



OpenShift Container Platform 4.6

Installing on RHV

Installing OpenShift Container Platform RHV clusters

OpenShift Container Platform 4.6 Installing on RHV

Installing OpenShift Container Platform RHV clusters

Legal Notice

Copyright © 2020 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 provides instructions for installing and uninstalling OpenShift Container Platform clusters on Red Hat Virtualization.

Table of Contents

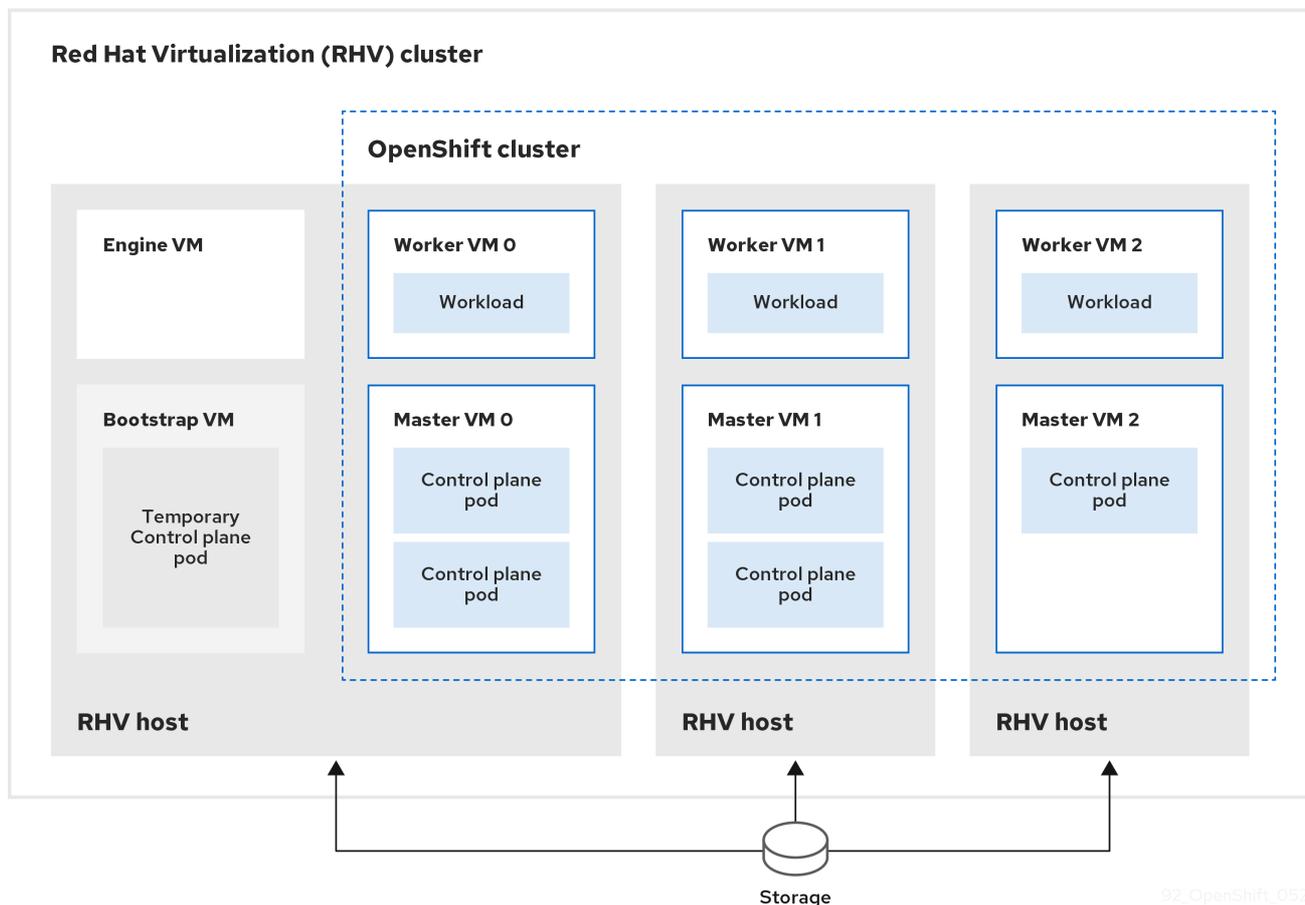
CHAPTER 1. INSTALLING ON RHV	4
1.1. INSTALLING A CLUSTER QUICKLY ON RHV	4
1.1.1. Prerequisites	4
1.1.2. Internet and Telemetry access for OpenShift Container Platform	4
1.1.3. Requirements for the RHV environment	5
1.1.4. Verifying the requirements for the RHV environment	6
1.1.5. Preparing the network environment on RHV	8
1.1.6. Setting up the CA certificate for RHV	9
1.1.7. Generating an SSH private key and adding it to the agent	10
1.1.8. Obtaining the installation program	11
1.1.9. Deploying the cluster	12
CHAPTER 2. INSTALLING THE CLI BY DOWNLOADING THE BINARY	15
2.1. INSTALLING THE CLI ON LINUX	15
2.2. INSTALLING THE CLI ON WINDOWS	15
2.3. INSTALLING THE CLI ON MACOS	16
CHAPTER 3. LOGGING IN TO THE CLUSTER	17
3.1. VERIFYING CLUSTER STATUS	17
3.2. ACCESSING THE OPENSIFT CONTAINER PLATFORM WEB CONSOLE ON RHV	18
3.3. TROUBLESHOOTING COMMON ISSUES WITH INSTALLING ON RED HAT VIRTUALIZATION (RHV)	18
3.3.1. CPU load increases and nodes go into a Not Ready state	18
3.3.2. Trouble connecting the OpenShift Container Platform cluster API	19
3.4. POST-INSTALLATION TASKS	19
3.5. INSTALLING A CLUSTER ON RHV WITH CUSTOMIZATIONS	19
3.5.1. Prerequisites	20
3.5.2. Internet and Telemetry access for OpenShift Container Platform	21
3.5.3. Requirements for the RHV environment	21
3.5.4. Verifying the requirements for the RHV environment	23
3.5.5. Preparing the network environment on RHV	24
3.5.6. Setting up the CA certificate for RHV	25
3.5.7. Generating an SSH private key and adding it to the agent	26
3.5.8. Obtaining the installation program	27
3.5.9. Creating the installation configuration file	28
3.5.9.1. Example install-config.yaml files for Red Hat Virtualization (RHV)	30
3.5.9.2. Installation configuration parameters	32
3.5.10. Deploying the cluster	40
3.5.11. Installing the CLI by downloading the binary	41
3.5.11.1. Installing the CLI on Linux	41
3.5.11.2. Installing the CLI on Windows	42
3.5.11.3. Installing the CLI on macOS	42
3.5.12. Logging in to the cluster	43
3.5.13. Verifying cluster status	43
3.5.14. Accessing the OpenShift Container Platform web console on RHV	44
3.5.15. Troubleshooting common issues with installing on Red Hat Virtualization (RHV)	44
3.5.15.1. CPU load increases and nodes go into a Not Ready state	45
3.5.15.2. Trouble connecting the OpenShift Container Platform cluster API	45
3.5.16. Post-installation tasks	45
3.5.17. Next steps	46
3.6. INSTALLING A CLUSTER ON RHV WITH USER-PROVISIONED INFRASTRUCTURE	46
3.6.1. Prerequisites	46

3.6.2. Internet and Telemetry access for OpenShift Container Platform	47
3.6.3. Requirements for the RHV environment	47
3.6.4. Verifying the requirements for the RHV environment	49
3.6.5. Network infrastructure configuration for installing OpenShift Container Platform on Red Hat Virtualization (RHV)	50
3.6.6. Setting up the installation machine	51
3.6.7. Setting up the CA certificate for RHV	52
3.6.8. Generating an SSH private key and adding it to the agent	53
3.6.9. Obtaining the installation program	54
3.6.10. Downloading the Ansible playbooks	55
3.6.11. The inventory.yml file	55
3.6.12. Specifying the RHCOS image settings	59
3.6.13. Creating the install config file	60
3.6.14. Customizing install-config.yaml	61
3.6.15. Generate manifest files	62
3.6.16. Making control-plane nodes non-schedulable	63
3.6.17. Building the ignition files	63
3.6.18. Creating templates and virtual machines	64
3.6.19. Creating the bootstrap machine	65
3.6.20. Creating the control plane nodes	65
3.6.21. Verifying cluster status	66
3.6.22. Removing the bootstrap machine	66
3.6.23. Creating the worker nodes and completing the installation	67
3.7. UNINSTALLING A CLUSTER ON RHV	69
3.7.1. Removing a cluster that uses installer-provisioned infrastructure	69
3.7.2. Removing a cluster that uses user-provisioned infrastructure	69

CHAPTER 1. INSTALLING ON RHV

1.1. INSTALLING A CLUSTER QUICKLY ON RHV

You can quickly install a default, non-customized, OpenShift Container Platform cluster on a Red Hat Virtualization (RHV) cluster, similar to the one shown in the following diagram.



The installation program uses installer-provisioned infrastructure to automate creating and deploying the cluster.

To install a default cluster, you prepare the environment, run the installation program and answer its prompts. Then, the installation program creates the OpenShift Container Platform cluster.

For an alternative to installing a default cluster, see [Installing a cluster with customizations](#).



NOTE

This installation program is available for Linux and macOS only.

1.1.1. Prerequisites

- Review details about the [OpenShift Container Platform installation and update](#) processes.
- If you use a firewall, [configure it to allow the sites](#) that your cluster requires access to.

1.1.2. Internet and Telemetry access for OpenShift Container Platform

In OpenShift Container Platform 4.5, you require access to the internet to install your cluster. The Telemetry service, which runs by default to provide metrics about cluster health and the success of updates, also requires internet access. If your cluster is connected to the internet, Telemetry runs automatically, and your cluster is registered to the [Red Hat OpenShift Cluster Manager \(OCM\)](#).

Once you confirm that your Red Hat OpenShift Cluster Manager inventory is correct, either maintained automatically by Telemetry or manually using OCM, [use subscription watch](#) to track your OpenShift Container Platform subscriptions at the account or multi-cluster level.

You must have internet access to:

- Access the [Red Hat OpenShift Cluster Manager](#) page to download the installation program and perform subscription management. If the cluster has internet access and you do not disable Telemetry, that service automatically entitles your cluster.
- Access [Quay.io](#) to obtain the packages that are required to install your cluster.
- Obtain the packages that are required to perform cluster updates.



IMPORTANT

If your cluster cannot have direct internet access, you can perform a restricted network installation on some types of infrastructure that you provision. During that process, you download the content that is required and use it to populate a mirror registry with the packages that you need to install a cluster and generate the installation program. With some installation types, the environment that you install your cluster in will not require internet access. Before you update the cluster, you update the content of the mirror registry.

1.1.3. Requirements for the RHV environment

To install and run an OpenShift Container Platform cluster, the RHV environment must meet the following requirements.

Not meeting these requirements can cause the installation or process to fail. Additionally, not meeting these requirements can cause the OpenShift Container Platform cluster to fail days or weeks after installation.

The following requirements for CPU, memory, and storage resources are based on **default** values multiplied by the default number of virtual machines the installation program creates. These resources must be available **in addition to** what the RHV environment uses for non-OpenShift Container Platform operations.

By default, the installation program creates seven virtual machines during the installation process. First, it creates a bootstrap virtual machine to provide temporary services and a control plane while it creates the rest of the OpenShift Container Platform cluster. When the installation program finishes creating the cluster, deleting the bootstrap machine frees up its resources.

If you increase the number of virtual machines in the RHV environment, you must increase the resources accordingly.

Requirements

- The RHV environment has one data center whose state is **Up**.
- The RHV data center contains an RHV cluster.

- The RHV cluster has the following resources exclusively for the OpenShift Container Platform cluster:
 - Minimum 28 vCPUs: four for each of the seven virtual machines created during installation.
 - 112 GiB RAM or more, including:
 - 16 GiB or more for the bootstrap machine, which provides the temporary control plane.
 - 16 GiB or more for each of the three control plane machines which provide the control plane.
 - 16 GiB or more for each of the three compute machines, which run the application workloads.
- The RHV storage domain must meet [these etcd backend performance requirements](#).
- In production environments, each virtual machine must have 120 GiB or more. Therefore, the storage domain must provide 840 GiB or more for the default OpenShift Container Platform cluster. In resource-constrained or non-production environments, each virtual machine must have 32 GiB or more, so the storage domain must have 230 GiB or more for the default OpenShift Container Platform cluster.
- To download images from the Red Hat Ecosystem Catalog during installation and update procedures, the RHV cluster must have access to an internet connection. The Telemetry service also needs an internet connection to simplify the subscription and entitlement process.
- The RHV cluster must have a virtual network with access to the REST API on the RHV Manager.
- A user account and group with the following least privileges for installing and managing an OpenShift Container Platform cluster on the target RHV cluster:
 - **DiskOperator**
 - **DiskCreator**
 - **UserTemplateBasedVm**
 - **TemplateOwner**
 - **TemplateCreator**
 - **ClusterAdmin** on the target cluster



WARNING

Apply the principle of least privilege: Avoid using an administrator account with **SuperUser** privileges on RHV during the installation process. The installation program saves the credentials you provide to a temporary **ovirt-config.yaml** file that might be compromised.

1.1.4. Verifying the requirements for the RHV environment

Verify that the RHV environment meets the requirements to install and run an OpenShift Container Platform cluster. Not meeting these requirements can cause failures.



IMPORTANT

These requirements are based on the default resources the installation program uses to create control plane and compute machines. These resources include vCPUs, memory, and storage. If you change these resources or increase the number of OpenShift Container Platform machines, adjust these requirements accordingly.

Procedure

1. Check the RHV version.
 - a. In the RHV Administration Portal, click the ? help icon in the upper-right corner and select **About**.
 - b. In the window that opens, make a note of the **RHV Software Version**
 - c. Confirm that version 4.5 of OpenShift Container Platform and the version of RHV you noted are one of the supported combinations in the [Support Matrix for OpenShift Container Platform on RHV](#).
2. Inspect the data center, cluster, and storage.
 - a. In the RHV Administration Portal, click **Compute → Data Centers**.
 - b. Confirm that the data center where you plan to install OpenShift Container Platform is accessible.
 - c. Click the name of that data center.
 - d. In the data center details, on the **Storage** tab, confirm the storage domain where you plan to install OpenShift Container Platform is **Active**.
 - e. Record the **Domain Name** for use later on.
 - f. Confirm **Free Space** has at least 230 GiB.
 - g. Confirm that the storage domain meets [these etcd backend performance requirements](#), which you [can measure by using the fio performance benchmarking tool](#).
 - h. In the data center details, click the **Clusters** tab.
 - i. Find the RHV cluster where you plan to install OpenShift Container Platform. Record the cluster name for use later on.
3. Inspect the RHV host resources.
 - a. In the RHV Administration Portal, click **Compute > Clusters**
 - b. Click the cluster where you plan to install OpenShift Container Platform.
 - c. In the cluster details, click the **Hosts** tab.
 - d. Inspect the hosts and confirm they have a combined total of at least 28 **Logical CPU Cores** available *exclusively* for the OpenShift Container Platform cluster.

