, by pressing **Ctrl + Alt + F3**. Alternatively you can connect to it using SSH.
4. Mount the DUD:

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

5. Copy the RPM file inside the DUD to the target machine's disk:

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

For example:

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

6. Change the root directory to **/mnt/sysroot**:

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

7. Back up the current initrd images. For example:

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

8. Install the RPM file for the driver from the copy you made earlier.
For example:

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



NOTE

This package is not visible on the system after you reboot into the installed environment, so if you need it, for example, to rebuild the **initramfs**, you need to install that package once again, after which the package remains.

If you update the host using **dnf**, the driver update persists, so you do not need to repeat this process.

TIP

If you do not have an internet connection, use the **rpm** command instead of **dnf**:

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

9. Create a new image, forcefully adding the driver:

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

For example:

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

10. Check the results. The new image should be larger, and include the driver. For example, compare the sizes of the original, backed-up image file and the new image file.
In this example, the new image file is 88739013 bytes, larger than the original 88717417 bytes:

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

The new drivers should be part of the image file. For example, the 3w-9xxx module should be included:

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

- Copy the image to the the directory under **/boot** that contains the kernel to be used in the layer being installed, for example:

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

- Exit chroot.
- Exit the shell.
- If you used **Ctrl + Alt + F3** to access a virtual terminal, then move back to the installer by pressing **Ctrl + Alt + F<n>**, usually **F1** or **F5**
- At the installer screen, reboot.

Verification

The machine should reboot successfully.

6.1.3.3. Automating Red Hat Virtualization Host deployment

You can install Red Hat Virtualization Host (RHVH) without a physical media device by booting from a PXE server over the network with a Kickstart file that contains the answers to the installation questions.



WARNING

When installing or reinstalling the host's operating system, Red Hat strongly recommends that you first detach any existing non-OS storage that is attached to the host to avoid accidental initialization of these disks, and with that, potential data loss.

General instructions for installing from a PXE server with a Kickstart file are available in the [Red Hat Enterprise Linux Installation Guide](#), as RHVH is installed in much the same way as Red Hat Enterprise Linux. RHVH-specific instructions, with examples for deploying RHVH with Red Hat Satellite, are described below.

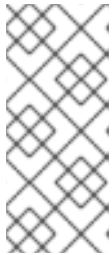
The automated RHVH deployment has 3 stages:

- [Preparing the Installation Environment](#)
- [Configuring the PXE Server and the Boot Loader](#)
- [Creating and Running a Kickstart File](#)

6.1.3.3.1. Preparing the installation environment

1. Go to the [Get Started with Red Hat Virtualization](#) on the Red Hat Customer Portal and log in.
2. Click **Download Latest** to access the product download page.
3. Choose the appropriate **Hypervisor Image for RHV** from the list and click **Download Now**.
4. Make the RHVH ISO image available over the network. See [Installation Source on a Network](#) in the *Red Hat Enterprise Linux Installation Guide*.
5. Extract the **squashfs.img** hypervisor image file from the RHVH ISO:

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



NOTE

This **squashfs.img** file, located in the **/tmp/usr/share/redhat-virtualization-host/image/** directory, is called **redhat-virtualization-host-version_number_version.squashfs.img**. It contains the hypervisor image for installation on the physical machine. It should not be confused with the **/LiveOS/squashfs.img** file, which is used by the Anaconda **inst.stage2** option.

6.1.3.3.2. Configuring the PXE server and the boot loader

1. Configure the PXE server. See [Preparing for a Network Installation](#) in the *Red Hat Enterprise Linux Installation Guide*.
2. Copy the RHVH boot images to the **/tftpboot** directory:

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

3. Create a **rhvh** label specifying the RHVH boot images in the boot loader configuration:

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

RHVH Boot loader configuration example for Red Hat Satellite

If you are using information from Red Hat Satellite to provision the host, you must create a global or host group level parameter called **rhvh_image** and populate it with the directory URL where the ISO is mounted or extracted:

```

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

DEFAULT rhvh

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

```

4. Make the content of the RHVH ISO locally available and export it to the network, for example, using an HTTPD server:

```

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

```

6.1.3.3.3. Creating and running a Kickstart file

1. Create a Kickstart file and make it available over the network. See [Kickstart Installations](#) in the *Red Hat Enterprise Linux Installation Guide*.
2. Ensure that the Kickstart file meets the following RHV-specific requirements:
 - The **%packages** section is not required for RHVH. Instead, use the **liveimg** option and specify the **redhat-virtualization-host-version_number_version.squashfs.img** file from the RHVH ISO image:

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

- Autopartitioning is highly recommended, but use caution: ensure that the local disk is detected first, include the **ignoredisk** command, and specify the local disk to ignore, such as **sda**. To ensure that a particular drive is used, Red Hat recommends using **ignoredisk --only-use=/dev/disk/<path>** or **ignoredisk --only-use=/dev/disk/<ID>**:

```

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

```



NOTE

Autopartitioning requires thin provisioning.

The **--no-home** option does not work in RHVH because **/home** is a required directory.

If your installation requires manual partitioning, see [Custom Partitioning](#) for a list of limitations that apply to partitions and an example of manual partitioning in a Kickstart file.

- A **%post** section that calls the **nodectl init** command is required:

```
%post
nodectl init
%end
```



NOTE

Ensure that the **nodectl init** command is at the very end of the **%post** section but before the reboot code, if any.

Kickstart example for deploying RHVH on its own

This Kickstart example shows you how to deploy RHVH. You can include additional commands and options as required.



WARNING

This example assumes that all disks are empty and can be initialized. If you have attached disks with data, either remove them or add them to the **ignoredisks** property.

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

reboot

%post --erroronfail
nodectl init
%end
```

Kickstart example for deploying RHVH with registration and network configuration from Satellite

This Kickstart example uses information from Red Hat Satellite to configure the host network and register the host to the Satellite server. You must create a global or host group level parameter called **rhvh_image** and populate it with the directory URL to the **squashfs.img** file. **ntp_server1** is also a global or host group level variable.

**WARNING**

This example assumes that all disks are empty and can be initialized. If you have attached disks with data, either remove them or add them to the **ignoredisks** property.

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

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

zerombr
clearpart --all
autopart --type=thinp

rootpw --iscrypted <%= root_pass %>

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

text
reboot

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

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

sync
systemctl reboot
%end
```

3. Add the Kickstart file location to the boot loader configuration file on the PXE server:

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

4. Install RHVH following the instructions in [Booting from the Network Using PXE](#) in the *Red Hat Enterprise Linux Installation Guide*.

6.2. RED HAT ENTERPRISE LINUX HOSTS

6.2.1. Installing Red Hat Enterprise Linux hosts

A Red Hat Enterprise Linux host is based on a standard basic installation of Red Hat Enterprise Linux 8 on a physical server, with the **Red Hat Enterprise Linux Server** and **Red Hat Virtualization** subscriptions attached.

For detailed installation instructions, see the [Performing a standard RHEL installation](#).

The host must meet the minimum [host requirements](#).



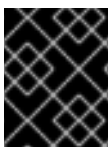
WARNING

When installing or reinstalling the host's operating system, Red Hat strongly recommends that you first detach any existing non-OS storage that is attached to the host to avoid accidental initialization of these disks, and with that, potential data loss.



