OpenShift Container Platform 4.6

Installing on RHV

Installing OpenShift Container Platform RHV clusters
Installing OpenShift Container Platform RHV clusters
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 [RH-VIRTUALIZATION]  4

  1.1.1. Prerequisites  4

  1.1.2. Internet access for OpenShift Container Platform  5

  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

  1.1.10. Installing the OpenShift CLI by downloading the binary  15

    1.1.10.1. Installing the OpenShift CLI on Linux  15

    1.1.10.2. Installing the OpenShift CLI on Windows  16

    1.1.10.3. Installing the OpenShift CLI on macOS  16

  1.1.11. Logging in to the cluster by using the CLI  17

  1.1.12. Verifying cluster status  17

  1.1.13. Accessing the OpenShift Container Platform web console on RHV  18

  1.1.14. Telemetry access for OpenShift Container Platform  19

  1.1.15. Troubleshooting common issues with installing on Red Hat Virtualization (RHV)  19

    1.1.15.1. CPU load increases and nodes go into a Not Ready state  19

    1.1.15.2. Trouble connecting the OpenShift Container Platform cluster API  19

  1.1.16. Post-installation tasks  20

1.2. INSTALLING A CLUSTER ON RHV WITH CUSTOMIZATIONS  20

  1.2.1. Prerequisites  21

  1.2.2. Internet access for OpenShift Container Platform  21

  1.2.3. Requirements for the RHV environment  22

  1.2.4. Verifying the requirements for the RHV environment  23

  1.2.5. Preparing the network environment on RHV  25

  1.2.6. Setting up the CA certificate for RHV  26

  1.2.7. Generating an SSH private key and adding it to the agent  27

  1.2.8. Obtaining the installation program  28

  1.2.9. Creating the installation configuration file  29

    1.2.9.1. Example install-config.yaml files for Red Hat Virtualization (RHV)  31

    1.2.9.2. Installation configuration parameters  33

      1.2.9.2.1. Required configuration parameters  33

      1.2.9.2.2. Network configuration parameters  35

      1.2.9.2.3. Optional configuration parameters  36

      1.2.9.2.4. Additional Red Hat Virtualization (RHV) configuration parameters  40

      1.2.9.2.5. Additional RHV parameters for machine pools  41

  1.2.10. Deploying the cluster  42

  1.2.11. Installing the OpenShift CLI by downloading the binary  44

    1.2.11.1. Installing the OpenShift CLI on Linux  44

    1.2.11.2. Installing the OpenShift CLI on Windows  44

    1.2.11.3. Installing the OpenShift CLI on macOS  45

  1.2.12. Logging in to the cluster by using the CLI  45

  1.2.13. Verifying cluster status  46

  1.2.14. Accessing the OpenShift Container Platform web console on RHV  47

  1.2.15. Telemetry access for OpenShift Container Platform  47

  1.2.16. Troubleshooting common issues with installing on Red Hat Virtualization (RHV)  47

    1.2.16.1. CPU load increases and nodes go into a Not Ready state  48
1.2.16.2. Trouble connecting the OpenShift Container Platform cluster API
1.2.17. Post-installation tasks
1.2.18. Next steps

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

1.3.1. Prerequisites
1.3.2. Internet access for OpenShift Container Platform
1.3.3. Requirements for the RHV environment
1.3.4. Verifying the requirements for the RHV environment
1.3.5. Networking requirements for user-provisioned infrastructure
   Network topology requirements
   Load balancers
   NTP configuration
1.3.6. Setting up the installation machine
1.3.7. Setting up the CA certificate for RHV
1.3.8. Generating an SSH private key and adding it to the agent
1.3.9. Obtaining the installation program
1.3.10. Downloading the Ansible playbooks
1.3.11. The inventory.yml file
1.3.12. Specifying the RHCOS image settings
1.3.13. Creating the install config file
1.3.14. Customizing install-config.yaml
1.3.15. Generate manifest files
1.3.16. Making control-plane nodes non-schedulable
1.3.17. Building the Ignition files
1.3.18. Creating templates and virtual machines
1.3.19. Creating the bootstrap machine
1.3.20. Creating the control plane nodes
1.3.21. Verifying cluster status
1.3.22. Removing the bootstrap machine
1.3.23. Creating the worker nodes and completing the installation
1.3.24. Telemetry access for OpenShift Container Platform

1.4. UNINSTALLING A CLUSTER ON RHV

1.4.1. Removing a cluster that uses installer-provisioned infrastructure
1.4.2. Removing a cluster that uses user-provisioned infrastructure
CHAPTER 1. INSTALLING ON RHV

1.1. INSTALLING A CLUSTER QUICKLY ON {RH-VIRTUALIZATION}

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.
- You have a supported combination of versions in the Support Matrix for OpenShift Container Platform on Red Hat Virtualization (RHV).
- If you use a firewall, configure it to allow the sites that your cluster requires access to.
1.1.2. Internet access for OpenShift Container Platform

In OpenShift Container Platform 4.6, you require access to the Internet to install your cluster.