- e. Record the number of available **Logical CPU Cores** for use later on.
 - f. Confirm that these CPU cores are distributed so that each of the seven virtual machines created during installation can have four cores.
 - g. Confirm that, all together, the hosts have 112 GiB of **Max free Memory for scheduling new virtual machines** distributed to meet the requirements for each of the following OpenShift Container Platform machines:
 - 16 GiB required for the bootstrap machine
 - 16 GiB required for each of the three control plane machines
 - 16 GiB for each of the three compute machines
 - h. Record the amount of **Max free Memory for scheduling new virtual machines** for use later on.
4. Verify that the virtual network for installing OpenShift Container Platform has access to the RHV Manager's REST API. From a virtual machine on this network, use curl to reach the RHV Manager's REST API:

```
$ curl -k -u <username>@<profile>:<password> \ 1
https://<engine-fqdn>/ovirt-engine/api 2
```

1 For **<username>**, specify the user name of an RHV account with privileges to create and manage an OpenShift Container Platform cluster on RHV. For **<profile>**, specify the login profile, which you can get by going to the RHV Administration Portal login page and reviewing the **Profile** dropdown list. For **<password>**, specify the password for that user name.

2 For **<engine-fqdn>**, specify the fully qualified domain name of the RHV environment.

For example:

```
$ curl -k -u ovirtadmin@internal:pw123 \
https://rhv-env.virtlab.example.com/ovirt-engine/api
```

1.1.5. Preparing the network environment on RHV

Configure three static IP addresses for the OpenShift Container Platform cluster and create DNS entries using two of these addresses.

Procedure

1. Reserve three static IP addresses
 - a. On the network where you plan to install OpenShift Container Platform, identify three static IP addresses that are outside the DHCP lease pool.
 - b. Connect to a host on this network and verify that each of the IP addresses is not in use. For example, use arp to check that none of the IP addresses have entries:

```
$ arp 10.35.1.19
```

Example output

```
10.35.1.19 (10.35.1.19) -- no entry
```

- c. Reserve three static IP addresses following the standard practices for your network environment.
 - d. Record these IP addresses for future reference.
2. Create DNS entries for the OpenShift Container Platform REST API and apps domain names using this format:

```
api.<cluster-name>.<base-domain> <ip-address> 1
*.apps.<cluster-name>.<base-domain> <ip-address> 2
```

- 1 For **<cluster-name>**, **<base-domain>**, and **<ip-address>**, specify the cluster name, base domain, and static IP address of your OpenShift Container Platform API.
- 2 Specify the cluster name, base domain, and static IP address of your OpenShift Container Platform apps (ingress/load balancer).

For example:

```
api.my-cluster.virtlab.example.com 10.35.1.19
*.apps.my-cluster.virtlab.example.com 10.35.1.20
```



NOTE

The third static IP address does not require a DNS entry. The OpenShift Container Platform cluster uses that address for its internal DNS service.

1.1.6. Setting up the CA certificate for RHV

Download the CA certificate from the Red Hat Virtualization (RHV) Manager and set it up on the installation machine.

You can download the certificate from a webpage on the RHV Manager or by using a **curl** command.

Later, you provide the certificate to the installation program.

Procedure

1. Use either of these two methods to download the CA certificate:
 - Go to the Manager's webpage, **<https://<engine-fqdn>/ovirt-engine/>**. Then, under **Downloads**, click the **CA Certificate** link.
 - Run the following command:

```
$ curl -k 'https://<engine-fqdn>/ovirt-engine/services/pki-resource?resource=ca-certificate&format=X509-PEM-CA' -o /tmp/ca.pem 1
```

- 1 For **<engine-fqdn>**, specify the fully qualified domain name of the RHV Manager, such

as **rhv-env.virtlab.example.com**.

- Configure the CA file to grant rootless user access to the Manager. Set the CA file permissions to have an octal value of **0644** (symbolic value: **-rw-r--r--**):

```
$ sudo chmod 0644 /tmp/ca.pem
```

- For Linux, copy the CA certificate to the directory for server certificates. Use **-p** to preserve the permissions:

```
$ sudo cp -p /tmp/ca.pem /etc/pki/ca-trust/source/anchors/ca.pem
```

- Add the certificate to the certificate manager for your operating system:

- For macOS, double-click the certificate file and use the **Keychain Access** utility to add the file to the **System** keychain.
- For Linux, update the CA trust:

```
$ sudo update-ca-trust
```



NOTE

If you use your own certificate authority, make sure the system trusts it.

Additional Resources

To learn more, see [Authentication and Security](#) in the RHV documentation.

1.1.7. Generating an SSH private key and adding it to the agent

If you want to perform installation debugging or disaster recovery on your cluster, you must provide an SSH key to both your **ssh-agent** and the installation program. You can use this key to access the bootstrap machine in a public cluster to troubleshoot installation issues.



NOTE

In a production environment, you require disaster recovery and debugging.

You can use this key to SSH into the master nodes as the user **core**. When you deploy the cluster, the key is added to the **core** user's `~/.ssh/authorized_keys` list.

Procedure

- If you do not have an SSH key that is configured for password-less authentication on your computer, create one. For example, on a computer that uses a Linux operating system, run the following command:

```
$ ssh-keygen -t rsa -b 4096 -N "" \
  -f <path>/<file_name> 1
```

- Specify the path and file name, such as `~/.ssh/id_rsa`, of the new SSH key.

Running this command generates an SSH key that does not require a password in the location that you specified.

2. Start the **ssh-agent** process as a background task:

```
$ eval "$(ssh-agent -s)"
```

Example output

```
Agent pid 31874
```

3. Add your SSH private key to the **ssh-agent**:

```
$ ssh-add <path>/<file_name> 1
```

Example output

```
Identity added: /home/<you>/<path>/<file_name> (<computer_name>)
```

- 1** Specify the path and file name for your SSH private key, such as `~/.ssh/id_rsa`

Next steps

- When you install OpenShift Container Platform, provide the SSH public key to the installation program.

1.1.8. Obtaining the installation program

Before you install OpenShift Container Platform, download the installation file on a local computer.

Prerequisites

- A computer that runs Linux or macOS, with 500 MB of local disk space

Procedure

1. Access the [Infrastructure Provider](#) page on the Red Hat OpenShift Cluster Manager site. If you have a Red Hat account, log in with your credentials. If you do not, create an account.
2. Navigate to the page for your installation type, download the installation program for your operating system, and place the file in the directory where you will store the installation configuration files.



IMPORTANT

The installation program creates several files on the computer that you use to install your cluster. You must keep both the installation program and the files that the installation program creates after you finish installing the cluster.



IMPORTANT

Deleting the files created by the installation program does not remove your cluster, even if the cluster failed during installation. To remove your cluster, complete the OpenShift Container Platform uninstallation procedures for your specific cloud provider.

3. Extract the installation program. For example, on a computer that uses a Linux operating system, run the following command:

```
$ tar xvf openshift-install-linux.tar.gz
```

4. From the [Pull Secret](#) page on the Red Hat OpenShift Cluster Manager site, download your installation pull secret as a **.txt** file. This pull secret allows you to authenticate with the services that are provided by the included authorities, including Quay.io, which serves the container images for OpenShift Container Platform components.

1.1.9. Deploying the cluster

You can install OpenShift Container Platform on a compatible cloud platform.



IMPORTANT

You can run the **create cluster** command of the installation program only once, during initial installation.

Prerequisites

- Obtain the OpenShift Container Platform installation program and the pull secret for your cluster.

Procedure

1. Run the installation program:

```
$. /openshift-install create cluster --dir=<installation_directory> \ 1  
--log-level=info 2
```

- 1** For **<installation_directory>**, specify the directory name to store the files that the installation program creates.

- 2** To view different installation details, specify **warn**, **debug**, or **error** instead of **info**.

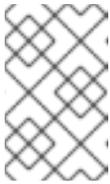


IMPORTANT

Specify an empty directory. Some installation assets, like bootstrap X.509 certificates have short expiration intervals, so you must not reuse an installation directory. If you want to reuse individual files from another cluster installation, you can copy them into your directory. However, the file names for the installation assets might change between releases. Use caution when copying installation files from an earlier OpenShift Container Platform version.

Respond to the installation program prompts.

- a. Optional: For **SSH Public Key**, select a password-less public key, such as `~/.ssh/id_rsa.pub`. This key authenticates connections with the new OpenShift Container Platform cluster.



NOTE

For production OpenShift Container Platform clusters on which you want to perform installation debugging or disaster recovery, select an SSH key that your **ssh-agent** process uses.

- b. For **Platform**, select **ovirt**.
- c. For **Enter oVirt's API endpoint URL**, enter the URL of the RHV API using this format:

```
https://<engine-fqdn>/ovirt-engine/api 1
```

- 1 For **<engine-fqdn>**, specify the fully qualified domain name of the RHV environment.

For example:

```
https://rhv-env.virtlab.example.com/ovirt-engine/api
```

- d. For **Is the oVirt CA trusted locally?**, enter **Yes** since you have already set up a CA certificate. Otherwise, enter **No**.
- e. For **oVirt's CA bundle**, if you entered **Yes** for the preceding question, copy the certificate content from `/etc/pki/ca-trust/source/anchors/ca.pem` and paste it here. Then, press **Enter** twice. Otherwise, if you entered **No** for the preceding question, this question does not appear.
- f. For **oVirt engine username**, enter the user name and profile of the RHV administrator using this format:

```
<username>@<profile> 1
```

- 1 For **<username>**, specify the user name of an RHV administrator. For **<profile>**, specify the login profile, which you can get by going to the RHV Administration Portal login page and reviewing the **Profile** dropdown list. Together, the user name and profile should look similar to this example:

```
admin@internal
```

- g. For **oVirt engine password**, enter the RHV admin password.
- h. For **oVirt cluster**, select the cluster for installing OpenShift Container Platform.
- i. For **oVirt storage domain**, select the storage domain for installing OpenShift Container Platform.
- j. For **oVirt network**, select a virtual network that has access to the RHV Manager REST API.

- k. For **Internal API Virtual IP**, enter the static IP address you set aside for the cluster's REST API.
- l. For **Internal DNS Virtual IP**, enter the static IP address you set aside for the cluster's internal DNS service.
- m. For **Ingress virtual IP**, enter the static IP address you reserved for the wildcard apps domain.
- n. For **Base Domain**, enter the base domain of the OpenShift Container Platform cluster. If this cluster is exposed to the outside world, this must be a valid domain recognized by DNS infrastructure. For example, enter: **virtlab.example.com**
- o. For **Cluster Name**, enter the name of the cluster. For example, **my-cluster**. Use cluster name from the externally registered/resolvable DNS entries you created for the OpenShift Container Platform REST API and apps domain names. The installation program also gives this name to the cluster in the RHV environment.
- p. For **Pull Secret**, copy the pull secret from the **pull-secret.txt** file you downloaded earlier and paste it here. You can also get a copy of the same pull secret from the [Pull Secret](#) page on the Red Hat OpenShift Cluster Manager site.



NOTE

If the cloud provider account that you configured on your host does not have sufficient permissions to deploy the cluster, the installation process stops, and the missing permissions are displayed.

When the cluster deployment completes, directions for accessing your cluster, including a link to its web console and credentials for the **kubeadmin** user, display in your terminal.



IMPORTANT

The Ignition config files that the installation program generates contain certificates that expire after 24 hours. You must keep the cluster running for 24 hours in a non-degraded state to ensure that the first certificate rotation has finished.



IMPORTANT

You must not delete the installation program or the files that the installation program creates. Both are required to delete the cluster.



IMPORTANT

You have completed the steps required to install the cluster. The remaining steps show you how to verify the cluster and troubleshoot the installation.

CHAPTER 2. INSTALLING THE CLI BY DOWNLOADING THE BINARY

You can install the OpenShift CLI (**oc**) in order to interact with OpenShift Container Platform from a command-line interface. You can install **oc** on Linux, Windows, or macOS.



IMPORTANT

If you installed an earlier version of **oc**, you cannot use it to complete all of the commands in OpenShift Container Platform 4.5. Download and install the new version of **oc**.

2.1. INSTALLING THE CLI ON LINUX

You can install the OpenShift CLI (**oc**) binary on Linux by using the following procedure.

Procedure

1. Navigate to the [Infrastructure Provider](#) page on the Red Hat OpenShift Cluster Manager site.
2. Select your infrastructure provider, and, if applicable, your installation type.
3. In the **Command-line interface** section, select **Linux** from the drop-down menu and click **Download command-line tools**.
4. Unpack the archive:

```
$ tar xvzf <file>
```

5. Place the **oc** binary in a directory that is on your **PATH**.
To check your **PATH**, execute the following command:

```
$ echo $PATH
```

After you install the CLI, it is available using the **oc** command:

```
$ oc <command>
```

2.2. INSTALLING THE CLI ON WINDOWS

You can install the OpenShift CLI (**oc**) binary on Windows by using the following procedure.

Procedure

1. Navigate to the [Infrastructure Provider](#) page on the Red Hat OpenShift Cluster Manager site.
2. Select your infrastructure provider, and, if applicable, your installation type.
3. In the **Command-line interface** section, select **Windows** from the drop-down menu and click **Download command-line tools**.
4. Unzip the archive with a ZIP program.

5. Move the **oc** binary to a directory that is on your **PATH**.
To check your **PATH**, open the command prompt and execute the following command:

```
C:\> path
```

After you install the CLI, it is available using the **oc** command:

```
C:\> oc <command>
```

2.3. INSTALLING THE CLI ON MACOS

You can install the OpenShift CLI (**oc**) binary on macOS by using the following procedure.

Procedure

1. Navigate to the [Infrastructure Provider](#) page on the Red Hat OpenShift Cluster Manager site.
2. Select your infrastructure provider, and, if applicable, your installation type.
3. In the **Command-line interface** section, select **MacOS** from the drop-down menu and click **Download command-line tools**.
4. Unpack and unzip the archive.
5. Move the **oc** binary to a directory on your PATH.
To check your **PATH**, open a terminal and execute the following command:

```
$ echo $PATH
```

After you install the CLI, it is available using the **oc** command:

```
$ oc <command>
```

To learn more, see [Getting started with the CLI](#).

CHAPTER 3. LOGGING IN TO THE CLUSTER

You can log in to your cluster as a default system user by exporting the cluster **kubeconfig** file. The **kubeconfig** file contains information about the cluster that is used by the CLI to connect a client to the correct cluster and API server. The file is specific to a cluster and is created during OpenShift Container Platform installation.

Prerequisites

- Deploy an OpenShift Container Platform cluster.
- Install the **oc** CLI.

Procedure

1. Export the **kubeadmin** credentials:

```
$ export KUBECONFIG=<installation_directory>/auth/kubeconfig 1
```

- 1** For **<installation_directory>**, specify the path to the directory that you stored the installation files in.

2. Verify you can run **oc** commands successfully using the exported configuration:

```
$ oc whoami
```

Example output

```
system:admin
```

3.1. VERIFYING CLUSTER STATUS

You can verify your OpenShift Container Platform cluster's status during or after installation.

Procedure

1. In the cluster environment, export the administrator's kubeconfig file:

```
$ export KUBECONFIG=<installation_directory>/auth/kubeconfig 1
```

- 1** For **<installation_directory>**, specify the path to the directory that you stored the installation files in.

The **kubeconfig** file contains information about the cluster that is used by the CLI to connect a client to the correct cluster and API server.