IMPORTANT

Virtualization must be enabled in your host's BIOS settings. For information on changing your host's BIOS settings, refer to your host's hardware documentation.



IMPORTANT

Do not install third-party watchdogs on Red Hat Enterprise Linux hosts. They can interfere with the watchdog daemon provided by VDSM.

6.2.2. Enabling the Red Hat Enterprise Linux host Repositories

To use a Red Hat Enterprise Linux machine as a host, you must register the system with the Content Delivery Network, attach the **Red Hat Enterprise Linux Server** and **Red Hat Virtualization** subscriptions, and enable the host repositories.

Procedure

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

```
# subscription-manager register
```

- Find the **Red Hat Enterprise Linux Server** and **Red Hat Virtualization** subscription pools and record the pool IDs:

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

- Use the pool IDs to attach the subscriptions to the system:

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



NOTE

To view currently attached subscriptions:

```
# subscription-manager list --consumed
```

To list all enabled repositories:

```
# dnf repolist
```

- Configure the repositories:

```
# subscription-manager repos \
  --disable='*' \
  --enable=rhel-8-for-x86_64-baseos-eus-rpms \
  --enable=rhel-8-for-x86_64-appstream-eus-rpms \
  --enable=rhv-4-mgmt-agent-for-rhel-8-x86_64-rpms \
  --enable=fast-datapath-for-rhel-8-x86_64-rpms \
  --enable=advanced-virt-for-rhel-8-x86_64-rpms \
  --enable=openstack-16.2-cinderlib-for-rhel-8-x86_64-rpms \
  --enable=rhceph-4-tools-for-rhel-8-x86_64-rpms \
  --enable=rhel-8-for-x86_64-appstream-tus-rpms \
  --enable=rhel-8-for-x86_64-baseos-tus-rpms
```

- Set the RHEL version to 8.6:

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

- Ensure that all packages currently installed are up to date:

```
# dnf upgrade --nobest
```

- Reboot the machine.



NOTE

If necessary, you can [prevent kernel modules from loading automatically](#).

6.2.3. Installing Cockpit on Red Hat Enterprise Linux hosts

You can install Cockpit for monitoring the host's resources and performing administrative tasks.

Procedure

1. Install the dashboard packages:

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

2. Enable and start the **cockpit.socket** service:

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

3. Check if Cockpit is an active service in the firewall:

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

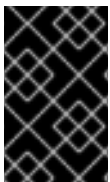
You should see **cockpit** listed. If it is not, enter the following with root permissions to add **cockpit** as a service to your firewall:

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

The **--permanent** option keeps the **cockpit** service active after rebooting.

You can log in to the Cockpit web interface at **https://*HostFQDN*or*IP*:9090**.

6.3. RECOMMENDED PRACTICES FOR CONFIGURING HOST NETWORKS



IMPORTANT

Always use the RHV Manager to modify the network configuration of hosts in your clusters. Otherwise, you might create an unsupported configuration. For details, see [Network Manager Stateful Configuration \(nmstate\)](#).

If your network environment is complex, you may need to configure a host network manually before adding the host to the Red Hat Virtualization Manager.

Consider the following practices for configuring a host network:

- Configure the network with Cockpit. Alternatively, you can use **nmtui** or **nmcli**.
- If a network is not required for a self-hosted engine deployment or for adding a host to the Manager, configure the network in the Administration Portal after adding the host to the Manager. See [Creating a New Logical Network in a Data Center or Cluster](#) .
- Use the following naming conventions:
 - VLAN devices: **VLAN_NAME_TYPE_RAW_PLUS_VID_NO_PAD**
 - VLAN interfaces: **physical_device.VLAN_ID** (for example, **eth0.23**, **eth1.128**, **enp3s0.50**)
 - Bond interfaces: **bondnumber** (for example, **bond0**, **bond1**)
 - VLANs on bond interfaces: **bondnumber.VLAN_ID** (for example, **bond0.50**, **bond1.128**)

- Use [network bonding](#). Network teaming is not supported in Red Hat Virtualization and will cause errors if the host is used to deploy a self-hosted engine or added to the Manager.
- Use recommended bonding modes:
 - If the **ovirtmgmt** network is not used by virtual machines, the network may use any supported bonding mode.
 - If the **ovirtmgmt** network is used by virtual machines, see [Which bonding modes work when used with a bridge that virtual machine guests or containers connect to?](#)
 - Red Hat Virtualization’s default bonding mode is **(Mode 4) Dynamic Link Aggregation**. If your switch does not support Link Aggregation Control Protocol (LACP), use **(Mode 1) Active-Backup**. See [Bonding Modes](#) for details.
- Configure a VLAN on a physical NIC as in the following example (although **nmcli** is used, you can use any tool):

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

- Configure a VLAN on a bond as in the following example (although **nmcli** is used, you can use any tool):

```
# nmcli connection add type bond con-name bond0 ifname bond0 bond.options
"mode=active-backup,miimon=100" ipv4.method disabled ipv6.method ignore
# nmcli connection add type ethernet con-name eth0 ifname eth0 master bond0 slave-type
bond
# nmcli connection add type ethernet con-name eth1 ifname eth1 master bond0 slave-type
bond
# nmcli connection add type vlan con-name vlan50 ifname bond0.50 dev bond0 id 50
# nmcli con mod vlan50 +ipv4.dns 8.8.8.8 +ipv4.addresses 123.123.0.1/24 +ipv4.gateway
123.123.0.254
```

- Do not disable **firewalld**.
- Customize the firewall rules in the Administration Portal after adding the host to the Manager. See [Configuring Host Firewall Rules](#).

6.4. ADDING SELF-HOSTED ENGINE NODES TO THE RED HAT VIRTUALIZATION MANAGER

Add self-hosted engine nodes in the same way as a standard host, with an additional step to deploy the host as a self-hosted engine node. The shared storage domain is automatically detected and the node can be used as a failover host to host the Manager virtual machine when required. You can also attach standard hosts to a self-hosted engine environment, but they cannot host the Manager virtual machine. Have at least two self-hosted engine nodes to ensure the Manager virtual machine is highly available. You can also add additional hosts using the REST API. See [Hosts](#) in the *REST API Guide*.

Prerequisites

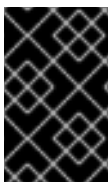
- All self-hosted engine nodes must be in the same cluster.

- If you are reusing a self-hosted engine node, remove its existing self-hosted engine configuration. See [Removing a Host from a Self-Hosted Engine Environment](#) .

Procedure

1. In the Administration Portal, click **Compute** → **Hosts**.
2. Click **New**.
For information on additional host settings, see [Explanation of Settings and Controls in the New Host and Edit Host Windows](#) in the *Administration Guide*.
3. Use the drop-down list to select the **Data Center** and **Host Cluster** for the new host.
4. Enter the **Name** and the **Address** of the new host. The standard SSH port, port 22, is auto-filled in the **SSH Port** field.
5. Select an authentication method to use for the Manager to access the host.
 - Enter the root user's password to use password authentication.
 - Alternatively, copy the key displayed in the **SSH PublicKey** field to `/root/.ssh/authorized_keys` on the host to use public key authentication.
6. Optionally, configure power management, where the host has a supported power management card. For information on power management configuration, see [Host Power Management Settings Explained](#) in the *Administration Guide*.
7. Click the **Hosted Engine** tab.
8. Select **Deploy**.
9. Click **OK**.