2. View the control plane and compute machines created after a deployment:

```
$ oc get nodes
```

3. View your cluster's version:

```
$ oc get clusterversion
```

4. View your operators' status:

```
$ oc get clusteroperator
```

5. View all running pods in the cluster:

```
$ oc get pods -A
```

Troubleshooting

If the installation fails, the installation program times out and displays an error message. To learn more, see [Troubleshooting installation issues](#).

3.2. ACCESSING THE OPENSIFT CONTAINER PLATFORM WEB CONSOLE ON RHV

After the OpenShift Container Platform cluster initializes, you can log into the OpenShift Container Platform web console.

Procedure

1. Optional: In the Red Hat Virtualization (RHV) Administration Portal, open **Compute → Cluster**.
2. Verify that the installation program creates the virtual machines.
3. Return to the command line where the installation program is running. When the installation program finishes, it displays the user name and temporary password for logging into the OpenShift Container Platform web console.
4. In a browser, open the URL of the OpenShift Container Platform web console. The URL uses this format:

```
console-openshift-console.apps.<clustername>.<basedomain> 1
```

- 1** For **<clustername>.<basedomain>**, specify the cluster name and base domain.

For example:

```
console-openshift-console.apps.my-cluster.virtlab.example.com
```

3.3. TROUBLESHOOTING COMMON ISSUES WITH INSTALLING ON RED HAT VIRTUALIZATION (RHV)

Here are some common issues you might encounter, along with proposed causes and solutions.

3.3.1. CPU load increases and nodes go into a Not Ready state

- **Symptom:** CPU load increases significantly and nodes start going into a Not Ready state.

- **Cause:** The storage domain latency might be too high, especially for master nodes.
- **Solution:**
Make the nodes Ready again by restarting the kubelet service:

```
$ systemctl restart kubelet
```

Inspect the OpenShift Container Platform metrics service, which automatically gathers and reports on some valuable data such as the etcd disk sync duration. If the cluster is operational, use this data to help determine whether storage latency or throughput is the root issue. If so, consider using a storage resource that has lower latency and higher throughput.

To get raw metrics, enter the following command as kubeadmin or user with cluster-admin privileges:

```
$ oc get --insecure-skip-tls-verify --server=https://localhost:<port> --raw=/metrics
```

To learn more, see [Exploring Application Endpoints for the purposes of Debugging with OpenShift 4.x](#)

3.3.2. Trouble connecting the OpenShift Container Platform cluster API

- **Symptom:** The installation program completes but the OpenShift Container Platform cluster API is not available. The bootstrap virtual machine remains up after the bootstrap process is complete. When you enter the following command, the response will time out.

```
$ oc login -u kubeadmin -p *** <apiurl>
```

- **Cause:** The bootstrap VM was not deleted by the installation program and has not released the cluster's API IP address.
- **Solution:** Use the **wait-for** subcommand to be notified when the bootstrap process is complete:

```
$ ./openshift-install wait-for bootstrap-complete
```

When the bootstrap process is complete, delete the bootstrap virtual machine:

```
$ ./openshift-install destroy bootstrap
```

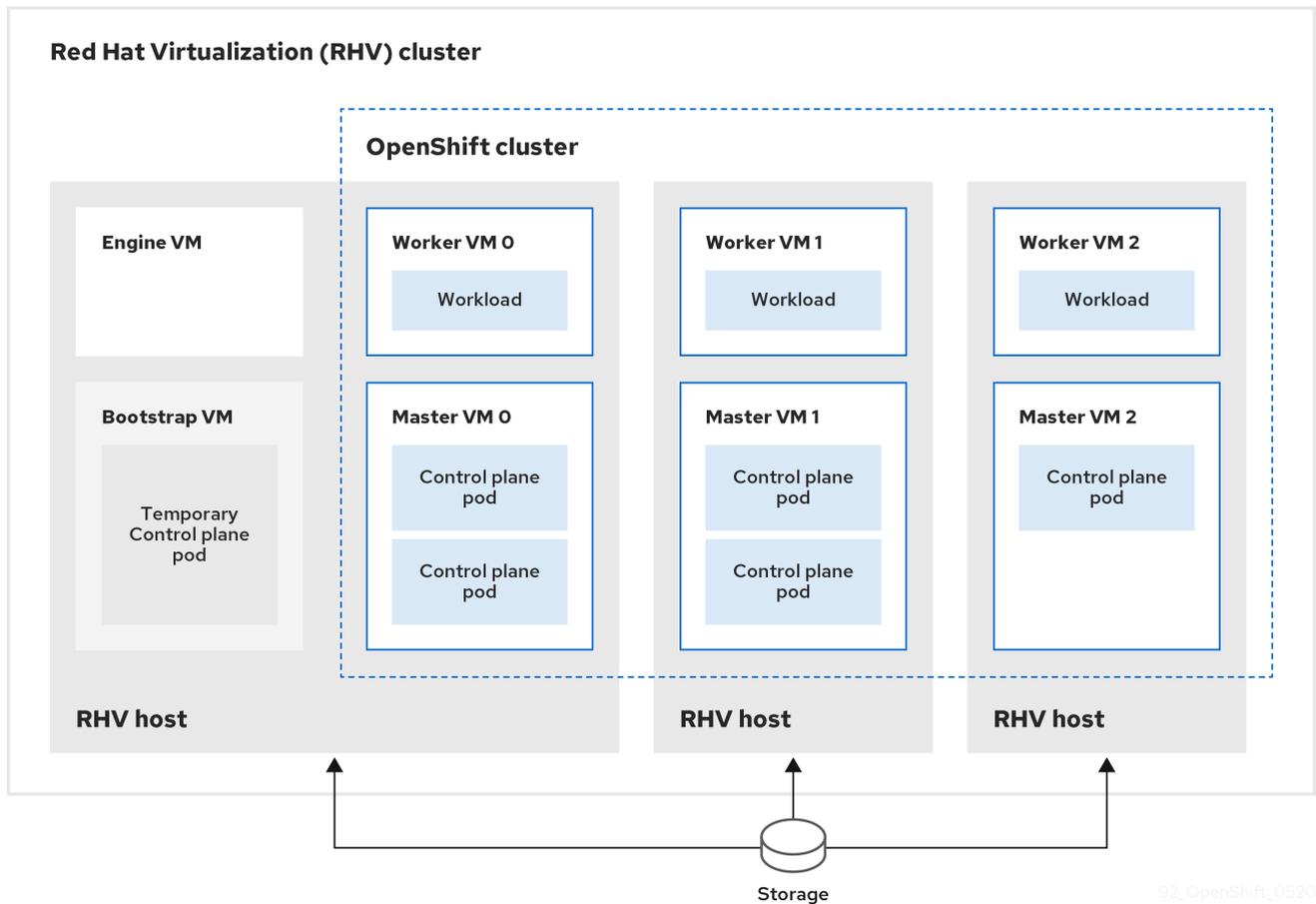
3.4. POST-INSTALLATION TASKS

After the OpenShift Container Platform cluster initializes, you can perform the following tasks.

- Optional: After deployment, add or replace SSH keys using the Machine Config Operator (MCO) in OpenShift Container Platform.
- Optional: Remove the **kubeadmin** user. Instead, use the authentication provider to create a user with cluster-admin privileges.

3.5. INSTALLING A CLUSTER ON RHV WITH CUSTOMIZATIONS

You can customize and install an OpenShift Container Platform cluster on Red Hat Virtualization (RHV), similar to the one shown in the following diagram.



92_OpenShift_0520

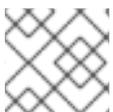
The installation program uses installer-provisioned infrastructure to automate creating and deploying the cluster.

To install a customized cluster, you prepare the environment and perform the following steps:

1. Create an installation configuration file, the **install-config.yaml** file, by running the installation program and answering its prompts.
2. Inspect and modify parameters in the **install-config.yaml** file.
3. Make a working copy of the **install-config.yaml** file.
4. Run the installation program with a copy of the **install-config.yaml** file.

Then, the installation program creates the OpenShift Container Platform cluster.

For an alternative to installing a customized cluster, see [Installing a default cluster](#).



NOTE

This installation program is available for Linux and macOS only.

3.5.1. Prerequisites

- Review details about the [OpenShift Container Platform installation and update](#) processes.
- If you use a firewall, [configure it to allow the sites](#) that your cluster requires access to.

3.5.2. Internet and Telemetry access for OpenShift Container Platform

In OpenShift Container Platform 4.5, you require access to the internet to install your cluster. The Telemetry service, which runs by default to provide metrics about cluster health and the success of updates, also requires internet access. If your cluster is connected to the internet, Telemetry runs automatically, and your cluster is registered to the [Red Hat OpenShift Cluster Manager \(OCM\)](#).

Once you confirm that your Red Hat OpenShift Cluster Manager inventory is correct, either maintained automatically by Telemetry or manually using OCM, [use subscription watch](#) to track your OpenShift Container Platform subscriptions at the account or multi-cluster level.

You must have internet access to:

- Access the [Red Hat OpenShift Cluster Manager](#) page to download the installation program and perform subscription management. If the cluster has internet access and you do not disable Telemetry, that service automatically entitles your cluster.
- Access [Quay.io](#) to obtain the packages that are required to install your cluster.
- Obtain the packages that are required to perform cluster updates.



IMPORTANT

If your cluster cannot have direct internet access, you can perform a restricted network installation on some types of infrastructure that you provision. During that process, you download the content that is required and use it to populate a mirror registry with the packages that you need to install a cluster and generate the installation program. With some installation types, the environment that you install your cluster in will not require internet access. Before you update the cluster, you update the content of the mirror registry.

3.5.3. Requirements for the RHV environment

To install and run an OpenShift Container Platform cluster, the RHV environment must meet the following requirements.

Not meeting these requirements can cause the installation or process to fail. Additionally, not meeting these requirements can cause the OpenShift Container Platform cluster to fail days or weeks after installation.

The following requirements for CPU, memory, and storage resources are based on **default** values multiplied by the default number of virtual machines the installation program creates. These resources must be available **in addition to** what the RHV environment uses for non-OpenShift Container Platform operations.

By default, the installation program creates seven virtual machines during the installation process. First, it creates a bootstrap virtual machine to provide temporary services and a control plane while it creates the rest of the OpenShift Container Platform cluster. When the installation program finishes creating the cluster, deleting the bootstrap machine frees up its resources.

If you increase the number of virtual machines in the RHV environment, you must increase the resources accordingly.

Requirements

- The RHV environment has one data center whose state is **Up**.

- The RHV data center contains an RHV cluster.
- The RHV cluster has the following resources exclusively for the OpenShift Container Platform cluster:
 - Minimum 28 vCPUs: four for each of the seven virtual machines created during installation.
 - 112 GiB RAM or more, including:
 - 16 GiB or more for the bootstrap machine, which provides the temporary control plane.
 - 16 GiB or more for each of the three control plane machines which provide the control plane.
 - 16 GiB or more for each of the three compute machines, which run the application workloads.
- The RHV storage domain must meet [these etcd backend performance requirements](#).
- In production environments, each virtual machine must have 120 GiB or more. Therefore, the storage domain must provide 840 GiB or more for the default OpenShift Container Platform cluster. In resource-constrained or non-production environments, each virtual machine must have 32 GiB or more, so the storage domain must have 230 GiB or more for the default OpenShift Container Platform cluster.
- To download images from the Red Hat Ecosystem Catalog during installation and update procedures, the RHV cluster must have access to an internet connection. The Telemetry service also needs an internet connection to simplify the subscription and entitlement process.
- The RHV cluster must have a virtual network with access to the REST API on the RHV Manager.
- A user account and group with the following least privileges for installing and managing an OpenShift Container Platform cluster on the target RHV cluster:
 - **DiskOperator**
 - **DiskCreator**
 - **UserTemplateBasedVm**
 - **TemplateOwner**
 - **TemplateCreator**
 - **ClusterAdmin** on the target cluster



WARNING

Apply the principle of least privilege: Avoid using an administrator account with **SuperUser** privileges on RHV during the installation process. The installation program saves the credentials you provide to a temporary **ovirt-config.yaml** file that might be compromised.

3.5.4. Verifying the requirements for the RHV environment

Verify that the RHV environment meets the requirements to install and run an OpenShift Container Platform cluster. Not meeting these requirements can cause failures.



IMPORTANT

These requirements are based on the default resources the installation program uses to create control plane and compute machines. These resources include vCPUs, memory, and storage. If you change these resources or increase the number of OpenShift Container Platform machines, adjust these requirements accordingly.

Procedure

1. Check the RHV version.
 - a. In the RHV Administration Portal, click the ? help icon in the upper-right corner and select **About**.
 - b. In the window that opens, make a note of the **RHV Software Version**
 - c. Confirm that version 4.5 of OpenShift Container Platform and the version of RHV you noted are one of the supported combinations in the [Support Matrix for OpenShift Container Platform on RHV](#).
2. Inspect the data center, cluster, and storage.
 - a. In the RHV Administration Portal, click **Compute → Data Centers**.
 - b. Confirm that the data center where you plan to install OpenShift Container Platform is accessible.
 - c. Click the name of that data center.
 - d. In the data center details, on the **Storage** tab, confirm the storage domain where you plan to install OpenShift Container Platform is **Active**.
 - e. Record the **Domain Name** for use later on.
 - f. Confirm **Free Space** has at least 230 GiB.
 - g. Confirm that the storage domain meets [these etcd backend performance requirements](#), which you [can measure by using the fio performance benchmarking tool](#).
 - h. In the data center details, click the **Clusters** tab.
 - i. Find the RHV cluster where you plan to install OpenShift Container Platform. Record the cluster name for use later on.
3. Inspect the RHV host resources.
 - a. In the RHV Administration Portal, click **Compute > Clusters**
 - b. Click the cluster where you plan to install OpenShift Container Platform.
 - c. In the cluster details, click the **Hosts** tab.

- d. Inspect the hosts and confirm they have a combined total of at least 28 **Logical CPU Cores** available *exclusively* for the OpenShift Container Platform cluster.
 - e. Record the number of available **Logical CPU Cores** for use later on.
 - f. Confirm that these CPU cores are distributed so that each of the seven virtual machines created during installation can have four cores.
 - g. Confirm that, all together, the hosts have 112 GiB of **Max free Memory for scheduling new virtual machines** distributed to meet the requirements for each of the following OpenShift Container Platform machines:
 - 16 GiB required for the bootstrap machine
 - 16 GiB required for each of the three control plane machines
 - 16 GiB for each of the three compute machines
 - h. Record the amount of **Max free Memory for scheduling new virtual machines** for use later on.
4. Verify that the virtual network for installing OpenShift Container Platform has access to the RHV Manager's REST API. From a virtual machine on this network, use curl to reach the RHV Manager's REST API:

```
$ curl -k -u <username>@<profile>:<password> \ 1
https://<engine-fqdn>/ovirt-engine/api 2
```

1 For **<username>**, specify the user name of an RHV account with privileges to create and manage an OpenShift Container Platform cluster on RHV. For **<profile>**, specify the login profile, which you can get by going to the RHV Administration Portal login page and reviewing the **Profile** dropdown list. For **<password>**, specify the password for that user name.

2 For **<engine-fqdn>**, specify the fully qualified domain name of the RHV environment.

For example:

```
$ curl -k -u ovirtadmin@internal:pw123 \
https://rhv-env.virtlab.example.com/ovirt-engine/api
```

3.5.5. Preparing the network environment on RHV

Configure three static IP addresses for the OpenShift Container Platform cluster and create DNS entries using two of these addresses.

Procedure

1. Reserve three static IP addresses
 - a. On the network where you plan to install OpenShift Container Platform, identify three static IP addresses that are outside the DHCP lease pool.
 - b. Connect to a host on this network and verify that each of the IP addresses is not in use. For example, use `arp` to check that none of the IP addresses have entries:

```
$ arp 10.35.1.19
```

Example output

```
10.35.1.19 (10.35.1.19) -- no entry
```

- c. Reserve three static IP addresses following the standard practices for your network environment.
 - d. Record these IP addresses for future reference.
2. Create DNS entries for the OpenShift Container Platform REST API and apps domain names using this format:

```
api.<cluster-name>.<base-domain> <ip-address> 1
*.apps.<cluster-name>.<base-domain> <ip-address> 2
```

- 1 For **<cluster-name>**, **<base-domain>**, and **<ip-address>**, specify the cluster name, base domain, and static IP address of your OpenShift Container Platform API.
- 2 Specify the cluster name, base domain, and static IP address of your OpenShift Container Platform apps (ingress/load balancer).

For example:

```
api.my-cluster.virtlab.example.com 10.35.1.19
*.apps.my-cluster.virtlab.example.com 10.35.1.20
```



NOTE

The third static IP address does not require a DNS entry. The OpenShift Container Platform cluster uses that address for its internal DNS service.

3.5.6. Setting up the CA certificate for RHV

Download the CA certificate from the Red Hat Virtualization (RHV) Manager and set it up on the installation machine.

You can download the certificate from a webpage on the RHV Manager or by using a **curl** command.

Later, you provide the certificate to the installation program.

Procedure

1. Use either of these two methods to download the CA certificate:
 - Go to the Manager's webpage, <https://<engine-fqdn>/ovirt-engine/>. Then, under **Downloads**, click the **CA Certificate** link.
 - Run the following command:

```
$ curl -k 'https://<engine-fqdn>/ovirt-engine/services/pki-resource?resource=ca-certificate&format=X509-PEM-CA' -o /tmp/ca.pem 1
```

- 1** For **<engine-fqdn>**, specify the fully qualified domain name of the RHV Manager, such as **rhv-env.virtlab.example.com**.

2. Configure the CA file to grant rootless user access to the Manager. Set the CA file permissions to have an octal value of **0644** (symbolic value: **-rw-r--r--**):

```
$ sudo chmod 0644 /tmp/ca.pem
```

3. For Linux, copy the CA certificate to the directory for server certificates. Use **-p** to preserve the permissions:

```
$ sudo cp -p /tmp/ca.pem /etc/pki/ca-trust/source/anchors/ca.pem
```

4. Add the certificate to the certificate manager for your operating system:

- For macOS, double-click the certificate file and use the **Keychain Access** utility to add the file to the **System** keychain.
- For Linux, update the CA trust:

```
$ sudo update-ca-trust
```



NOTE

If you use your own certificate authority, make sure the system trusts it.

Additional Resources

To learn more, see [Authentication and Security](#) in the RHV documentation.

3.5.7. Generating an SSH private key and adding it to the agent

If you want to perform installation debugging or disaster recovery on your cluster, you must provide an SSH key to both your **ssh-agent** and the installation program. You can use this key to access the bootstrap machine in a public cluster to troubleshoot installation issues.



NOTE

In a production environment, you require disaster recovery and debugging.

You can use this key to SSH into the master nodes as the user **core**. When you deploy the cluster, the key is added to the **core** user's **~/.ssh/authorized_keys** list.

Procedure

1. If you do not have an SSH key that is configured for password-less authentication on your computer, create one. For example, on a computer that uses a Linux operating system, run the following command:

```
$ ssh-keygen -t rsa -b 4096 -N "" \
-f <path>/<file_name> 1
```

- 1 Specify the path and file name, such as `~/.ssh/id_rsa`, of the new SSH key.

Running this command generates an SSH key that does not require a password in the location that you specified.

2. Start the **ssh-agent** process as a background task:

```
$ eval "$(ssh-agent -s)"
```

Example output

```
Agent pid 31874
```

3. Add your SSH private key to the **ssh-agent**:

```
$ ssh-add <path>/<file_name> 1
```

Example output

```
Identity added: /home/<you>/<path>/<file_name> (<computer_name>)
```

- 1 Specify the path and file name for your SSH private key, such as `~/.ssh/id_rsa`

Next steps

- When you install OpenShift Container Platform, provide the SSH public key to the installation program.

3.5.8. Obtaining the installation program

Before you install OpenShift Container Platform, download the installation file on a local computer.

Prerequisites

- A computer that runs Linux or macOS, with 500 MB of local disk space

Procedure

1. Access the [Infrastructure Provider](#) page on the Red Hat OpenShift Cluster Manager site. If you have a Red Hat account, log in with your credentials. If you do not, create an account.
2. Navigate to the page for your installation type, download the installation program for your operating system, and place the file in the directory where you will store the installation configuration files.

**IMPORTANT**

The installation program creates several files on the computer that you use to install your cluster. You must keep both the installation program and the files that the installation program creates after you finish installing the cluster.

**IMPORTANT**

Deleting the files created by the installation program does not remove your cluster, even if the cluster failed during installation. To remove your cluster, complete the OpenShift Container Platform uninstallation procedures for your specific cloud provider.

3. Extract the installation program. For example, on a computer that uses a Linux operating system, run the following command:

```
$ tar xvf openshift-install-linux.tar.gz
```

4. From the [Pull Secret](#) page on the Red Hat OpenShift Cluster Manager site, download your installation pull secret as a **.txt** file. This pull secret allows you to authenticate with the services that are provided by the included authorities, including Quay.io, which serves the container images for OpenShift Container Platform components.

3.5.9. Creating the installation configuration file

You can customize the OpenShift Container Platform cluster you install on Red Hat Virtualization (RHV).

Prerequisites

- Obtain the OpenShift Container Platform installation program and the pull secret for your cluster.

Procedure

1. Create the **install-config.yaml** file.
 - a. Run the following command:

```
$ ./openshift-install create install-config --dir=<installation_directory> 1
```

- 1** For **<installation_directory>**, specify the directory name to store the files that the installation program creates.

**IMPORTANT**

Specify an empty directory. Some installation assets, like bootstrap X.509 certificates have short expiration intervals, so you must not reuse an installation directory. If you want to reuse individual files from another cluster installation, you can copy them into your directory. However, the file names for the installation assets might change between releases. Use caution when copying installation files from an earlier OpenShift Container Platform version.

- b. Respond to the installation program prompts.
- i. For **SSH Public Key**, select a password-less public key, such as `~/.ssh/id_rsa.pub`. This key authenticates connections with the new OpenShift Container Platform cluster.

**NOTE**

For production OpenShift Container Platform clusters on which you want to perform installation debugging or disaster recovery, select an SSH key that your **ssh-agent** process uses.

- ii. For **Platform**, select **ovirt**.
- iii. For **Enter oVirt's API endpoint URL**, enter the URL of the RHV API using this format:

```
https://<engine-fqdn>/ovirt-engine/api 1
```

- 1** For **<engine-fqdn>**, specify the fully qualified domain name of the RHV environment.

For example:

```
https://rhv-env.virtlab.example.com/ovirt-engine/api
```

- iv. For **Is the oVirt CA trusted locally?**, enter **Yes** since you have already set up a CA certificate. Otherwise, enter **No**.
- v. For **oVirt's CA bundle**, if you entered **Yes** for the preceding question, copy the certificate content from `/etc/pki/ca-trust/source/anchors/ca.pem` and paste it here. Then, press **Enter** twice. Otherwise, if you entered **No** for the preceding question, this question does not appear.
- vi. For **oVirt engine username**, enter the user name and profile of the RHV administrator using this format:

```
<username>@<profile> 1
```

- 1** For **<username>**, specify the user name of an RHV administrator. For **<profile>**, specify the login profile, which you can get by going to the RHV Administration Portal login page and reviewing the **Profile** dropdown list. Together, the user name and profile should look similar to this example:

```
admin@internal
```

- vii. For **oVirt engine password**, enter the RHV admin password.
- viii. For **oVirt cluster**, select the cluster for installing OpenShift Container Platform.
- ix. For **oVirt storage domain**, select the storage domain for installing OpenShift Container Platform.
- x. For **oVirt network** select a virtual network that has access to the RHV Manager RFE

- x. For **External Network**, select a virtual network that has access to the RHV manager REST API.
 - xi. For **Internal API Virtual IP**, enter the static IP address you set aside for the cluster's REST API.
 - xii. For **Internal DNS Virtual IP**, enter the static IP address you set aside for the cluster's internal DNS service.
 - xiii. For **Ingress virtual IP**, enter the static IP address you reserved for the wildcard apps domain.
 - xiv. For **Base Domain**, enter the base domain of the OpenShift Container Platform cluster. If this cluster is exposed to the outside world, this must be a valid domain recognized by DNS infrastructure. For example, enter: **virtlab.example.com**
 - xv. For **Cluster Name**, enter the name of the cluster. For example, **my-cluster**. Use cluster name from the externally registered/resolvable DNS entries you created for the OpenShift Container Platform REST API and apps domain names. The installation program also gives this name to the cluster in the RHV environment.
 - xvi. For **Pull Secret**, copy the pull secret from the **pull-secret.txt** file you downloaded earlier and paste it here. You can also get a copy of the same pull secret from the [Pull Secret](#) page on the Red Hat OpenShift Cluster Manager site.
2. Modify the **install-config.yaml** file. You can find more information about the available parameters in the **Installation configuration parameters** section.
 3. Back up the **install-config.yaml** file so that you can use it to install multiple clusters.



IMPORTANT

The **install-config.yaml** file is consumed during the installation process. If you want to reuse the file, you must back it up now.

3.5.9.1. Example **install-config.yaml** files for Red Hat Virtualization (RHV)

You can customize the OpenShift Container Platform cluster the installation program creates by changing the parameters and parameter values in the **install-config.yaml** file.

The following example is specific to installing OpenShift Container Platform on RHV.

This file is located in the **<installation_directory>** you specified when you ran the following command.

```
$ ./openshift-install create install-config --dir=<installation_directory>
```



NOTE

- These example files are provided for reference only. You must obtain your **install-config.yaml** file by using the installation program.
- Changing the **install-config.yaml** file can increase the resources your cluster requires. Verify that your RHV environment has those additional resources. Otherwise, the installation or cluster will fail.

Example: This is the default install-config.yaml file

```

apiVersion: v1
baseDomain: example.com
compute:
- architecture: amd64
  hyperthreading: Enabled
  name: worker
  platform: {}
  replicas: 3
controlPlane:
  architecture: amd64
  hyperthreading: Enabled
  name: master
  platform: {}
  replicas: 3
metadata:
  creationTimestamp: null
  name: my-cluster
networking:
  clusterNetwork:
  - cidr: 10.128.0.0/14
    hostPrefix: 23
  machineNetwork:
  - cidr: 10.0.0.0/16
  networkType: OpenShiftSDN
  serviceNetwork:
  - 172.30.0.0/16
platform:
  ovirt:
    api_vip: 10.46.8.230
    ingress_vip: 192.168.1.5
    ovirt_cluster_id: 68833f9f-e89c-4891-b768-e2ba0815b76b
    ovirt_storage_domain_id: ed7b0f4e-0e96-492a-8fff-279213ee1468
    ovirt_network_name: ovirtmgmt
    vnicProfileID: 3fa86930-0be5-4052-b667-b79f0a729692
publish: External
pullSecret: '{"auths": ...}'
sshKey: ssh-ed12345 AAAA...

```

Example: A minimal install-config.yaml file

```

apiVersion: v1
baseDomain: example.com
metadata:
  name: test-cluster
platform:
  ovirt:
    api_vip: 10.46.8.230
    ingress_vip: 10.46.8.232
    ovirt_cluster_id: 68833f9f-e89c-4891-b768-e2ba0815b76b
    ovirt_storage_domain_id: ed7b0f4e-0e96-492a-8fff-279213ee1468
    ovirt_network_name: ovirtmgmt

```

```

vnicProfileID: 3fa86930-0be5-4052-b667-b79f0a729692
pullSecret: '{"auths": ...}'
sshKey: ssh-ed12345 AAAA...

```

Example: Custom machine pools in an `install-config.yaml` file

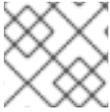
```

apiVersion: v1
baseDomain: example.com
controlPlane:
  name: master
  platform:
    ovirt:
      cpu:
        cores: 4
        sockets: 2
      memoryMB: 65536
      osDisk:
        sizeGB: 100
      vmType: server
  replicas: 3
compute:
- name: worker
  platform:
    ovirt:
      cpu:
        cores: 4
        sockets: 4
      memoryMB: 65536
      osDisk:
        sizeGB: 200
      vmType: server
  replicas: 5
metadata:
  name: test-cluster
platform:
  ovirt:
    api_vip: 10.46.8.230
    ingress_vip: 10.46.8.232
    ovirt_cluster_id: 68833f9f-e89c-4891-b768-e2ba0815b76b
    ovirt_storage_domain_id: ed7b0f4e-0e96-492a-8fff-279213ee1468
    ovirt_network_name: ovirtmgmt
    vnicProfileID: 3fa86930-0be5-4052-b667-b79f0a729692
  pullSecret: '{"auths": ...}'
  sshKey: ssh-ed25519 AAAA...

```

3.5.9.2. Installation configuration parameters

Before you deploy an OpenShift Container Platform cluster, you provide parameter values to describe your account on the cloud platform that hosts your cluster and optionally customize your cluster's platform. When you create the **`install-config.yaml`** installation configuration file, you provide values for the required parameters through the command line. If you customize your cluster, you can modify the **`install-config.yaml`** file to provide more details about the platform.

**NOTE**

After installation, you cannot modify these parameters in the **install-config.yaml** file.