6.5. ADDING STANDARD HOSTS TO THE RED HAT VIRTUALIZATION MANAGER



IMPORTANT


Always use the RHV Manager to modify the network configuration of hosts in your clusters. Otherwise, you might create an unsupported configuration. For details, see [Network Manager Stateful Configuration \(nmstate\)](#) .

Adding a host to your Red Hat Virtualization environment can take some time, as the following steps are completed by the platform: virtualization checks, installation of packages, and creation of a bridge.

Procedure

1. From the Administration Portal, click **Compute** → **Hosts**.
2. Click **New**.
3. Use the drop-down list to select the **Data Center** and **Host Cluster** for the new host.
4. Enter the **Name** and the **Address** of the new host. The standard SSH port, port 22, is auto-filled in the **SSH Port** field.

5. Select an authentication method to use for the Manager to access the host.
 - Enter the root user's password to use password authentication.
 - Alternatively, copy the key displayed in the **SSH PublicKey** field to `/root/.ssh/authorized_keys` on the host to use public key authentication.
6. Optionally, click the **Advanced Parameters** button to change the following advanced host settings:
 - Disable automatic firewall configuration.
 - Add a host SSH fingerprint to increase security. You can add it manually, or fetch it automatically.
7. Optionally configure power management, where the host has a supported power management card. For information on power management configuration, see [Host Power Management Settings Explained](#) in the *Administration Guide*.
8. Click **OK**.

The new host displays in the list of hosts with a status of **Installing**, and you can view the progress of the installation in the **Events** section of the **Notification Drawer** (). After a brief delay the host status changes to **Up**.

CHAPTER 7. ADDING STORAGE FOR RED HAT VIRTUALIZATION

Add storage as data domains in the new environment. A Red Hat Virtualization environment must have at least one data domain, but adding more is recommended.

Add the storage you prepared earlier:

- [NFS](#)
- [iSCSI](#)
- [Fibre Channel \(FCP\)](#)
- [Red Hat Gluster Storage](#)



IMPORTANT

If you are using iSCSI storage, new data domains must not use the same iSCSI target as the self-hosted engine storage domain.



WARNING

Creating additional data domains in the same data center as the self-hosted engine storage domain is highly recommended. If you deploy the self-hosted engine in a data center with only one active data storage domain, and that storage domain is corrupted, you will not be able to add new storage domains or remove the corrupted storage domain; you will have to redeploy the self-hosted engine.

7.1. ADDING NFS STORAGE

This procedure shows you how to attach existing NFS storage to your Red Hat Virtualization environment as a data domain.

If you require an ISO or export domain, use this procedure, but select **ISO** or **Export** from the **Domain Function** list.

Procedure

1. In the Administration Portal, click **Storage** → **Domains**.
2. Click **New Domain**.
3. Enter a **Name** for the storage domain.
4. Accept the default values for the **Data Center**, **Domain Function**, **Storage Type**, **Format**, and **Host** lists.

5. Enter the **Export Path** to be used for the storage domain. The export path should be in the format of *123.123.0.10:/data* (for IPv4), *[2001:0:0:0:0:0:5db1]:/data* (for IPv6), or *domain.example.com:/data*.
6. Optionally, you can configure the advanced parameters:
 - a. Click **Advanced Parameters**.
 - b. Enter a percentage value into the **Warning Low Space Indicator** field. If the free space available on the storage domain is below this percentage, warning messages are displayed to the user and logged.
 - c. Enter a GB value into the **Critical Space Action Blocker** field. If the free space available on the storage domain is below this value, error messages are displayed to the user and logged, and any new action that consumes space, even temporarily, will be blocked.
 - d. Select the **Wipe After Delete** check box to enable the wipe after delete option. This option can be edited after the domain is created, but doing so will not change the wipe after delete property of disks that already exist.
7. Click **OK**.

The new NFS data domain has a status of **Locked** until the disk is prepared. The data domain is then automatically attached to the data center.

7.2. ADDING ISCSI STORAGE

This procedure shows you how to attach existing iSCSI storage to your Red Hat Virtualization environment as a data domain.

Procedure

1. Click **Storage → Domains**.
2. Click **New Domain**.
3. Enter the **Name** of the new storage domain.
4. Select a **Data Center** from the drop-down list.
5. Select **Data** as the **Domain Function** and **iSCSI** as the **Storage Type**.
6. Select an active host as the **Host**.



IMPORTANT

Communication to the storage domain is from the selected host and not directly from the Manager. Therefore, all hosts must have access to the storage device before the storage domain can be configured.

7. The Manager can map iSCSI targets to LUNs or LUNs to iSCSI targets. The **New Domain** window automatically displays known targets with unused LUNs when the iSCSI storage type is selected. If the target that you are using to add storage does not appear, you can use target discovery to find it; otherwise proceed to the next step.
 - a. Click **Discover Targets** to enable target discovery options. When targets have been

discovered and logged in to, the **New Domain** window automatically displays targets with LUNs unused by the environment.



NOTE

LUNs used externally for the environment are also displayed.

You can use the **Discover Targets** options to add LUNs on many targets or multiple paths to the same LUNs.



IMPORTANT

If you use the REST API method **discoveriscsi** to discover the iscsi targets, you can use an FQDN or an IP address, but you must use the iscsi details from the discovered targets results to log in using the REST API method **iscsilogin**. See [discoveriscsi](#) in the *REST API Guide* for more information.

- b. Enter the FQDN or IP address of the iSCSI host in the **Address** field.
- c. Enter the port with which to connect to the host when browsing for targets in the **Port** field. The default is **3260**.
- d. If CHAP is used to secure the storage, select the **User Authentication** check box. Enter the **CHAP user name** and **CHAP password**.



NOTE

You can define credentials for an iSCSI target for a specific host with the REST API. See [StorageServerConnectionExtensions: add](#) in the *REST API Guide* for more information.

- e. Click **Discover**.
- f. Select one or more targets from the discovery results and click **Login** for one target or **Login All** for multiple targets.



IMPORTANT

If more than one path access is required, you must discover and log in to the target through all the required paths. Modifying a storage domain to add additional paths is currently not supported.



IMPORTANT

When using the REST API **iscsilogin** method to log in, you must use the iscsi details from the discovered targets results in the **discoveriscsi** method. See [iscsilogin](#) in the *REST API Guide* for more information.

8. Click the **+** button next to the desired target. This expands the entry and displays all unused LUNs attached to the target.
9. Select the check box for each LUN that you are using to create the storage domain.

10. Optionally, you can configure the advanced parameters:
 - a. Click **Advanced Parameters**.
 - b. Enter a percentage value into the **Warning Low Space Indicator** field. If the free space available on the storage domain is below this percentage, warning messages are displayed to the user and logged.
 - c. Enter a GB value into the **Critical Space Action Blocker** field. If the free space available on the storage domain is below this value, error messages are displayed to the user and logged, and any new action that consumes space, even temporarily, will be blocked.
 - d. Select the **Wipe After Delete** check box to enable the wipe after delete option. This option can be edited after the domain is created, but doing so will not change the wipe after delete property of disks that already exist.
 - e. Select the **Discard After Delete** check box to enable the discard after delete option. This option can be edited after the domain is created. This option is only available to block storage domains.
11. Click **OK**.