Table 3.1. Required parameters

Parameter	Description	Values
apiVersion	The API version for the install-config.yaml content. The current version is v1 . The installer may also support older API versions.	String
baseDomain	The base domain of your cloud provider. The base domain is used to create routes to your OpenShift Container Platform cluster components. The full DNS name for your cluster is a combination of the baseDomain and metadata.name parameter values that uses the <metadata.name> . <baseDomain> format.	A fully-qualified domain or subdomain name, such as example.com .
metadata	Kubernetes resource ObjectMeta, from which only the name parameter is consumed.	Object
metadata.name	The name of the cluster. DNS records for the cluster are all subdomains of {{.metadata.name}} . {{.baseDomain}} .	String of lowercase letters, hyphens (-), and periods (.), such as dev .
platform	The configuration for the specific platform upon which to perform the installation: aws, baremetal, azure, openstack, ovirt, vsphere . For additional information about platform.<platform> parameters, consult the following table for your specific platform.	Object

Parameter	Description	Values
pullSecret	Get this pull secret from https://cloud.redhat.com/openshift/install/pull-secret to authenticate downloading container images for OpenShift Container Platform components from services such as Quay.io.	<pre>{ "auths":{ "cloud.openshift.com":{ "auth":"b3Blb=", "email":"you@example.com" }, "quay.io":{ "auth":"b3Blb=", "email":"you@example.com" } } }</pre>

Table 3.2. Optional parameters

Parameter	Description	Values
additionalTrustBundle	A PEM-encoded X.509 certificate bundle that is added to the nodes' trusted certificate store. This trust bundle may also be used when a proxy has been configured.	String
compute	The configuration for the machines that comprise the compute nodes.	Array of machine-pool objects. For details, see the following "Machine-pool" table.
compute.architecture	Determines the instruction set architecture of the machines in the pool. Currently, heterogeneous clusters are not supported, so all pools must specify the same architecture. Valid values are amd64 (the default).	String

Parameter	Description	Values
compute.hyperthreading	<p>Whether to enable or disable simultaneous multithreading, or hyperthreading, on compute machines. By default, simultaneous multithreading is enabled to increase the performance of your machines' cores.</p> <div style="display: flex; align-items: center;">  <div> <p>IMPORTANT</p> <p>If you disable simultaneous multithreading, ensure that your capacity planning accounts for the dramatically decreased machine performance.</p> </div> </div>	Enabled or Disabled
compute.name	Required if you use compute . The name of the machine pool.	worker
compute.platform	Required if you use compute . Use this parameter to specify the cloud provider to host the worker machines. This parameter value must match the controlPlane.platform parameter value.	aws, azure, gcp, openstack, ovirt, vsphere , or {}
compute.replicas	The number of compute machines, which are also known as worker machines, to provision.	A positive integer greater than or equal to 2 . The default value is 3 .
controlPlane	The configuration for the machines that comprise the control plane.	Array of machine-pool objects. For details, see the following "Machine-pool" table.
controlPlane.architecture	Determines the instruction set architecture of the machines in the pool. Currently, heterogeneous clusters are not supported, so all pools must specify the same architecture. Valid values are amd64 (the default).	String

Parameter	Description	Values
controlPlane.hyperthreading	<p>Whether to enable or disable simultaneous multithreading, or hyperthreading, on control plane machines. By default, simultaneous multithreading is enabled to increase the performance of your machines' cores.</p>  <p>IMPORTANT</p> <p>If you disable simultaneous multithreading, ensure that your capacity planning accounts for the dramatically decreased machine performance.</p>	Enabled or Disabled
controlPlane.name	Required if you use controlPlane . The name of the machine pool.	master
controlPlane.platform	Required if you use controlPlane . Use this parameter to specify the cloud provider that hosts the control plane machines. This parameter value must match the compute.platform parameter value.	aws, azure, gcp, openstack, ovirt, vsphere , or {}
controlPlane.replicas	The number of control plane machines to provision.	A positive integer greater than or equal to 3 . The default value is 3 .
credentialsMode	<p>The Cloud Credential Operator (CCO) mode. If no mode is specified, the CCO dynamically tries to determine the capabilities of the provided credentials, with a preference for mint mode on the platforms where multiple modes are supported.</p>  <p>NOTE</p> <p>Not all CCO modes are supported for all cloud providers. For more information on CCO modes, see the <i>Cloud Credential Operator</i> entry in the <i>Red Hat Operators reference content</i>.</p>	Mint, Passthrough, Manual , or an empty string (<code>""</code>).

Parameter	Description	Values
fips	Enable or disable FIPS mode. The default is false (disabled). If FIPS mode is enabled, the Red Hat Enterprise Linux CoreOS (RHCOS) machines that OpenShift Container Platform runs on bypass the default Kubernetes cryptography suite and use the cryptography modules that are provided with RHCOS instead.	false or true
imageContentSources	Sources and repositories for the release-image content.	Array of objects. Includes a source and, optionally, mirrors , as described in the following rows of this table.
imageContentSources.source	Required if you use imageContentSources . Specify the repository that users refer to, for example, in image pull specifications.	String
imageContentSources.mirrors	Specify one or more repositories that may also contain the same images.	Array of strings
networking	The configuration for the Pod network provider in the cluster.	Object
networking.clusterNetwork	The IP address pools for pods. The default is 10.128.0.0/14 with a host prefix of /23 .	Array of objects
networking.clusterNetwork.cidr	Required if you use networking.clusterNetwork . The IP block address pool.	IP network. IP networks are represented as strings using Classless Inter-Domain Routing (CIDR) notation with a traditional IP address or network number, followed by the forward slash (/) character, followed by a decimal value between 0 and 32 that describes the number of significant bits. For example, 10.0.0.0/16 represents IP addresses 10.0.0.0 through 10.0.255.255 .
networking.clusterNetwork.hostPrefix	Required if you use networking.clusterNetwork . The prefix size to allocate to each node from the CIDR. For example, 24 would allocate $2^8=256$ addresses to each node.	Integer
networking.machineNetwork	The IP address pools for machines.	Array of objects

Parameter	Description	Values
networking.machineNetwork.cidr	Required if you use networking.machineNetwork . The IP block address pool. The default is 10.0.0.0/16 for all platforms other than libvirt. For libvirt, the default is 192.168.126.0/24 .	IP network. IP networks are represented as strings using Classless Inter-Domain Routing (CIDR) notation with a traditional IP address or network number, followed by the forward slash (/) character, followed by a decimal value between 0 and 32 that describes the number of significant bits. For example, 10.0.0.0/16 represents IP addresses 10.0.0.0 through 10.0.255.255 .
networking.networkType	The type of network to install. The default is OpenShiftSDN .	String
networking.serviceNetwork	The IP address pools for services. The default is 172.30.0.0/16.	Array of IP networks. IP networks are represented as strings using Classless Inter-Domain Routing (CIDR) notation with a traditional IP address or network number, followed by the forward slash (/) character, followed by a decimal value between 0 and 32 that describes the number of significant bits. For example, 10.0.0.0/16 represents IP addresses 10.0.0.0 through 10.0.255.255 .
publish	How to publish or expose the user-facing endpoints of your cluster, such as the Kubernetes API, OpenShift routes.	Internal or External . To deploy a private cluster, which cannot be accessed from the internet, set publish to Internal . The default value is External .
sshKey	The SSH key or keys to authenticate access your cluster machines.  NOTE For production OpenShift Container Platform clusters on which you want to perform installation debugging or disaster recovery, specify an SSH key that your ssh-agent process uses.	One or more keys. For example:  sshKey: <key1> <key2> <key3>

Table 3.3. Additional Red Hat Virtualization (RHV) parameters for clusters

Parameter	Description	Values
platform.ovirt.ovirt_cluster_id	Required. The Cluster where the VMs will be created.	String. For example: 68833f9f-e89c-4891-b768-e2ba0815b76b
platform.ovirt.ovirt_storage_domain_id	Required. The Storage Domain ID where the VM disks will be created.	String. For example: ed7b0f4e-0e96-492a-8fff-279213ee1468
platform.ovirt.ovirt_network_name	Required. The network name where the VM nics will be created.	String. For example: ocpcluster
platform.ovirt.vnicProfileID	Required. The vNIC profile ID of the VM network interfaces. This can be inferred if the cluster network has a single profile.	String. For example: 3fa86930-0be5-4052-b667-b79f0a729692
platform.ovirt.api_vip	Required. An IP address on the machineNetwork that will be assigned to the API virtual IP (VIP). You can access the OpenShift API at this endpoint.	String. Example: 10.46.8.230
platform.ovirt.ingress_vip	Required. An IP address on the machineNetwork that will be assigned to the Ingress virtual IP (VIP).	String. Example: 10.46.8.232

Table 3.4. Additional RHV parameters for machine pools

Parameter	Description	Values
<machine-pool>.platform.ovirt.cpu	Optional. Defines the CPU of the VM.	Object
<machine-pool>.platform.ovirt.cpu.cores	Required if you use <machine-pool>.platform.ovirt.cpu . The number of cores. Total virtual CPUs (vCPUs) is cores * sockets.	Integer
<machine-pool>.platform.ovirt.cpu.sockets	Required if you use <machine-pool>.platform.ovirt.cpu . The number of sockets per core. Total virtual CPUs (vCPUs) is cores * sockets.	Integer
<machine-pool>.platform.ovirt.memoryMB	Optional. Memory of the VM in MiB.	Integer

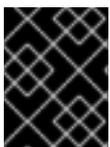
Parameter	Description	Values
<code><machine-pool>.platform.ovirt.instanceTypeID</code>	Optional. An instance type UUID, such as 00000009-0009-0009-0009-0000000000f1 , which you can get from the <a href="https://<engine-fqdn>/ovirt-engine/api/instancetypees">https://<engine-fqdn>/ovirt-engine/api/instancetypees endpoint.	String of UUID
<code><machine-pool>.platform.ovirt.osDisk</code>	Optional. Defines the first and bootable disk of the VM.	String
<code><machine-pool>.platform.ovirt.osDisk.sizeGB</code>	Required if you use <code><machine-pool>.platform.ovirt.osDisk</code> . Size of the disk in GiB.	Number
<code><machine-pool>.platform.ovirt.vmType</code>	Optional. The VM workload type, such as high-performance , server , or desktop .	String

**NOTE**

You can replace `<machine-pool>` with **controlPlane** or **compute**.

3.5.10. Deploying the cluster

You can install OpenShift Container Platform on a compatible cloud platform.

**IMPORTANT**

You can run the **create cluster** command of the installation program only once, during initial installation.

Prerequisites

- Obtain the OpenShift Container Platform installation program and the pull secret for your cluster.

Procedure

1. Run the installation program:

```
$ ./openshift-install create cluster --dir=<installation_directory> \ 1
--log-level=info 2
```

- 1** For `<installation_directory>`, specify the location of your customized `./install-config.yaml` file.

- 2 To view different installation details, specify **warn**, **debug**, or **error** instead of **info**.



NOTE

If the cloud provider account that you configured on your host does not have sufficient permissions to deploy the cluster, the installation process stops, and the missing permissions are displayed.

When the cluster deployment completes, directions for accessing your cluster, including a link to its web console and credentials for the **kubeadmin** user, display in your terminal.



IMPORTANT

The Ignition config files that the installation program generates contain certificates that expire after 24 hours. You must keep the cluster running for 24 hours in a non-degraded state to ensure that the first certificate rotation has finished.



IMPORTANT

You must not delete the installation program or the files that the installation program creates. Both are required to delete the cluster.

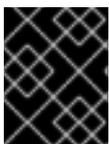


IMPORTANT

You have completed the steps required to install the cluster. The remaining steps show you how to verify the cluster and troubleshoot the installation.

3.5.11. Installing the CLI by downloading the binary

You can install the OpenShift CLI (**oc**) in order to interact with OpenShift Container Platform from a command-line interface. You can install **oc** on Linux, Windows, or macOS.



IMPORTANT

If you installed an earlier version of **oc**, you cannot use it to complete all of the commands in OpenShift Container Platform 4.5. Download and install the new version of **oc**.

3.5.11.1. Installing the CLI on Linux

You can install the OpenShift CLI (**oc**) binary on Linux by using the following procedure.

Procedure

1. Navigate to the [Infrastructure Provider](#) page on the Red Hat OpenShift Cluster Manager site.
2. Select your infrastructure provider, and, if applicable, your installation type.
3. In the **Command-line interface** section, select **Linux** from the drop-down menu and click **Download command-line tools**.
4. Unpack the archive:

-

```
$ tar xvzf <file>
```

5. Place the **oc** binary in a directory that is on your **PATH**. To check your **PATH**, execute the following command:

```
$ echo $PATH
```

After you install the CLI, it is available using the **oc** command:

```
$ oc <command>
```

3.5.11.2. Installing the CLI on Windows

You can install the OpenShift CLI (**oc**) binary on Windows by using the following procedure.

Procedure

1. Navigate to the [Infrastructure Provider](#) page on the Red Hat OpenShift Cluster Manager site.
2. Select your infrastructure provider, and, if applicable, your installation type.
3. In the **Command-line interface** section, select **Windows** from the drop-down menu and click **Download command-line tools**.
4. Unzip the archive with a ZIP program.
5. Move the **oc** binary to a directory that is on your **PATH**. To check your **PATH**, open the command prompt and execute the following command:

```
C:\> path
```

After you install the CLI, it is available using the **oc** command:

```
C:\> oc <command>
```

3.5.11.3. Installing the CLI on macOS

You can install the OpenShift CLI (**oc**) binary on macOS by using the following procedure.

Procedure

1. Navigate to the [Infrastructure Provider](#) page on the Red Hat OpenShift Cluster Manager site.
2. Select your infrastructure provider, and, if applicable, your installation type.
3. In the **Command-line interface** section, select **MacOS** from the drop-down menu and click **Download command-line tools**.
4. Unpack and unzip the archive.
5. Move the **oc** binary to a directory on your **PATH**. To check your **PATH**, open a terminal and execute the following command:

-

```
$ echo $PATH
```

After you install the CLI, it is available using the **oc** command:

```
$ oc <command>
```

3.5.12. Logging in to the cluster

You can log in to your cluster as a default system user by exporting the cluster **kubeconfig** file. The **kubeconfig** file contains information about the cluster that is used by the CLI to connect a client to the correct cluster and API server. The file is specific to a cluster and is created during OpenShift Container Platform installation.

Prerequisites

- Deploy an OpenShift Container Platform cluster.
- Install the **oc** CLI.

Procedure

1. Export the **kubeadmin** credentials:

```
$ export KUBECONFIG=<installation_directory>/auth/kubeconfig 1
```

- 1** For **<installation_directory>**, specify the path to the directory that you stored the installation files in.

2. Verify you can run **oc** commands successfully using the exported configuration:

```
$ oc whoami
```

Example output

```
system:admin
```

To learn more, see [Getting started with the CLI](#).

3.5.13. Verifying cluster status

You can verify your OpenShift Container Platform cluster's status during or after installation.

Procedure

1. In the cluster environment, export the administrator's kubeconfig file:

```
$ export KUBECONFIG=<installation_directory>/auth/kubeconfig 1
```

- 1** For **<installation_directory>**, specify the path to the directory that you stored the installation files in.

The **kubeconfig** file contains information about the cluster that is used by the CLI to connect a client to the correct cluster and API server.

2. View the control plane and compute machines created after a deployment:

```
$ oc get nodes
```

3. View your cluster's version:

```
$ oc get clusterversion
```

4. View your operators' status:

```
$ oc get clusteroperator
```

5. View all running pods in the cluster:

```
$ oc get pods -A
```

Troubleshooting

If the installation fails, the installation program times out and displays an error message. To learn more, see [Troubleshooting installation issues](#).

3.5.14. Accessing the OpenShift Container Platform web console on RHV

After the OpenShift Container Platform cluster initializes, you can log into the OpenShift Container Platform web console.

Procedure

1. Optional: In the Red Hat Virtualization (RHV) Administration Portal, open **Compute** → **Cluster**.
2. Verify that the installation program creates the virtual machines.
3. Return to the command line where the installation program is running. When the installation program finishes, it displays the user name and temporary password for logging into the OpenShift Container Platform web console.
4. In a browser, open the URL of the OpenShift Container Platform web console. The URL uses this format:

```
console-openshift-console.apps.<clustername>.<basedomain> 1
```

1 For **<clustername>.<basedomain>**, specify the cluster name and base domain.

For example:

```
console-openshift-console.apps.my-cluster.virtlab.example.com
```

3.5.15. Troubleshooting common issues with installing on Red Hat Virtualization (RHV)

Here are some common issues you might encounter, along with proposed causes and solutions.