If you have configured multiple storage connection paths to the same target, follow the procedure in [Configuring iSCSI Multipathing](#) to complete iSCSI bonding.

If you want to migrate your current storage network to an iSCSI bond, see [Migrating a Logical Network to an iSCSI Bond](#).

7.3. ADDING FCP STORAGE

This procedure shows you how to attach existing FCP storage to your Red Hat Virtualization environment as a data domain.

Procedure

1. Click **Storage → Domains**.
2. Click **New Domain**.
3. Enter the **Name** of the storage domain.
4. Select an FCP **Data Center** from the drop-down list.
If you do not yet have an appropriate FCP data center, select **(none)**.
5. Select the **Domain Function** and the **Storage Type** from the drop-down lists. The storage domain types that are not compatible with the chosen data center are not available.
6. Select an active host in the **Host** field. If this is not the first data domain in a data center, you must select the data center's SPM host.



IMPORTANT

All communication to the storage domain is through the selected host and not directly from the Red Hat Virtualization Manager. At least one active host must exist in the system and be attached to the chosen data center. All hosts must have access to the storage device before the storage domain can be configured.

7. The **New Domain** window automatically displays known targets with unused LUNs when **Fibre Channel** is selected as the storage type. Select the **LUN ID** check box to select all of the available LUNs.
8. Optionally, you can configure the advanced parameters.
 - a. Click **Advanced Parameters**.
 - b. Enter a percentage value into the **Warning Low Space Indicator** field. If the free space available on the storage domain is below this percentage, warning messages are displayed to the user and logged.
 - c. Enter a GB value into the **Critical Space Action Blocker** field. If the free space available on the storage domain is below this value, error messages are displayed to the user and logged, and any new action that consumes space, even temporarily, will be blocked.
 - d. Select the **Wipe After Delete** check box to enable the wipe after delete option. This option can be edited after the domain is created, but doing so will not change the wipe after delete property of disks that already exist.
 - e. Select the **Discard After Delete** check box to enable the discard after delete option. This option can be edited after the domain is created. This option is only available to block storage domains.
9. Click **OK**.

The new FCP data domain remains in a **Locked** status while it is being prepared for use. When ready, it is automatically attached to the data center.

7.4. ADDING RED HAT GLUSTER STORAGE

To use Red Hat Gluster Storage with Red Hat Virtualization, see [Configuring Red Hat Virtualization with Red Hat Gluster Storage](#).

For the Red Hat Gluster Storage versions that are supported with Red Hat Virtualization, see [Red Hat Gluster Storage Version Compatibility and Support](#).

APPENDIX A. TROUBLESHOOTING A SELF-HOSTED ENGINE DEPLOYMENT

To confirm whether the self-hosted engine has already been deployed, run **hosted-engine --check-deployed**. An error will only be displayed if the self-hosted engine has not been deployed.

A.1. TROUBLESHOOTING THE MANAGER VIRTUAL MACHINE

Check the status of the Manager virtual machine by running **hosted-engine --vm-status**.



NOTE

Any changes made to the Manager virtual machine will take about 20 seconds before they are reflected in the status command output.

Depending on the **Engine status** in the output, see the following suggestions to find or fix the issue.

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

1. If the Manager virtual machine is up and running as normal, you will see the following output:

```

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

```

2. If the output is normal but you cannot connect to the Manager, check the network connection.

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

1. If the **health** is **bad** and the **vm** is **up**, the HA services will try to restart the Manager virtual machine to get the Manager back. If it does not succeed within a few minutes, enable the global maintenance mode from the command line so that the hosts are no longer managed by the HA services.

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

2. Connect to the console. When prompted, enter the operating system's root password. For more console options, see [How to access Hosted Engine VM console from RHEV-H host?](#) .

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

3. Ensure that the Manager virtual machine's operating system is running by logging in.
4. Check the status of the **ovirt-engine** service:

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

5. Check the following logs: `/var/log/messages`, `/var/log/ovirt-engine/engine.log`, and `/var/log/ovirt-engine/server.log`.
6. After fixing the issue, reboot the Manager virtual machine manually from one of the self-hosted engine nodes:

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



NOTE

When the self-hosted engine nodes are in global maintenance mode, the Manager virtual machine must be rebooted manually. If you try to reboot the Manager virtual machine by sending a **reboot** command from the command line, the Manager virtual machine will remain powered off. This is by design.

7. On the Manager virtual machine, verify that the **ovirt-engine** service is up and running:

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

8. After ensuring the Manager virtual machine is up and running, close the console session and disable the maintenance mode to enable the HA services again:

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

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



NOTE

This message is expected on a host that is not currently running the Manager virtual machine.

1. If you have more than one host in your environment, ensure that another host is not currently trying to restart the Manager virtual machine.
2. Ensure that you are not in global maintenance mode.
3. Check the **ovirt-ha-agent** logs in `/var/log/ovirt-hosted-engine-ha/agent.log`.
4. Try to reboot the Manager virtual machine manually from one of the self-hosted engine nodes:

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

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

This status means that **ovirt-ha-agent** failed to get the virtual machine's details from VDSM.

1. Check the VDSM logs in `/var/log/vdsm/vdsm.log`.
2. Check the `ovirt-ha-agent` logs in `/var/log/ovirt-hosted-engine-ha/agent.log`.

Engine status: The self-hosted engine's configuration has not been retrieved from shared storage

If you receive the status **The hosted engine configuration has not been retrieved from shared storage. Please ensure that ovirt-ha-agent is running and the storage server is reachable** there is an issue with the `ovirt-ha-agent` service, or with the storage, or both.

1. Check the status of `ovirt-ha-agent` on the host:

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

2. If the `ovirt-ha-agent` is down, restart it:

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

3. Check the `ovirt-ha-agent` logs in `/var/log/ovirt-hosted-engine-ha/agent.log`.
4. Check that you can ping the shared storage.
5. Check whether the shared storage is mounted.

Additional Troubleshooting Commands



IMPORTANT

Contact the Red Hat Support Team if you feel you need to run any of these commands to troubleshoot your self-hosted engine environment.

- **hosted-engine --reinitialize-lockspace:** This command is used when the sanlock lockspace is broken. Ensure that the global maintenance mode is enabled and that the Manager virtual machine is stopped before reinitializing the sanlock lockspaces.
- **hosted-engine --clean-metadata:** Remove the metadata for a host's agent from the global status database. This makes all other hosts forget about this host. Ensure that the target host is down and that the global maintenance mode is enabled.
- **hosted-engine --check-liveliness:** This command checks the liveliness page of the ovirt-engine service. You can also check by connecting to <https://engine-fqdn/ovirt-engine/services/health/> in a web browser.
- **hosted-engine --connect-storage:** This command instructs VDSM to prepare all storage connections needed for the host and the Manager virtual machine. This is normally run in the back-end during the self-hosted engine deployment. Ensure that the global maintenance mode is enabled if you need to run this command to troubleshoot storage issues.

A.2. CLEANING UP A FAILED SELF-HOSTED ENGINE DEPLOYMENT

If a self-hosted engine deployment was interrupted, subsequent deployments will fail with an error message. The error will differ depending on the stage in which the deployment failed.