3.5.15.1. CPU load increases and nodes go into a Not Ready state

- **Symptom:** CPU load increases significantly and nodes start going into a Not Ready state.
- **Cause:** The storage domain latency might be too high, especially for master nodes.
- **Solution:**
Make the nodes Ready again by restarting the kubelet service:

```
$ systemctl restart kubelet
```

Inspect the OpenShift Container Platform metrics service, which automatically gathers and reports on some valuable data such as the etcd disk sync duration. If the cluster is operational, use this data to help determine whether storage latency or throughput is the root issue. If so, consider using a storage resource that has lower latency and higher throughput.

To get raw metrics, enter the following command as kubeadmin or user with cluster-admin privileges:

```
$ oc get --insecure-skip-tls-verify --server=https://localhost:<port> --raw=/metrics
```

To learn more, see [Exploring Application Endpoints for the purposes of Debugging with OpenShift 4.x](#)

3.5.15.2. Trouble connecting the OpenShift Container Platform cluster API

- **Symptom:** The installation program completes but the OpenShift Container Platform cluster API is not available. The bootstrap virtual machine remains up after the bootstrap process is complete. When you enter the following command, the response will time out.

```
$ oc login -u kubeadmin -p *** <apiurl>
```

- **Cause:** The bootstrap VM was not deleted by the installation program and has not released the cluster's API IP address.
- **Solution:** Use the **wait-for** subcommand to be notified when the bootstrap process is complete:

```
$ ./openshift-install wait-for bootstrap-complete
```

When the bootstrap process is complete, delete the bootstrap virtual machine:

```
$ ./openshift-install destroy bootstrap
```

3.5.16. Post-installation tasks

After the OpenShift Container Platform cluster initializes, you can perform the following tasks.

- **Optional:** After deployment, add or replace SSH keys using the Machine Config Operator (MCO) in OpenShift Container Platform.

- Optional: Remove the **kubeadmin** user. Instead, use the authentication provider to create a user with cluster-admin privileges.

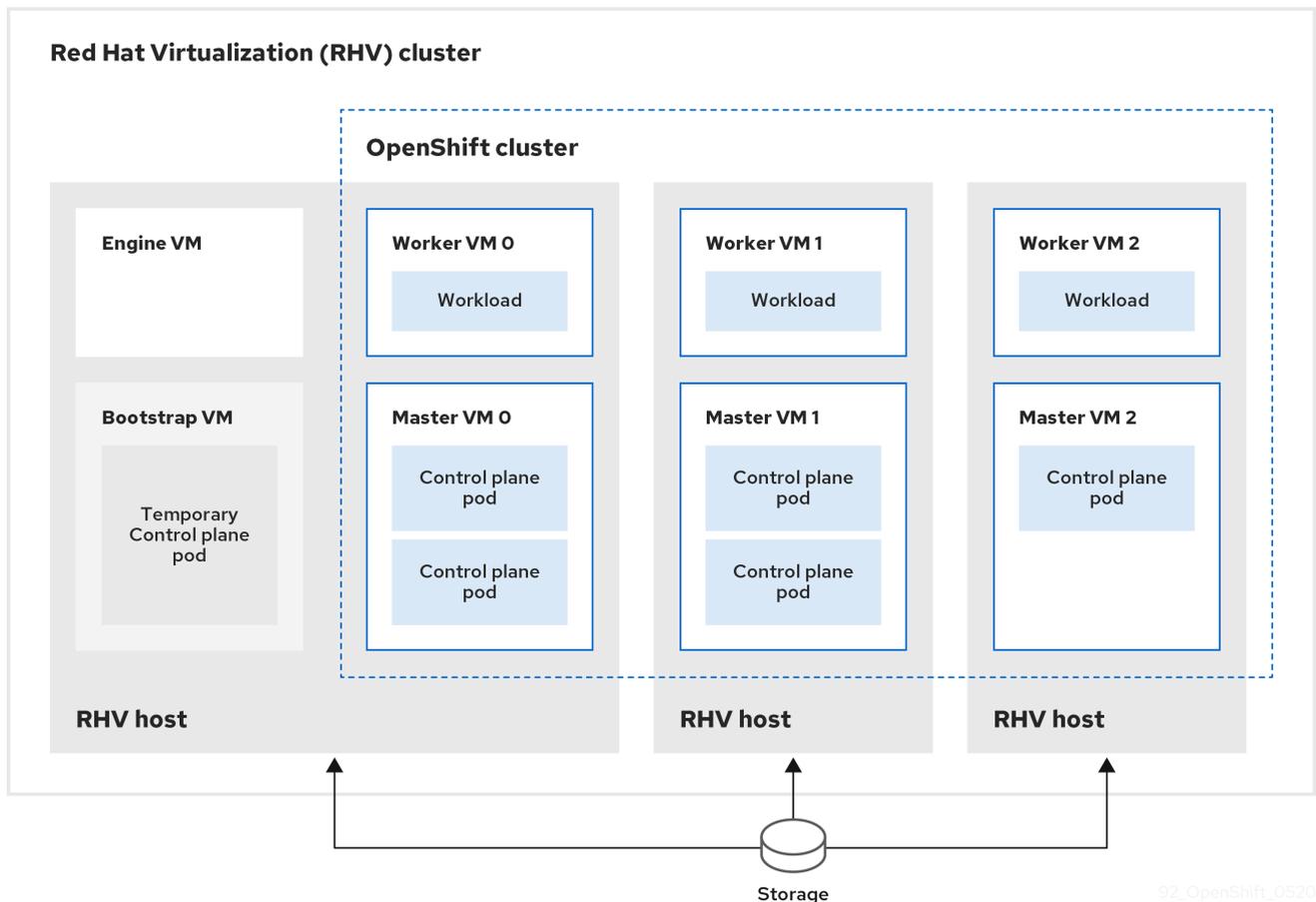
3.5.17. Next steps

- [Customize your cluster.](#)
- If necessary, you can [opt out of remote health reporting](#).

3.6. INSTALLING A CLUSTER ON RHV WITH USER-PROVISIONED INFRASTRUCTURE

In OpenShift Container Platform version 4.5, you can install a customized OpenShift Container Platform cluster on Red Hat Virtualization (RHV) and other infrastructure that you provide. The OpenShift Container Platform documentation uses the term *user-provisioned infrastructure* to refer to this infrastructure type.

The following diagram shows an example of a potential OpenShift Container Platform cluster running on a RHV cluster.



3.6.1. Prerequisites

The following items are required to install an OpenShift Container Platform cluster on a RHV environment.

- You have a supported combination of versions in the [Support Matrix for OpenShift Container Platform on RHV](#).

- You are familiar with the [OpenShift Container Platform installation and update](#) processes.

3.6.2. Internet and Telemetry access for OpenShift Container Platform

In OpenShift Container Platform 4.5, you require access to the internet to install your cluster. The Telemetry service, which runs by default to provide metrics about cluster health and the success of updates, also requires internet access. If your cluster is connected to the internet, Telemetry runs automatically, and your cluster is registered to the [Red Hat OpenShift Cluster Manager \(OCM\)](#).

Once you confirm that your Red Hat OpenShift Cluster Manager inventory is correct, either maintained automatically by Telemetry or manually using OCM, [use subscription watch](#) to track your OpenShift Container Platform subscriptions at the account or multi-cluster level.

You must have internet access to:

- Access the [Red Hat OpenShift Cluster Manager](#) page to download the installation program and perform subscription management. If the cluster has internet access and you do not disable Telemetry, that service automatically entitles your cluster.
- Access [Quay.io](#) to obtain the packages that are required to install your cluster.
- Obtain the packages that are required to perform cluster updates.



IMPORTANT

If your cluster cannot have direct internet access, you can perform a restricted network installation on some types of infrastructure that you provision. During that process, you download the content that is required and use it to populate a mirror registry with the packages that you need to install a cluster and generate the installation program. With some installation types, the environment that you install your cluster in will not require internet access. Before you update the cluster, you update the content of the mirror registry.

3.6.3. Requirements for the RHV environment

To install and run an OpenShift Container Platform cluster, the RHV environment must meet the following requirements.

Not meeting these requirements can cause the installation or process to fail. Additionally, not meeting these requirements can cause the OpenShift Container Platform cluster to fail days or weeks after installation.

The following requirements for CPU, memory, and storage resources are based on **default** values multiplied by the default number of virtual machines the installation program creates. These resources must be available **in addition to** what the RHV environment uses for non-OpenShift Container Platform operations.

By default, the installation program creates seven virtual machines during the installation process. First, it creates a bootstrap virtual machine to provide temporary services and a control plane while it creates the rest of the OpenShift Container Platform cluster. When the installation program finishes creating the cluster, deleting the bootstrap machine frees up its resources.

If you increase the number of virtual machines in the RHV environment, you must increase the resources accordingly.

Requirements

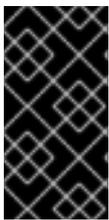
- The RHV environment has one data center whose state is **Up**.
- The RHV data center contains an RHV cluster.
- The RHV cluster has the following resources exclusively for the OpenShift Container Platform cluster:
 - Minimum 28 vCPUs: four for each of the seven virtual machines created during installation.
 - 112 GiB RAM or more, including:
 - 16 GiB or more for the bootstrap machine, which provides the temporary control plane.
 - 16 GiB or more for each of the three control plane machines which provide the control plane.
 - 16 GiB or more for each of the three compute machines, which run the application workloads.
- The RHV storage domain must meet [these etcd backend performance requirements](#).
- In production environments, each virtual machine must have 120 GiB or more. Therefore, the storage domain must provide 840 GiB or more for the default OpenShift Container Platform cluster. In resource-constrained or non-production environments, each virtual machine must have 32 GiB or more, so the storage domain must have 230 GiB or more for the default OpenShift Container Platform cluster.
- To download images from the Red Hat Ecosystem Catalog during installation and update procedures, the RHV cluster must have access to an internet connection. The Telemetry service also needs an internet connection to simplify the subscription and entitlement process.
- The RHV cluster must have a virtual network with access to the REST API on the RHV Manager.
- A user account and group with the following least privileges for installing and managing an OpenShift Container Platform cluster on the target RHV cluster:
 - **DiskOperator**
 - **DiskCreator**
 - **UserTemplateBasedVm**
 - **TemplateOwner**
 - **TemplateCreator**
 - **ClusterAdmin** on the target cluster

**WARNING**

Apply the principle of least privilege: Avoid using an administrator account with **SuperUser** privileges on RHV during the installation process. The installation program saves the credentials you provide to a temporary **ovirt-config.yaml** file that might be compromised.

3.6.4. Verifying the requirements for the RHV environment

Verify that the RHV environment meets the requirements to install and run an OpenShift Container Platform cluster. Not meeting these requirements can cause failures.

**IMPORTANT**

These requirements are based on the default resources the installation program uses to create control plane and compute machines. These resources include vCPUs, memory, and storage. If you change these resources or increase the number of OpenShift Container Platform machines, adjust these requirements accordingly.

Procedure

1. Check the RHV version.
 - a. In the RHV Administration Portal, click the ? help icon in the upper-right corner and select **About**.
 - b. In the window that opens, make a note of the **RHV Software Version**
 - c. Confirm that version 4.5 of OpenShift Container Platform and the version of RHV you noted are one of the supported combinations in the [Support Matrix for OpenShift Container Platform on RHV](#).
2. Inspect the data center, cluster, and storage.
 - a. In the RHV Administration Portal, click **Compute → Data Centers**.
 - b. Confirm that the data center where you plan to install OpenShift Container Platform is accessible.
 - c. Click the name of that data center.
 - d. In the data center details, on the **Storage** tab, confirm the storage domain where you plan to install OpenShift Container Platform is **Active**.
 - e. Record the **Domain Name** for use later on.
 - f. Confirm **Free Space** has at least 230 GiB.
 - g. Confirm that the storage domain meets [these etcd backend performance requirements](#), which you [can measure by using the fio performance benchmarking tool](#).
 - h. In the data center details, click the **Clusters** tab.

- i. Find the RHV cluster where you plan to install OpenShift Container Platform. Record the cluster name for use later on.
3. Inspect the RHV host resources.
 - a. In the RHV Administration Portal, click **Compute > Clusters**.
 - b. Click the cluster where you plan to install OpenShift Container Platform.
 - c. In the cluster details, click the **Hosts** tab.
 - d. Inspect the hosts and confirm they have a combined total of at least 28 **Logical CPU Cores** available *exclusively* for the OpenShift Container Platform cluster.
 - e. Record the number of available **Logical CPU Cores** for use later on.
 - f. Confirm that these CPU cores are distributed so that each of the seven virtual machines created during installation can have four cores.
 - g. Confirm that, all together, the hosts have 112 GiB of **Max free Memory for scheduling new virtual machines** distributed to meet the requirements for each of the following OpenShift Container Platform machines:
 - 16 GiB required for the bootstrap machine
 - 16 GiB required for each of the three control plane machines
 - 16 GiB for each of the three compute machines
 - h. Record the amount of **Max free Memory for scheduling new virtual machines** for use later on.
 4. Verify that the virtual network for installing OpenShift Container Platform has access to the RHV Manager's REST API. From a virtual machine on this network, use curl to reach the RHV Manager's REST API:

```
$ curl -k -u <username>@<profile>:<password> \ 1
https://<engine-fqdn>/ovirt-engine/api 2
```

- 1 For **<username>**, specify the user name of an RHV account with privileges to create and manage an OpenShift Container Platform cluster on RHV. For **<profile>**, specify the login profile, which you can get by going to the RHV Administration Portal login page and reviewing the **Profile** dropdown list. For **<password>**, specify the password for that user name.

- 2 For **<engine-fqdn>**, specify the fully qualified domain name of the RHV environment.

For example:

```
$ curl -k -u ovirtadmin@internal:pw123 \
https://rhv-env.virtlab.example.com/ovirt-engine/api
```

3.6.5. Network infrastructure configuration for installing OpenShift Container Platform on Red Hat Virtualization (RHV)

Before installing OpenShift Container Platform, configure your network environment to meet the following requirements.

When they boot, virtual machines must have IP addresses get the Ignition config files. Consider configuring DHCP to provide persistent IP addresses and hostnames to the cluster machines.

Firewall

Configure your firewall so your cluster has access to required sites.

Network connectivity

Configure your network to enable the following connections:

- Grant every machine access to every other machine on ports **30000-32767**. This provides connectivity to OpenShift Container Platform components.
- Grant every machine access to reserved ports **10250-10259** and **9000-9999**.
- Grant every machine access on ports **2379-2380**. This provides access to etcd, peers, and metrics on the control plane.
- Grant every machine access to the Kubernetes API on port **6443**.

Load balancers

Configure one or two (preferred) layer-4 load balancers:

- Provide load balancing for ports **6443** and **22623** on the control-plane and bootstrap machines. Port **6443** provides access to the Kubernetes API server and must be reachable both internally and externally. Port **22623** must be accessible to nodes within the cluster.
- Provide load balancing for port **443** and **80** for machines that run the ingress router (usually worker nodes in the default configuration). Both ports must be accessible from within and outside the cluster.

DNS

Configure infrastructure-provided DNS to allow the correct resolution of the main components and services. If you use only one load balancer, these DNS records can point to the same IP address.

- Create DNS records for **api.<cluster_name>.<base_domain>** (internal and external resolution) and **api-int.<cluster_name>.<base_domain>** (internal resolution) that point to the load balancer for the control plane machines.
- Create a DNS record for ***.apps.<cluster_name>.<base_domain>** that points to the load balancer for the ingress router (ports **443** and **80** of the compute machines).