If you receive an error message, you can run the cleanup script on the deployment host to clean up the failed deployment. However, it's best to reinstall your base operating system and start the deployment from the beginning.



NOTE

The cleanup script has the following limitations:

- A disruption in the network connection while the script is running might cause the script to fail to remove the management bridge or to recreate a working network configuration.
- The script is not designed to clean up any shared storage device used during a failed deployment. You need to clean the shared storage device before you can reuse it in a subsequent deployment.

Procedure

1. Run **/usr/sbin/ovirt-hosted-engine-cleanup** and select **y** to remove anything left over from the failed self-hosted engine deployment.

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

2. Define whether to reinstall on the same shared storage device or select a different shared storage device.
 - To deploy the installation on the same storage domain, clean up the storage domain by running the following command in the appropriate directory on the server for NFS, Gluster, PosixFS or local storage domains:

```
# rm -rf storage_location/*
```
 - For iSCSI or Fibre Channel Protocol (FCP) storage, see [How to Clean Up a Failed Self-hosted Engine Deployment?](#) for information on how to clean up the storage.
 - Reboot the self-hosted engine host or select a different shared storage device.



NOTE

The reboot is needed to make sure all the connections to the storage are cleaned before the next attempt.

3. Redeploy the self-hosted engine.

APPENDIX B. CUSTOMIZING THE MANAGER VIRTUAL MACHINE USING AUTOMATION DURING DEPLOYMENT

You can use automation to adjust or otherwise customize the Manager virtual machine during deployment by using one or more Ansible playbooks. You can run playbooks at the following points during deployment:

- before the self-hosted engine setup
- after the self-hosted engine setup, but before storage is configured
- after adding the deployment host to the Manager
- after the deployment completes entirely

Procedure

1. Write one or more Ansible playbooks to run on the Manager virtual machine at specific points in the deployment process.
2. Add the playbooks to the appropriate directory under `/usr/share/ansible/collections/ansible_collections/redhat/rhv/roles/hosted_engine_setup/hooks/`:

enginevm_before_engine_setup

Run the playbook before the self-hosted engine setup.

enginevm_after_engine_setup

Run the playbook after the self-hosted engine setup, but before storage is configured.

after_add_host

Run the playbook after adding the deployment host to the Manager.

after_setup

Run the playbook after deployment is completed.

When you run the self-hosted-engine installer, the deployment script runs the **ovirt-engine-setup** role, which automatically runs any playbooks in either of these directories.

Additional resources

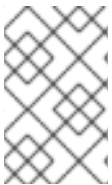
- [Deploying the self-hosted engine using the command line](#)
- [Automating Configuration Tasks using Ansible](#)
- [Intro to playbooks](#) in the Ansible documentation

APPENDIX C. MIGRATING DATABASES AND SERVICES TO A REMOTE SERVER

Although you cannot configure remote databases and services during the automated installation, you can migrate them to a separate server post-installation.

C.1. MIGRATING THE DATA WAREHOUSE TO A SEPARATE MACHINE

This section describes how to migrate the Data Warehouse database and service from the Red Hat Virtualization Manager machine to a separate machine. Hosting the Data Warehouse service on a separate machine reduces the load on each individual machine, and avoids potential conflicts caused by sharing CPU and memory resources with other processes.



NOTE

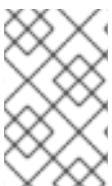
Red Hat only supports installing the Data Warehouse database, the Data Warehouse service and Grafana all on the same machine as each other, even though you can install each of these components on separate machines from each other.

You have the following migration options:

- You can migrate the Data Warehouse service away from the Manager machine and connect it with the existing Data Warehouse database (**ovirt_engine_history**).
- You can migrate the Data Warehouse database away from the Manager machine and then migrate the Data Warehouse service.

C.1.1. Migrating the Data Warehouse Database to a Separate Machine

Migrate the Data Warehouse database (**ovirt_engine_history**) before you migrate the Data Warehouse service. Use **engine-backup** to create a database backup and restore it on the new database machine. For more information on **engine-backup**, run **engine-backup --help**.



NOTE

Red Hat only supports installing the Data Warehouse database, the Data Warehouse service and Grafana all on the same machine as each other, even though you can install each of these components on separate machines from each other.

The new database server must have Red Hat Enterprise Linux 8 installed.

Enable the required repositories on the new database server.

C.1.1.1. Enabling the Red Hat Virtualization Manager Repositories

You need to log in and register the Data Warehouse machine with Red Hat Subscription Manager, attach the **Red Hat Virtualization Manager** subscription, and enable the Manager repositories.

Procedure

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

-

```
# subscription-manager register
```

**NOTE**

If you are using an IPv6 network, use an IPv6 transition mechanism to access the Content Delivery Network and subscription manager.

2. Find the **Red Hat Virtualization Manager** subscription pool and record the pool ID:

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

3. Use the pool ID to attach the subscription to the system:

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

**NOTE**

To view currently attached subscriptions:

```
# subscription-manager list --consumed
```

To list all enabled repositories:

```
# dnf repolist
```

4. Configure the repositories:

```
# subscription-manager repos \
  --disable='*' \
  --enable=rhel-8-for-x86_64-baseos-eus-rpms \
  --enable=rhel-8-for-x86_64-appstream-eus-rpms \
  --enable=rhv-4.4-manager-for-rhel-8-x86_64-rpms \
  --enable=fast-datapath-for-rhel-8-x86_64-rpms \
  --enable=jb-eap-7.4-for-rhel-8-x86_64-rpms \
  --enable=openstack-16.2-cinderlib-for-rhel-8-x86_64-rpms \
  --enable=rhceph-4-tools-for-rhel-8-x86_64-rpms \
  --enable=rhel-8-for-x86_64-appstream-tus-rpms \
  --enable=rhel-8-for-x86_64-baseos-tus-rpms
```

5. Set the RHEL version to 8.6:

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

6. Enable version 12 of the **postgresql** module.

```
# dnf module -y enable postgresql:12
```

7. Enable version 14 of the **nodejs** module:

```
# dnf module -y enable nodejs:14
```

- Update the Self-Hosted Engine using the procedure [Updating a Self-Hosted Engine](#) in the *Upgrade Guide*.

Additional resources

For information on modules and module streams, see the following sections in *Installing, managing, and removing user-space components*

- [Module streams](#)
- [Selecting a stream before installation of packages](#)
- [Resetting module streams](#)
- [Switching to a later stream](#)

C.1.1.2. Migrating the Data Warehouse Database to a Separate Machine

Procedure

- Create a backup of the Data Warehouse database and configuration files on the Manager:

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

- Copy the backup file from the Manager to the new machine:

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

- Install **engine-backup** on the new machine:

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

- Install the PostgreSQL server package:

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

- Initialize the PostgreSQL database, start the **postgresql** service, and ensure that this service starts on boot:

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

- Restore the Data Warehouse database on the new machine. *file_name* is the backup file copied from the Manager.

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

When the **--provision-*** option is used in restore mode, **--restore-permissions** is applied by default.

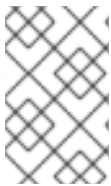
The Data Warehouse database is now hosted on a separate machine from that on which the Manager is hosted. After successfully restoring the Data Warehouse database, a prompt instructs you to run the **engine-setup** command. Before running this command, migrate the Data Warehouse service.

C.1.2. Migrating the Data Warehouse Service to a Separate Machine

You can migrate the Data Warehouse service installed and configured on the Red Hat Virtualization Manager to a separate machine. Hosting the Data Warehouse service on a separate machine helps to reduce the load on the Manager machine.

Notice that this procedure migrates the Data Warehouse service only.

To migrate the Data Warehouse database (**ovirt_engine_history**) prior to migrating the Data Warehouse service, see [Migrating the Data Warehouse Database to a Separate Machine](#) .



NOTE

Red Hat only supports installing the Data Warehouse database, the Data Warehouse service and Grafana all on the same machine as each other, even though you can install each of these components on separate machines from each other.

Prerequisites

- You must have installed and configured the Manager and Data Warehouse on the same machine.
- To set up the new Data Warehouse machine, you must have the following:
 - The password from the Manager's `/etc/ovirt-engine/engine.conf.d/10-setup-database.conf` file.
 - Allowed access from the Data Warehouse machine to the Manager database machine's TCP port 5432.
 - The username and password for the Data Warehouse database from the Manager's `/etc/ovirt-engine-dwh/ovirt-engine-dwhd.conf.d/10-setup-database.conf` file. If you migrated the **ovirt_engine_history** database using the procedures described in [Migrating the Data Warehouse Database to a Separate Machine](#) , the backup includes these credentials, which you defined during the database setup on that machine.

Installing this scenario requires four steps:

1. Setting up the New Data Warehouse Machine
2. Stopping the Data Warehouse service on the Manager machine
3. Configuring the new Data Warehouse machine
4. Disabling the Data Warehouse package on the Manager machine

C.1.2.1. Setting up the New Data Warehouse Machine

Enable the Red Hat Virtualization repositories and install the Data Warehouse setup package on a Red Hat Enterprise Linux 8 machine:

1. Enable the required repositories:

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

```
# subscription-manager register
```

- b. Find the **Red Hat Virtualization Manager** subscription pool and record the pool ID:

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

- c. Use the pool ID to attach the subscription to the system:

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

- d. Configure the repositories:

```
# subscription-manager repos \
  --disable='*' \
  --enable=rhel-8-for-x86_64-baseos-eus-rpms \
  --enable=rhel-8-for-x86_64-appstream-eus-rpms \
  --enable=rhv-4.4-manager-for-rhel-8-x86_64-rpms \
  --enable=fast-datapath-for-rhel-8-x86_64-rpms \
  --enable=jb-eap-7.4-for-rhel-8-x86_64-rpms

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

2. Enable the **pki-deps** module.

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

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

```
# dnf upgrade --nobest
```

4. Install the **ovirt-engine-dwh-setup** package:

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

C.1.2.2. Stopping the Data Warehouse Service on the Manager Machine

Procedure

1. Stop the Data Warehouse service:

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

2. If the database is hosted on a remote machine, you must manually grant access by editing the postgres.conf file. Edit the **/var/lib/pgsql/data/postgresql.conf** file and modify the listen_addresses line so that it matches the following:

```
listen_addresses = '*'
```

If the line does not exist or has been commented out, add it manually.

If the database is hosted on the Manager machine and was configured during a clean setup of the Red Hat Virtualization Manager, access is granted by default.

- Restart the postgresql service:

```
# systemctl restart postgresql
```

C.1.2.3. Configuring the New Data Warehouse Machine

The order of the options or settings shown in this section may differ depending on your environment.

- If you are migrating both the **ovirt_engine_history** database and the Data Warehouse service to the **same** machine, run the following, otherwise proceed to the next step.

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

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

- Remove the apache/grafana PKI files, so that they are regenerated by **engine-setup** with correct values:

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

- Run the **engine-setup** command to begin configuration of Data Warehouse on the machine:

```
# engine-setup
```

- Press **Enter** to accept the automatically detected host name, or enter an alternative host name and press **Enter**:

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

- Press **Enter** to automatically configure the firewall, or type **No** and press **Enter** to maintain existing settings:

```
Setup can automatically configure the firewall on this system.
Note: automatic configuration of the firewall may overwrite current settings.
Do you want Setup to configure the firewall? (Yes, No) [Yes]:
```

If you choose to automatically configure the firewall, and no firewall managers are active, you are prompted to select your chosen firewall manager from a list of supported options. Type the name of the firewall manager and press **Enter**. This applies even in cases where only one option is listed.

- Enter the fully qualified domain name and password for the Manager. Press **Enter** to accept the default values in each other field:

```
Host fully qualified DNS name of the engine server []: engine-fqdn
Setup needs to do some actions on the remote engine server. Either automatically, using ssh
as root to access it, or you will be prompted to manually perform each such action.
Please choose one of the following:
1 - Access remote engine server using ssh as root
2 - Perform each action manually, use files to copy content around
(1, 2) [1]:
ssh port on remote engine server [22]:
root password on remote engine server engine-fqdn: password
```

- Enter the FQDN and password for the Manager database machine. Press **Enter** to accept the default values in each other field:

```
Engine database host []: manager-db-fqdn
Engine database port [5432]:
Engine database secured connection (Yes, No) [No]:
Engine database name [engine]:
Engine database user [engine]:
Engine database password: password
```

- Confirm your installation settings:

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

The Data Warehouse service is now configured on the remote machine. Proceed to disable the Data Warehouse service on the Manager machine.

C.1.2.4. Disabling the Data Warehouse Service on the Manager Machine

Prerequisites

- The Grafana service on the Manager machine is disabled:

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

Procedure

- On the Manager machine, restart the Manager:

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

- Run the following command to modify the file `/etc/ovirt-engine-setup.conf.d/20-setup-ovirt-post.conf` and set the options to **False**:

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



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

3. Disable the Data Warehouse service:

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

4. Remove the Data Warehouse files:

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

The Data Warehouse service is now hosted on a separate machine from the Manager.

APPENDIX D. CONFIGURING A HOST FOR PCI PASSTHROUGH



NOTE

This is one in a series of topics that show how to set up and configure SR-IOV on Red Hat Virtualization. For more information, see [Setting Up and Configuring SR-IOV](#)

Enabling PCI passthrough allows a virtual machine to use a host device as if the device were directly attached to the virtual machine. To enable the PCI passthrough function, you must enable virtualization extensions and the IOMMU function. The following procedure requires you to reboot the host. If the host is attached to the Manager already, ensure you place the host into maintenance mode first.

Prerequisites

- Ensure that the host hardware meets the requirements for PCI device passthrough and assignment. See [PCI Device Requirements](#) for more information.

Configuring a Host for PCI Passthrough

1. Enable the virtualization extension and IOMMU extension in the BIOS. See [Enabling Intel VT-x and AMD-V virtualization hardware extensions in BIOS](#) in the *Red Hat Enterprise Linux Virtualization Deployment and Administration Guide* for more information.
2. Enable the IOMMU flag in the kernel by selecting the **Hostdev Passthrough & SR-IOV** check box when adding the host to the Manager or by editing the **grub** configuration file manually.
 - To enable the IOMMU flag from the Administration Portal, see [Adding Standard Hosts to the Red Hat Virtualization Manager](#) and [Kernel Settings Explained](#).
 - To edit the **grub** configuration file manually, see [Enabling IOMMU Manually](#).
3. For GPU passthrough, you need to run additional configuration steps on both the host and the guest system. See [GPU device passthrough: Assigning a host GPU to a single virtual machine](#) in *Setting up an NVIDIA GPU for a virtual machine in Red Hat Virtualization* for more information.

Enabling IOMMU Manually

1. Enable IOMMU by editing the grub configuration file.



NOTE

If you are using IBM POWER8 hardware, skip this step as IOMMU is enabled by default.

- For Intel, boot the machine, and append **intel_iommu=on** to the end of the **GRUB_CMDLINE_LINUX** line in the **grub** configuration file.

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

- For AMD, boot the machine, and append **amd_iommu=on** to the end of the **GRUB_CMDLINE_LINUX** line in the **grub** configuration file.

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

NOTE

If **intel_iommu=on** or an AMD IOMMU is detected, you can try adding **iommu=pt**. The **pt** option only enables IOMMU for devices used in passthrough and provides better host performance. However, the option might not be supported on all hardware. Revert to the previous option if the **pt** option doesn't work for your host.

If the passthrough fails because the hardware does not support interrupt remapping, you can consider enabling the **allow_unsafe_interrupts** option if the virtual machines are trusted. The **allow_unsafe_interrupts** is not enabled by default because enabling it potentially exposes the host to MSI attacks from virtual machines. To enable the option:

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

2. Refresh the **grub.cfg** file and reboot the host for these changes to take effect:

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