3.6.6. Setting up the installation machine

To run the binary **openshift-install** installation program and Ansible scripts, set up the RHV Manager or an Red Hat Enterprise Linux (RHEL) computer with network access to the RHV environment and the REST API on the RHV Manager/oVirt Engine.

Procedure

1. Update or install Python3 and Ansible. For example:

```
# dnf update python3 ansible
```

2. Install the [python3-ovirt-engine-sdk4](#) package to get the Python Software Development Kit.
3. Install the **ovirt.image-template** Ansible role. On RHV Manager and other Red Hat Enterprise Linux (RHEL) machines, this role is distributed as the **ovirt-ansible-image-template** package. For example, enter:

```
# dnf install ovirt-ansible-image-template
```

4. Install the **ovirt.vm-infra** Ansible role. On RHV Manager and other RHEL machines, this role is distributed as the **ovirt-ansible-vm-infra** package.

```
# dnf install ovirt-ansible-vm-infra
```

5. Create an environment variable and assign an absolute or relative path to it. For example, enter:

```
$ export ASSETS_DIR=./wrk
```



NOTE

The installation program uses this variable to create a directory where it saves important installation-related files. Later, the installation process reuses this variable to locate those asset files. Avoid deleting this assets directory: It is required for uninstalling the cluster.

3.6.7. Setting up the CA certificate for RHV

Download the CA certificate from the Red Hat Virtualization (RHV) Manager and set it up on the installation machine.

You can download the certificate from a webpage on the RHV Manager or by using a **curl** command.

Later, you provide the certificate to the installation program.

Procedure

1. Use either of these two methods to download the CA certificate:

- Go to the Manager's webpage, <https://<engine-fqdn>/ovirt-engine/>. Then, under **Downloads**, click the **CA Certificate** link.
- Run the following command:

```
$ curl -k 'https://<engine-fqdn>/ovirt-engine/services/pki-resource?resource=ca-certificate&format=X509-PEM-CA' -o /tmp/ca.pem 1
```

- 1 For **<engine-fqdn>**, specify the fully qualified domain name of the RHV Manager, such as **rhv-env.virtlab.example.com**.

2. Configure the CA file to grant rootless user access to the Manager. Set the CA file permissions to have an octal value of **0644** (symbolic value: **-rw-r--r--**):

■

```
$ sudo chmod 0644 /tmp/ca.pem
```

- For Linux, copy the CA certificate to the directory for server certificates. Use **-p** to preserve the permissions:

```
$ sudo cp -p /tmp/ca.pem /etc/pki/ca-trust/source/anchors/ca.pem
```

- Add the certificate to the certificate manager for your operating system:
 - For macOS, double-click the certificate file and use the **Keychain Access** utility to add the file to the **System** keychain.
 - For Linux, update the CA trust:

```
$ sudo update-ca-trust
```



NOTE

If you use your own certificate authority, make sure the system trusts it.

Additional Resources

To learn more, see [Authentication and Security](#) in the RHV documentation.

3.6.8. Generating an SSH private key and adding it to the agent

If you want to perform installation debugging or disaster recovery on your cluster, you must provide an SSH key to both your **ssh-agent** and the installation program. You can use this key to access the bootstrap machine in a public cluster to troubleshoot installation issues.



NOTE

In a production environment, you require disaster recovery and debugging.

You can use this key to SSH into the master nodes as the user **core**. When you deploy the cluster, the key is added to the **core** user's `~/.ssh/authorized_keys` list.



NOTE

You must use a local key, not one that you configured with platform-specific approaches such as [AWS key pairs](#).

Procedure

- If you do not have an SSH key that is configured for password-less authentication on your computer, create one. For example, on a computer that uses a Linux operating system, run the following command:

```
$ ssh-keygen -t rsa -b 4096 -N "" \
  -f <path>/<file_name> 1
```

- Specify the path and file name, such as `~/.ssh/id_rsa`, of the new SSH key.

Running this command generates an SSH key that does not require a password in the location that you specified.

2. Start the **ssh-agent** process as a background task:

```
$ eval "$(ssh-agent -s)"
```

Example output

```
Agent pid 31874
```

3. Add your SSH private key to the **ssh-agent**:

```
$ ssh-add <path>/<file_name> 1
```

Example output

```
Identity added: /home/<you>/<path>/<file_name> (<computer_name>)
```

- 1** Specify the path and file name for your SSH private key, such as `~/.ssh/id_rsa`

Next steps

- When you install OpenShift Container Platform, provide the SSH public key to the installation program.

3.6.9. Obtaining the installation program

Before you install OpenShift Container Platform, download the installation file on a local computer.

Prerequisites

- A computer that runs Linux or macOS, with 500 MB of local disk space

Procedure

1. Access the [Infrastructure Provider](#) page on the Red Hat OpenShift Cluster Manager site. If you have a Red Hat account, log in with your credentials. If you do not, create an account.
2. Navigate to the page for your installation type, download the installation program for your operating system, and place the file in the directory where you will store the installation configuration files.



IMPORTANT

The installation program creates several files on the computer that you use to install your cluster. You must keep both the installation program and the files that the installation program creates after you finish installing the cluster.



IMPORTANT

Deleting the files created by the installation program does not remove your cluster, even if the cluster failed during installation. To remove your cluster, complete the OpenShift Container Platform uninstallation procedures for your specific cloud provider.

3. Extract the installation program. For example, on a computer that uses a Linux operating system, run the following command:

```
$ tar xvf openshift-install-linux.tar.gz
```

4. From the [Pull Secret](#) page on the Red Hat OpenShift Cluster Manager site, download your installation pull secret as a **.txt** file. This pull secret allows you to authenticate with the services that are provided by the included authorities, including Quay.io, which serves the container images for OpenShift Container Platform components.

3.6.10. Downloading the Ansible playbooks

Download the Ansible playbooks for installing OpenShift Container Platform version 4.5 on RHV.

Procedure

1. On your installation machine, run the following commands:

```
$ mkdir playbooks
$ cd playbooks
$ curl -L -X GET https://api.github.com/repos/openshift/installer/contents/upi/ovirt?
ref=$release-{product-version} |
grep 'download_url.*\.yml' |
awk '{ print $2 }' | sed -r 's/("|",)//g' |
xargs -n 1 curl -O
```

Next steps

- After you download these Ansible playbooks, you must also create the environment variable for the assets directory and customize the **inventory.yml** file before you create an installation configuration file by running the installation program.

3.6.11. The inventory.yml file

You use the **inventory.yml** file to define and create elements of the OpenShift Container Platform cluster you are installing. This includes elements such as the Red Hat Enterprise Linux CoreOS (RHCOS) image, virtual machine templates, bootstrap machine, control plane nodes, and worker nodes. You also use **inventory.yml** to destroy the cluster.

The following **inventory.yml** example shows you the parameters and their default values. The quantities and numbers in these default values meet the requirements for running a production OpenShift Container Platform cluster in a RHV environment.

Example inventory.yml file

```
---
```

```

all:
  vars:

    ovirt_cluster: "Default"
    ocp:
      assets_dir: "{{ lookup('env', 'ASSETS_DIR') }}"
      ovirt_config_path: "{{ lookup('env', 'HOME') }}/.ovirt/ovirt-config.yaml"

    # ---
    # {op-system} section
    # ---
    rhcos:
      image_url: "https://mirror.openshift.com/pub/openshift-v4/dependencies/rhcos/latest/latest/rhcos-
openstack.x86_64.qcow2.gz"
      local_cmp_image_path: "/tmp/rhcos.qcow2.gz"
      local_image_path: "/tmp/rhcos.qcow2"

    # ---
    # Profiles section
    # ---
    control_plane:
      cluster: "{{ ovirt_cluster }}"
      memory: 16GiB
      sockets: 4
      cores: 1
      template: rhcos_tpl
      operating_system: "rhcos_x64"
      type: high_performance
      graphical_console:
        headless_mode: false
      protocol:
        - spice
        - vnc
      disks:
        - size: 120GiB
          name: os
          interface: virtio_scsi
          storage_domain: depot_nvme
      nics:
        - name: nic1
          network: lab
          profile: lab

    compute:
      cluster: "{{ ovirt_cluster }}"
      memory: 16GiB
      sockets: 4
      cores: 1
      template: worker_rhcos_tpl
      operating_system: "rhcos_x64"
      type: high_performance
      graphical_console:
        headless_mode: false
      protocol:
        - spice
        - vnc

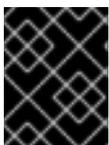
```

```

disks:
- size: 120GiB
  name: os
  interface: virtio_scsi
  storage_domain: depot_nvme
nics:
- name: nic1
  network: lab
  profile: lab

# ---
# Virtual machines section
# ---
vms:
- name: "{{ metadata.infraID }}-bootstrap"
  ocp_type: bootstrap
  profile: "{{ control_plane }}"
  type: server
- name: "{{ metadata.infraID }}-master0"
  ocp_type: master
  profile: "{{ control_plane }}"
- name: "{{ metadata.infraID }}-master1"
  ocp_type: master
  profile: "{{ control_plane }}"
- name: "{{ metadata.infraID }}-master2"
  ocp_type: master
  profile: "{{ control_plane }}"
- name: "{{ metadata.infraID }}-worker0"
  ocp_type: worker
  profile: "{{ compute }}"
- name: "{{ metadata.infraID }}-worker1"
  ocp_type: worker
  profile: "{{ compute }}"
- name: "{{ metadata.infraID }}-worker2"
  ocp_type: worker
  profile: "{{ compute }}"

```



IMPORTANT

Enter values for parameters whose descriptions begin with "Enter." Otherwise, you can use the default value or replace it with a new value.

General section

- **ovirt_cluster**: Enter the name of an existing RHV cluster in which to install the OCP cluster.
- **ocp.assets_dir**: The path of a directory the **openshift-install** installation program creates to store the files that it generates.
- **ocp.ovirt_config_path**: The path of the **ovirt-config.yaml** file the installation program generates, for example, **./wrk/install-config.yaml**. This file contains the credentials required to interact with the REST API of the oVirt Engine/RHV Manager.

Red Hat Enterprise Linux CoreOS (RHCOS) section

- **image_url**: Enter the URL of the RHCOS image you specified for download.
- **local_cmp_image_path**: The path of a local download directory for the compressed RHCOS image.
- **local_image_path**: The path of a local directory for the extracted RHCOS image.

Profiles section

This section consists of two profiles:

- **control_plane**: The profile of the bootstrap and control plane nodes.
- **compute**: The profile of workers nodes in the compute plane.

These profiles have the following parameters. The default values of the parameters meet the minimum requirements for running a production cluster. You can increase or customize these values to meet your workload requirements.

- **cluster**: The value gets the cluster name from **ovirt_cluster** in the General Section.
- **memory**: The amount of memory, in GB, for the virtual machine.
- **sockets**: The number of sockets for the virtual machine.
- **cores**: The number of cores for the virtual machine.
- **template**: The name of the virtual machine template. If plan to install multiple clusters, and these clusters use templates that contain different specifications, prepend the template name with the ID of the cluster.
- **operating_system**: The type of guest operating system in the virtual machine. With oVirt/RHV version 4.4, this value must be **rhcos_x64** so the value of **Ignition script** can be passed to the VM.
- **type**: Enter **server** as the type of the virtual machine.



IMPORTANT

You must change the value of the **type** parameter from **high_performance** to **server**.

- **disks**: The disk specifications. The **control_plane** and **compute** nodes can have different storage domains.
- **size**: The minimum disk size.
- **name**: Enter the name of a disk connected to the target cluster in RHV.
- **interface**: Enter the interface type of the disk you specified.
- **storage_domain**: Enter the storage domain of the disk you specified.
- **nics**: Enter the **name** and **network** the virtual machines use. You can also specify the virtual network interface profile. By default, NICs obtain their MAC addresses from the oVirt/RHV MAC pool.

Virtual machines section

This final section, **vms**, defines the virtual machines you plan to create and deploy in the cluster. By default, it provides the minimum number of control plane and worker nodes for a production environment.

vms contains three required elements:

- **name**: The name of the virtual machine. In this case, **metadata.infraID** prepends the virtual machine name with the infrastructure ID from the **metadata.yml** file.
- **ocp_type**: The role of the virtual machine in the OCP cluster. Possible values are **bootstrap**, **master**, **worker**.
- **profile**: The name of the profile from which each virtual machine inherits specifications. Possible values in this example are **control_plane** or **compute**.
You can override the value a virtual machine inherits from its profile. To do this, you add the name of the profile attribute to the virtual machine in **inventory.yml** and assign it an overriding value. To see an example of this, examine the **name: "{{ metadata.infraID }}-bootstrap"** virtual machine in the preceding **inventory.yml** example: It has a **type** attribute whose value, **server**, overrides the value of the **type** attribute this virtual machine would otherwise inherit from the **control_plane** profile.

Metadata variables

For virtual machines, **metadata.infraID** prepends the name of the virtual machine with the infrastructure ID from the **metadata.json** file you create when you build the ignition files.

The playbooks use the following code to read **infraID** from the specific file located in the **ocp.assets_dir**.

```
---
- name: include metadata.json vars
  include_vars:
    file: "{{ ocp.assets_dir }}/metadata.json"
    name: metadata
...
```

3.6.12. Specifying the RHCOS image settings

Update the Red Hat Enterprise Linux CoreOS (RHCOS) image settings of the **inventory.yml** file. Later, when you run this file one of the playbooks, it downloads a compressed Red Hat Enterprise Linux CoreOS (RHCOS) image from the **image_url** URL to the **local_cmp_image_path** directory. The playbook then uncompresses the image to the **local_image_path** directory and uses it to create oVirt/RHV templates.

Procedure

1. Locate the RHCOS image download page for the version of OpenShift Container Platform you are installing, such as [Index of /pub/openshift-v4/dependencies/rhcos/latest/latest](#).
2. From that download page, copy the URL of an **OpenStack** qcow2 image, such as **https://mirror.openshift.com/pub/openshift-v4/dependencies/rhcos/latest/latest/rhcos-openstack.x86_64.qcow2.gz**.

3. Edit the **inventory.yml** playbook you downloaded earlier. In it, paste the URL as the value for **image_url**. For example:

```
rhcos:
  "https://mirror.openshift.com/pub/openshift-v4/dependencies/rhcos/latest/latest/rhcos-
  openstack.x86_64.qcow2.gz"
```

3.6.13. Creating the install config file

You create an installation configuration file by running the installation program, **openshift-install**, and responding to its prompts with information you specified or gathered earlier.

When you finish responding to the prompts, the installation program creates an initial version of the **install-config.yaml** file in the assets directory you specified earlier, for example, **./wrk/install-config.yaml**

The installation program also creates a file, **\$HOME/.ovirt/ovirt-config.yaml**, that contains all the connection parameters that are required to reach the oVirt Engine/RHV Manager and use its REST API.

NOTE: The installation process does not use values you supply for some parameters, such as **Internal API virtual IP** and **Ingress virtual IP**, because you have already configured them in your infrastructure DNS.

It also uses the values you supply for parameters in **inventory.yml**, like the ones for **oVirt cluster**, **oVirt storage**, and **oVirt network**. And uses a script to remove or replace these same values from **install-config.yaml** with the previously mentioned **virtual IPs**.

Procedure

1. Run the installation program:

```
$ openshift-install create install-config --dir $ASSETS_DIR
```

2. Respond to the installation program's prompts with information about your system.

Example output

```
? SSH Public Key /home/user/.ssh/id_dsa.pub
? Platform <ovirt>
? Engine FQDN[:PORT] [? for help] <engine.fqdn>
? Enter ovirt-engine username <ovirtadmin@internal>
? Enter password <*****>
? oVirt cluster <cluster>
? oVirt storage <storage>
? oVirt network <net>
? Internal API virtual IP <172.16.0.252>
? Ingress virtual IP <172.16.0.251>
? Base Domain <example.org>
? Cluster Name <ocp4>
? Pull Secret [? for help] <*****>
```

For **Internal API virtual IP** and **Ingress virtual IP**, supply the IP addresses you specified when you configured the DNS service.

Together, the values you enter for the **oVirt cluster** and **Base Domain** prompts form the FQDN portion of URLs for the REST API and any applications you create, such as <https://api.ocp4.example.org:6443/> and <https://console-openshift-console.apps.ocp4.example.org>.

To get your **Pull secret**, visit <https://cloud.redhat.com/openshift/install/pull-secret>.

3.6.14. Customizing install-config.yaml

Here, you use three python scripts to override some of the installation program's default behaviors:

- By default, the installation program uses the machine API to create nodes. To override this default behavior, you set the number of compute nodes to zero replicas. Later, you use Ansible playbooks to create the compute nodes.
- By default, the installation program sets the IP range of the machine network for nodes. To override this default behavior, you set the IP range to match your infrastructure.
- By default, the installation program sets the platform to **ovirt**. However, installing a cluster on user-provisioned infrastructure is more similar to installing a cluster on bare metal. Therefore, you delete the ovirt platform section from **install-config.yaml** and change the platform to **none**. Instead, you use **inventory.yml** to specify all of the required settings.



NOTE

These snippets work with Python 3 and Python 2.

Procedure

1. Set the number of compute nodes to zero replicas:

```
$ python3 -c 'import os, yaml
path = "%s/install-config.yaml" % os.environ["ASSETS_DIR"]
conf = yaml.safe_load(open(path))
conf["compute"][0]["replicas"] = 0
open(path, "w").write(yaml.dump(conf, default_flow_style=False))'
```

2. Set the IP range of the machine network. For example, to set the range to **172.16.0.0/16**, enter:

```
$ python3 -c 'import os, yaml
path = "%s/install-config.yaml" % os.environ["ASSETS_DIR"]
conf = yaml.safe_load(open(path))
conf["networking"]["machineNetwork"][0]["cidr"] = "172.16.0.0/16"
open(path, "w").write(yaml.dump(conf, default_flow_style=False))'
```

3. Remove the **ovirt** section and change the platform to **none**:

```
$ python3 -c 'import os, yaml
path = "%s/install-config.yaml" % os.environ["ASSETS_DIR"]
conf = yaml.safe_load(open(path))
platform = conf["platform"]
del platform["ovirt"]
platform["none"] = {}
open(path, "w").write(yaml.dump(conf, default_flow_style=False))'
```

3.6.15. Generate manifest files

Use the installation program to generate a set of manifest files in the assets directory.

The command to generate the manifest files displays a warning message before it consumes the **install-config.yaml** file.

If you plan to reuse the **install-config.yaml** file, create a backup copy of it before you back it up before you generate the manifest files.

Procedure

1. Optional: Create a backup copy of the **install-config.yaml** file:

```
$ cp install-config.yaml install-config.yaml.backup
```

2. Generate a set of manifests in your assets directory:

```
$ openshift-install create manifests --dir $ASSETS_DIR
```

This command displays the following messages.

Example output

```
INFO Consuming Install Config from target directory
WARNING Making control-plane schedulable by setting MastersSchedulable to true for
Scheduler cluster settings
```

The command generates the following manifest files:

Example output

```
$ tree
.
├── wrk
│   └── manifests
│       ├── 04-openshift-machine-config-operator.yaml
│       ├── cluster-config.yaml
│       ├── cluster-dns-02-config.yml
│       ├── cluster-infrastructure-02-config.yml
│       ├── cluster-ingress-02-config.yml
│       ├── cluster-network-01-crd.yml
│       ├── cluster-network-02-config.yml
│       ├── cluster-proxy-01-config.yaml
│       ├── cluster-scheduler-02-config.yml
│       ├── cvo-overrides.yaml
│       ├── etcd-ca-bundle-configmap.yaml
│       ├── etcd-client-secret.yaml
│       ├── etcd-host-service-endpoints.yaml
│       ├── etcd-host-service.yaml
│       ├── etcd-metric-client-secret.yaml
│       ├── etcd-metric-serving-ca-configmap.yaml
│       ├── etcd-metric-signer-secret.yaml
│       ├── etcd-namespace.yaml
│       └── etcd-service.yaml
```


**NOTE**

Certificates in Ignition configuration files expire after 24 hours. Complete the cluster installation and keep the cluster running in a non-degraded state for 24 hours so that the first certificate rotation can finish.

Procedure

1. To build the ignition files, enter:

```
$ openshift-install create ignition-configs --dir $ASSETS_DIR
```

Example output

```
$ tree
.
├── wrk
│   ├── auth
│   │   ├── kubeadmin-password
│   │   └── kubeconfig
│   ├── bootstrap.ign
│   ├── master.ign
│   ├── metadata.json
│   └── worker.ign
```

3.6.18. Creating templates and virtual machines

After confirming the variables in the **inventory.yml**, you run the first Ansible provisioning playbook, **create-templates-and-vms.yml**.

This playbook uses the connection parameters for the RHV Manager from **\$HOME/.ovirt/ovirt-config.yaml** and reads **metadata.json** in the assets directory.

If a local Red Hat Enterprise Linux CoreOS (RHCOS) image is not already present, the playbook downloads one from the URL you specified for **image_url** in **inventory.yml**. It extracts the image and uploads it to RHV to create templates.

The playbook creates a template based on the **control_plane** and **compute** profiles in the **inventory.yml**. If these profiles have different names, it creates two templates.

When the playbook finishes, the virtual machines it creates are stopped. You can get information from them to help configure other infrastructure elements. For example, you can get the virtual machines' MAC addresses to configure DHCP to assign permanent IP addresses to the virtual machines.

Procedure

1. In **inventory.yml**, under the **control_plane** and **compute** variables, change both instances of **type: high_performance** to **type: server**.
2. Optional: If you plan to perform multiple installations to the same cluster, create different templates for each OCP installation. In **inventory.yml**, prepend the value of **template** with **infraID**. For example:

```
control_plane:
  cluster: "{{ ovirt_cluster }}"
```

```
memory: 16GiB
sockets: 4
cores: 1
template: "{{ metadata.infraID }}-rhcos_tpl"
operating_system: "rhcos_x64"
...
```

3. Create the templates and virtual machines:

```
$ ansible-playbook -i inventory.yml create-templates-and-vms.yml
```

3.6.19. Creating the bootstrap machine

You create a bootstrap machine by running the **bootstrap.yml** playbook. This playbook starts the bootstrap virtual machine, and passes it the **bootstrap.ign** Ignition file from the assets directory. The bootstrap node configures itself so it can serve Ignition files to the control plane nodes.

To monitor the bootstrap process, you use the console in the RHV Administration Portal or connect to the virtual machine by using SSH.

Procedure

1. Create the bootstrap machine:

```
$ ansible-playbook -i inventory.yml bootstrap.yml
```

2. Connect to the bootstrap machine using a console in the Administration Portal or SSH. Replace **<bootstrap_ip>** with the bootstrap node's IP address. To use SSH, enter:

```
$ ssh core@<bootstrap.ip>
```

3. Collect **bootkube.service** journald unit logs for the release image service from the bootstrap node:

```
[core@ocp4-1k6b4-bootstrap ~]$ journalctl -b -f -u release-image.service -u bootkube.service
```



NOTE

The **bootkube.service** log on the bootstrap node outputs **etcd connection refused** errors, indicating that the bootstrap server is unable to connect to etcd on master nodes. After etcd has started on each master node and the nodes have joined the cluster, the errors should stop.

3.6.20. Creating the control plane nodes

You create the control plane nodes by running the **masters.yml** playbook. This playbook passes the **master.ign** Ignition file to each of the virtual machines. The Ignition file contains a directive for the control plane node to get the Ignition from a URL such as <https://api-int.ocp4.example.org:22623/config/master>. The port number in this URL is managed by the load balancer, and is accessible only inside the cluster.

Procedure

1. Create the control plane nodes:

```
$ ansible-playbook -i inventory.yml masters.yml
```

2. While the playbook creates your control plane, monitor the bootstrapping process:

```
$ openshift-install wait-for bootstrap-complete --dir $ASSETS_DIR
```

Example output

```
INFO API v1.18.3+b74c5ed up  
INFO Waiting up to 40m0s for bootstrapping to complete...
```

3. When all the pods on the control plane nodes and etcd are up and running, the installation program displays the following output.

Example output

```
INFO It is now safe to remove the bootstrap resources
```

3.6.21. Verifying cluster status

You can verify your OpenShift Container Platform cluster's status during or after installation.

Procedure

1. In the cluster environment, export the administrator's kubeconfig file:

```
$ export KUBECONFIG=$ASSETS_DIR/auth/kubeconfig
```

The **kubeconfig** file contains information about the cluster that is used by the CLI to connect a client to the correct cluster and API server.

2. View the control plane and compute machines created after a deployment:

```
$ oc get nodes
```

3. View your cluster's version:

```
$ oc get clusterversion
```

4. View your operators' status:

```
$ oc get clusteroperator
```

5. View all running pods in the cluster:

```
$ oc get pods -A
```

3.6.22. Removing the bootstrap machine

After the **wait-for** command shows that the bootstrap process is complete, you must remove the bootstrap virtual machine to free up compute, memory, and storage resources. Also, remove settings for the bootstrap machine from the load balancer directives.

Procedure

1. To remove the bootstrap machine from the cluster, enter:

```
$ ansible-playbook -i inventory.yml retire-bootstrap.yml
```

2. Remove settings for the bootstrap machine from the load balancer directives.

3.6.23. Creating the worker nodes and completing the installation

Creating worker nodes is similar to creating control plane nodes. However, worker nodes workers do not automatically join the cluster. To add them to the cluster, you review and approve the workers' pending CSRs (Certificate Signing Requests).

After approving the first requests, you continue approving CSR until all of the worker nodes are approved. When you complete this process, the worker nodes become **Ready** and can have pods scheduled to run on them.

Finally, monitor the command line to see when the installation process completes.

Procedure

1. Create the worker nodes:

```
$ ansible-playbook -i inventory.yml workers.yml
```

2. To list all of the CSRs, enter:

```
$ oc get csr -A
```

Eventually, this command displays one CSR per node. For example:

Example output

```
NAME          AGE   SIGNERNAME                                REQUESTOR
CONDITION
csr-2lnxd     63m   kubernetes.io/kubelet-serving             system:node:ocp4-lk6b4-
master0.ocp4.example.org                 Approved,Issued
csr-hff4q     64m   kubernetes.io/kube-apiserver-client-kubelet
system:serviceaccount:openshift-machine-config-operator:node-bootstrapper
Approved,Issued
csr-hsn96     60m   kubernetes.io/kubelet-serving             system:node:ocp4-lk6b4-
master2.ocp4.example.org                 Approved,Issued
csr-m724n     6m2s  kubernetes.io/kube-apiserver-client-kubelet
system:serviceaccount:openshift-machine-config-operator:node-bootstrapper Pending
csr-p4dz2     60m   kubernetes.io/kube-apiserver-client-kubelet
system:serviceaccount:openshift-machine-config-operator:node-bootstrapper
Approved,Issued
csr-t9vfj     60m   kubernetes.io/kubelet-serving             system:node:ocp4-lk6b4-
master1.ocp4.example.org                 Approved,Issued
```

```
csr-tggtr 61m kubernetes.io/kube-apiserver-client-kubelet
system:serviceaccount:openshift-machine-config-operator:node-bootstrapper
Approved,Issued
csr-wcbrf 7m6s kubernetes.io/kube-apiserver-client-kubelet
system:serviceaccount:openshift-machine-config-operator:node-bootstrapper Pending
```

- To filter the list and see only pending CSRs, enter:

```
$ watch "oc get csr -A | grep pending -i"
```

This command refreshes the output every two seconds and displays only pending CSRs. For example:

Example output

```
Every 2.0s: oc get csr -A | grep pending -i

csr-m724n 10m kubernetes.io/kube-apiserver-client-kubelet
system:serviceaccount:openshift-machine-config-operator:node-bootstrapper Pending
csr-wcbrf 11m kubernetes.io/kube-apiserver-client-kubelet
system:serviceaccount:openshift-machine-config-operator:node-bootstrapper Pending
```

- Inspect each pending request. For example:

Example output

```
$ oc describe csr csr-m724n
```

Example output

```
Name:          csr-m724n
Labels:        <none>
Annotations:   <none>
CreationTimestamp: Sun, 19 Jul 2020 15:59:37 +0200
Requesting User: system:serviceaccount:openshift-machine-config-operator:node-
bootstrapper
Signer:        kubernetes.io/kube-apiserver-client-kubelet
Status:        Pending
Subject:
  Common Name:  system:node:ocp4-lk6b4-worker1.ocp4.example.org
  Serial Number:
  Organization: system:nodes
Events: <none>
```

- If the CSR information is correct, approve the request:

```
$ oc adm certificate approve csr-m724n
```

- Wait for the installation process to finish:

```
$ openshift-install wait-for install-complete --dir $ASSETS_DIR --log-level debug
```

When the installation completes, the command line displays the URL of the OpenShift Container Platform web console and the administrator user name and password.

3.7. UNINSTALLING A CLUSTER ON RHV

You can remove an OpenShift Container Platform cluster from Red Hat Virtualization (RHV).

3.7.1. Removing a cluster that uses installer-provisioned infrastructure

You can remove a cluster that uses installer-provisioned infrastructure from your cloud.

Prerequisites

- Have a copy of the installation program that you used to deploy the cluster.
- Have the files that the installation program generated when you created your cluster.

Procedure

1. From the computer that you used to install the cluster, run the following command:

```
$ ./openshift-install destroy cluster \
--dir=<installation_directory> --log-level=info 1 2
```

- 1** For **<installation_directory>**, specify the path to the directory that you stored the installation files in.
- 2** To view different details, specify **warn**, **debug**, or **error** instead of **info**.



NOTE

You must specify the directory that contains the cluster definition files for your cluster. The installation program requires the **metadata.json** file in this directory to delete the cluster.

2. Optional: Delete the **<installation_directory>** directory and the OpenShift Container Platform installation program.

3.7.2. Removing a cluster that uses user-provisioned infrastructure

When you are finished using the cluster, You can remove a cluster that uses user-provisioned infrastructure from your cloud.

Prerequisites

- Have the original playbook files, assets directory and files, and **\$ASSETS_DIR** environment variable that you used to you install the cluster. Typically, you can achieve this by using the same computer you used when you installed the cluster.

Procedure

1. To remove the cluster, enter:

```
$ ansible-playbook -i inventory.yml \  
  retire-bootstrap.yml \  
  retire-masters.yml \  
  retire-workers.yml
```

2. Remove any configurations you added to DNS, load balancers, and any other infrastructure for this cluster.