```
# reboot
```

APPENDIX E. PREVENTING KERNEL MODULES FROM LOADING AUTOMATICALLY

You can prevent a kernel module from being loaded automatically, whether the module is loaded directly, loaded as a dependency from another module, or during the boot process.

Procedure

1. The module name must be added to a configuration file for the **modprobe** utility. This file must reside in the configuration directory **/etc/modprobe.d**.
For more information on this configuration directory, see the man page **modprobe.d**.

2. Ensure the module is not configured to get loaded in any of the following:

- **/etc/modprobe.conf**
- **/etc/modprobe.d/***
- **/etc/rc.modules**
- **/etc/sysconfig/modules/***

```
# modprobe --showconfig <_configuration_file_name_>
```

3. If the module appears in the output, ensure it is ignored and not loaded:

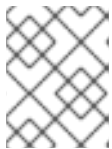
```
# modprobe --ignore-install <_module_name_>
```

4. Unload the module from the running system, if it is loaded:

```
# modprobe -r <_module_name_>
```

5. Prevent the module from being loaded directly by adding the **blacklist** line to a configuration file specific to the system - for example **/etc/modprobe.d/local-dontload.conf**:

```
# echo "blacklist <_module_name_>" >> /etc/modprobe.d/local-dontload.conf
```



NOTE

This step does not prevent a module from loading if it is a required or an optional dependency of another module.

6. Prevent optional modules from being loading on demand:

```
# echo "install <_module_name_>/bin/false" >> /etc/modprobe.d/local-dontload.conf
```



IMPORTANT

If the excluded module is required for other hardware, excluding it might cause unexpected side effects.

7. Make a backup copy of your **initramfs**:

```
# cp /boot/initramfs-$(uname -r).img /boot/initramfs-$(uname -r).img.$(date +%m-%d-%H%M%S).bak
```

8. If the kernel module is part of the **initramfs**, rebuild your initial **ramdisk** image, omitting the module:

```
# dracut --omit-drivers <_module_name_> -f
```

9. Get the current kernel command line parameters:

```
# grub2-editenv - list | grep kernelopts
```

10. Append **<_module_name_>.blacklist=1 rd.driver.blacklist=<_module_name_>** to the generated output:

```
# grub2-editenv - set kernelopts="<> <_module_name_>.blacklist=1 rd.driver.blacklist=<_module_name_>"
```

For example:

```
# grub2-editenv - set kernelopts="root=/dev/mapper/rhel_example-root ro crashkernel=auto resume=/dev/mapper/rhel_example-swap rd.lvm.lv=rhel_example/root rd.lvm.lv=rhel_example/swap <_module_name_>.blacklist=1 rd.driver.blacklist=<_module_name_>"
```

11. Make a backup copy of the **kdump initramfs**:

```
# cp /boot/initramfs-$(uname -r)kdump.img /boot/initramfs-$(uname -r)kdump.img.$(date +%m-%d-%H%M%S).bak
```

12. Append **rd.driver.blacklist=<_module_name_>** to the **KDUMP_COMMANDLINE_APPEND** setting in **/etc/sysconfig/kdump** to omit it from the **kdump initramfs**:

```
# sed -i '/^KDUMP_COMMANDLINE_APPEND=/s/"$/ rd.driver.blacklist=module_name"/ /etc/sysconfig/kdump
```

13. Restart the **kdump** service to pick up the changes to the **kdump initrd**:

```
# kdumpctl restart
```

14. Rebuild the **kdump** initial **ramdisk** image:

```
# mkdumprd -f /boot/initramfs-$(uname -r)kdump.img
```

15. Reboot the system.

E.1. REMOVING A MODULE TEMPORARILY

You can remove a module temporarily.

Procedure

1. Run **modprobe** to remove any currently-loaded module:

```
# modprobe -r <module name>
```

2. If the module cannot be unloaded, a process or another module might still be using the module. If so, terminate the process and run the **modprobe** command written above another time to unload the module.

APPENDIX F. SECURING RED HAT VIRTUALIZATION

This information is specific to Red Hat Virtualization. It does not cover fundamental security practices related to any of the following:

- Disabling unnecessary services
- Authentication
- Authorization
- Accounting
- Penetration testing and hardening of non-RHV services
- Encryption of sensitive application data

Prerequisites

- You should be proficient in your organization's security standards and practices. If possible, consult with your organization's Security Officer.
- Consult the Red Hat Enterprise Linux [Security hardening](#) before deploying RHEL hosts.

F.1. APPLYING THE DISA STIG PROFILE IN RHEL BASED HOSTS AND THE STANDALONE MANAGER

When installing RHV, you can select the DISA STIG profile with the UI installer, which is the profile provided by RHEL 8.



IMPORTANT

The DISA STIG profile is not supported for Red Hat Virtualization Host (RHVH).

Procedure

1. In the **Installation Summary** screen, select **Security Policy**.
2. In the **Security Policy** screen, set the **Apply security policy** to **On**.
3. Select **DISA STIG for Red Hat Enterprise Linux 8**
4. Click **Select profile**. This action adds a green checkmark next to the profile and adds packages to the list of **Changes that were done or need to be done** Follow the onscreen instructions if they direct you to make any changes.
5. Click **Done**.
6. On the **Installation Summary** screen, verify that the status of **Security Policy** is **Everything okay**.
7. Reboot the host.

F.1.1. Enabling DISA STIG in a self-hosted engine

You can enable DISA STIG in a self-hosted engine during deployment when using the command-line.

Procedure

1. Start the self-hosted engine deployment script. See [Installing Red Hat Virtualization as a self-hosted engine using the command line](#).
2. When the deployment script prompts **Do you want to apply an OpenSCAP security profile?**, enter **Yes**.
3. When the deployment script prompts **Please provide the security profile you would like to use?**, enter **stig**.

F.2. APPLYING THE PCI-DSS PROFILE IN RHV HOSTS AND THE STANDALONE MANAGER

When installing RHVH, you can select the PCI-DSS profile with the UI installer, which is the profile provided by RHEL 8.

Procedure

1. In the **Installation Summary** screen, select **Security Policy**.
2. In the **Security Policy** screen, set the **Apply security policy** to **On**.
3. Select **PCI-DSS v3.2.1 Control Baseline for Red Hat Enterprise Linux 8**
4. Click **Select profile**. This action adds a green checkmark next to the profile and adds packages to the list of **Changes that were done or need to be done**. Follow the onscreen instructions if they direct you to make any changes.
5. Click **Done**.
6. In the **Installation Summary** screen, verify that the status of **Security Policy** is **Everything okay**.
7. Reboot the host.

F.2.1. Enabling PCI-DSS in a self-hosted engine

You can enable PCI-DSS in a self-hosted engine during deployment when using the command-line.

Procedure

1. Start the self-hosted engine deployment script. See [Installing Red Hat Virtualization as a self-hosted engine using the command line](#).
2. When the deployment script prompts **Do you want to apply an OpenSCAP security profile?**, enter **Yes**.
3. When the deployment script prompts **Please provide the security profile you would like to use?**, enter **pci-dss**.

APPENDIX G. DEFINING ALLOWED CPU TYPES IN SELF-HOSTED ENGINE DEPLOYMENT

Procedure

1. Create a file named **deploy.json**, and from the table shown below, select a CPU type for the **he_cluster_cpu_type**. For example, if the CPU type you want is **Secure Intel Nehalem Family**, then the **deploy.json** should look like the following:

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

2. Provide the deploy.json file to the **hosted-engine --deploy** process.

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

Table G.1. Allowed CPU Types

CPU type name	CPU properties
Intel Nehalem Family	vmx,nx,model_Nehalem:Nehalem:x86_64
Secure Intel Nehalem Family	vmx,spec_ctrl,ssbd,model_Nehalem:Nehalem,+spec-ctrl,+ssbd:x86_64
Intel Westmere Family	aes,vmx,nx,model_Westmere:Westmere:x86_64
Secure Intel Westmere Family	aes,vmx,spec_ctrl,ssbd,model_Westmere:Westmere,+pcid,+spec-ctrl,+ssbd:x8_64
Intel SandyBridge Family	vmx,nx,model_SandyBridge:SandyBridge:x86_64
Secure Intel SandyBridge Family	vmx,spec_ctrl,ssbd,md_clear,model_SandyBridge:SandyBridge,+pcid,+spec-ctrl,+ssbd,+md-clear:x86_64
Intel IvyBridge Family	vmx,nx,model_IvyBridge:IvyBridge:x86_64
Secure Intel IvyBridge Family	vmx,spec_ctrl,ssbd,md_clear,model_IvyBridge:IvyBridge,+pcid,+spec-ctrl,+ssbd,+md-clear:x86_64
Intel Haswell Family	vmx,nx,model_Haswell-noTSX:Haswell-noTSX:x86_64
Secure Intel Haswell Family	vmx,spec_ctrl,ssbd,md_clear,model_Haswell-noTSX:Haswell-noTSX,+spec-ctrl,+ssbd,+md-clear:x86_64

CPU type name	CPU properties
Intel Broadwell Family	vmx,nx,model_Broadwell-noTSX:Broadwell-noTSX:x86_64
Secure Intel Broadwell Family	vmx,spec_ctrl,ssbd,md_clear,model_Broadwell-noTSX:Broadwell-noTSX,+spec-ctrl,+ssbd,+md-clear:x86_64
Intel Skylake Client Family	vmx,nx,model_Skylake-Client:Skylake-Client,-hle,-rtm,-mpx:x86_64
Secure Intel Skylake Client Family	vmx,ssbd,md_clear,model_Skylake-Client-noTSX-IBRS:Skylake-Client-noTSX-IBRS,+ssbd,+md-clear,-mpx:x86_64
Intel Skylake Server Family	vmx,nx,model_Skylake-Server:Skylake-Server,-hle,-rtm,-mpx:x86_64
Secure Intel Skylake Server Family	vmx,ssbd,md_clear,model_Skylake-Server-noTSX-IBRS:Skylake-Server-noTSX-IBRS,+ssbd,+md-clear,-mpx:x86_64
Intel Cascadelake Server Family	vmx,model_Cascadelake-Server:Cascadelake-Server,-hle,-rtm,-mpx:x86_64
Secure Intel Cascadelake Server Family	vmx,model_Cascadelake-Server-noTSX:Cascadelake-Server-noTSX,-mpx:x86_64
Intel Icelake Server Family	vmx,model_Icelake-Server-noTSX:Icelake-Server-noTSX,-mpx:x86_64
Secure Intel Icelake Server Family	vmx,arch-capabilities,rdctl-no,ibrs-all,skip-lldfl-vmentry,mds-no,pschange-mc-no,taa-no,model_Icelake-Server-noTSX:Icelake-Server-noTSX,+arch-capabilities,+rdctl-no,+ibrs-all,+skip-lldfl-vmentry,+mds-no,+pschange-mc-no,+taa-no,-mpx:x86_64
AMD Opteron G4	svm,nx,model_Opteron_G4:Opteron_G4:x86_64
AMD Opteron G5	svm,nx,model_Opteron_G5:Opteron_G5:x86_64
AMD EPYC	svm,nx,model_EPYC:EPYC:x86_64
Secure AMD EPYC	svm,nx,ibpb,ssbd,model_EPYC:EPYC,+ibpb,+virt-ssbd:x86_64

CPU type name	CPU properties
IBM POWER8	powernv,model_POWER8:POWER8:ppc64
IBM POWER9	powernv,model_POWER9:POWER9:ppc64
IBM z114, z196	sie,model_z196-base:z196-base:s390x
IBM zBC12, zEC12	sie,model_zEC12-base:zEC12-base:s390x
IBM z13s, z13	sie,model_z13-base:z13-base:s390x
IBM z14	sie,model_z14-base:z14-base:s390x

APPENDIX H. LEGAL NOTICE

Copyright © 2022 Red Hat, Inc.

Licensed under the ([Creative Commons Attribution–ShareAlike 4.0 International License](#)). Derived from documentation for the ([oVirt Project](#)). If you distribute this document or an adaptation of it, you must provide the URL for the original version.

Modified versions must remove all Red Hat trademarks.

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

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

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

XFS® is a trademark of Silicon Graphics International Corp. or its subsidiaries in the United States and/or other countries.

MySQL® is a registered trademark of MySQL AB in the United States, the European Union and other countries.

Node.js® is an official trademark of Joyent. Red Hat Software Collections is not formally related to or endorsed by the official Joyent Node.js open source or commercial project.

The OpenStack® Word Mark and OpenStack logo are either registered trademarks/service marks or trademarks/service marks of the OpenStack Foundation, in the United States and other countries and are used with the OpenStack Foundation's permission. We are not affiliated with, endorsed or sponsored by the OpenStack Foundation, or the OpenStack community.

All other trademarks are the property of their respective owners.