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 the 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. Ensure that DHCP is enabled on this network, because the VMs that the installer creates obtain their IP address by using DHCP.

- 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 4.6 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
```

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.

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

For example:

```
$ curl -k -u ocpadmin@internal:pw123 \ 1
  https://rhv-env.virtlab.example.com/ovirt-engine/api 2
```

1.1.5. Preparing the network environment on RHV

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

Procedure

1. Reserve two static IP addresses

   a. On the network where you plan to install OpenShift Container Platform, identify two 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 Address Resolution Protocol (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 two 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:

   for `<cluster-name>`, `<base-domain>`, and `<ip-address>`, specify the cluster name, base domain, and static IP address of your OpenShift Container Platform API.

   Specify the cluster name, base domain, and static IP address of your OpenShift Container Platform apps for Ingress and the load balancer.

   For example:

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

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:

     ```
     ```

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

   ```bash
   $ 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:

     ```bash
     $ 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**

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:

   ```bash
   $ ssh-keygen -t ed25519 -N "" \
   -f <path>/<file_name> ①
   ```

   ① Specify the path and file name, such as `~/.ssh/id_rsa`, of the new SSH key. If you have an existing key pair, ensure your public key is in the your `~/.ssh` directory.

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

If you plan to install an OpenShift Container Platform cluster that uses FIPS Validated / Modules in Process cryptographic libraries on the x86_64 architecture, do not create a key that uses the ed25519 algorithm. Instead, create a key that uses the rsa or ecdsa algorithm.

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

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

Example output

   Agent pid 31874

NOTE

If your cluster is in FIPS mode, only use FIPS-compliant algorithms to generate the SSH key. The key must be either RSA or ECDSA.

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

   $ ssh-add /path/to/your/private/key

Example output

   Identity added: /home/you/path/to/your/private/key (computer_name)

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

- You have 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. Select your infrastructure provider.
3. 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 the installation program and the files that the installation program creates after you finish installing the cluster. Both files are required to delete 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.

4. 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
```

5. 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**

- Open the `ovirt-imageio` port to the Manager from the machine running the installer. By default, the port is 54322.

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

**Procedure**

1. Change to the directory that contains the installation program and initialize the cluster deployment:

```
$ ./openshift-install create cluster --dir <installation_directory> --log-level=info
```
For `<installation_directory>`, specify the directory name to store the files that the installation program creates.

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 **Engine FQDN[:PORT]**, enter the fully qualified domain name (FQDN) of the RHV environment.

   For example:

   ```
   rhv-env.virtlab.example.com:443
   ```

d. The installer automatically generates a CA certificate. For **Would you like to use the above certificate to connect to the Manager?**, answer `y` or `N`. If you answer `N`, you must install OpenShift Container Platform in insecure mode.

e. For **Engine username**, enter the user name and profile of the RHV administrator using this format:

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

   **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. For example: `admin@internal`.

f. For **Engine password**, enter the RHV admin password.

g. For **Cluster**, select the RHV cluster for installing OpenShift Container Platform.
h. For Storage domain, select the storage domain for installing OpenShift Container Platform.

i. For Network, select a virtual network that has access to the RHV Manager REST API.

j. For Internal API Virtual IP, enter the static IP address you set aside for the cluster’s REST API.

k. For Ingress virtual IP, enter the static IP address you reserved for the wildcard apps domain.

l. 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

m. 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.

n. 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.

**Example output**

```
... INFO Install complete!
INFO To access the cluster as the system:admin user when using 'oc', run 'export KUBECONFIG=/home/myuser/install_dir/auth/kubeconfig'
INFO Access the OpenShift web-console here: https://console-openshift-console.apps.mycluster.example.com
INFO Login to the console with user: "kubeadmin", and password: "4vYBz-Ee6gm-ymBZj-Wt5AL"
INFO Time elapsed: 36m22s
```

**NOTE**

The cluster access and credential information also outputs to `<installation_directory>/.openshift_install.log` when an installation succeeds.
IMPORTANT

- The Ignition config files that the installation program generates contain certificates that expire after 24 hours, which are then renewed at that time. If the cluster is shut down before renewing the certificates and the cluster is later restarted after the 24 hours have elapsed, the cluster automatically recovers the expired certificates. The exception is that you must manually approve the pending node-bootstrapper certificate signing requests (CSRs) to recover kubelet certificates. See the documentation for Recovering from expired control plane certificates for more information.

- It is recommended that you use Ignition config files within 12 hours after they are generated because the 24-hour certificate rotates from 16 to 22 hours after the cluster is installed. By using the Ignition config files within 12 hours, you can avoid installation failure if the certificate update runs during installation.

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.

1.1.10. Installing the OpenShift 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 4.6. Download and install the new version of oc.

1.1.10.1. Installing the OpenShift CLI on Linux

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

Procedure


2. Select the appropriate version in the Version drop-down menu.

3. Click Download Now next to the OpenShift v4.6 Linux Client entry and save the file.

4. Unpack the archive:

   $ tar xzvf <file>
5. Place the `oc` binary in a directory that is on your `PATH`. To check your `PATH`, execute the following command:

```bash
$ echo $PATH
```

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

```bash
$ oc <command>
```

### 1.1.10.2. Installing the OpenShift CLI on Windows

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

**Procedure**

2. Select the appropriate version in the `Version` drop-down menu.
3. Click **Download Now** next to the **OpenShift v4.6 Windows Client** entry and save the file.
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:

```cmd
C:\> path
```

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

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

### 1.1.10.3. Installing the OpenShift CLI on macOS

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

**Procedure**

2. Select the appropriate version in the `Version` drop-down menu.
3. Click **Download Now** next to the **OpenShift v4.6 MacOSX Client** entry and save the file.
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:

```bash
$ echo $PATH
```

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

```bash
$ oc <command>
```
After you install the OpenShift CLI, it is available using the `oc` command:

```
$ oc <command>
```

To learn more, see Getting started with the OpenShift CLI.

### 1.1.11. Logging in to the cluster by using the CLI

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**

- You deployed an OpenShift Container Platform cluster.
- You installed the `oc` CLI.

**Procedure**

1. Export the `kubeadmin` credentials:

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

   **Example output**

   ```
   system:admin
   ```

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

```
$ oc whoami
```

**Additional resources**

- See Accessing the web console for more details about accessing and understanding the OpenShift Container Platform web console.

### 1.1.12. 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
```
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.

### 1.1.13. Accessing the OpenShift Container Platform web console on RHV

After the OpenShift Container Platform cluster initializes, you can log in to 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>
   ```

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

   For example:

   ```
   console-openshift-console.apps.my-cluster.virtlab.example.com
   ```
1.1.14. Telemetry access for OpenShift Container Platform

In OpenShift Container Platform 4.6, the Telemetry service, which runs by default to provide metrics about cluster health and the success of updates, 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).

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

Additional resources

- See About remote health monitoring for more information about the Telemetry service

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

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

1.1.15.1. 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 control plane nodes (also known as the 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:

  ```
  ```

  To learn more, see Exploring Application Endpoints for the purposes of Debugging with OpenShift 4.x

1.1.15.2. 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
```

### 1.16. 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.

### 1.2. 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.
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.

### 1.2.1. Prerequisites

- Review details about the OpenShift Container Platform installation and update processes.

- You have a supported combination of versions in the Support Matrix for OpenShift Container Platform on Red Hat Virtualization (RHV).

- If you use a firewall, configure it to allow the sites that your cluster requires access to.

### 1.2.2. Internet access for OpenShift Container Platform

In OpenShift Container Platform 4.6, you require access to the Internet to install your cluster.

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.
1.2.3. 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. Ensure that DHCP is enabled on this network, because the VMs that the installer creates obtain their IP address by using DHCP.

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.

1.2.4. 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 4.6 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
```
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.

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

For example:

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

1.2.5. Preparing the network environment on RHV

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

**Procedure**

1. Reserve two static IP addresses
   - a. On the network where you plan to install OpenShift Container Platform, identify two 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 Address Resolution Protocol (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 two 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 for Ingress and the load balancer.
For example:

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

1.2.6. 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:

     ```bash
     ```

     ![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--`):

   ```bash
   $ 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:

   ```bash
   $ 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:

     ```bash
     $ 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.

1.2.7. 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:

   ```bash
   $ ssh-keygen -t ed25519 -N "" \
   -f <path>/<file_name>
   ```

   Specify the path and file name, such as ~/.ssh/id_rsa, of the new SSH key. If you have an existing key pair, ensure your public key is in the your ~/.ssh directory.

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

   **NOTE**

   If you plan to install an OpenShift Container Platform cluster that uses FIPS Validated / Modules in Process cryptographic libraries on the x86_64 architecture, do not create a key that uses the ed25519 algorithm. Instead, create a key that uses the rsa or ecdsa algorithm.

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

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

   **Example output**

   Agent pid 31874

   **NOTE**

   If your cluster is in FIPS mode, only use FIPS-compliant algorithms to generate the SSH key. The key must be either RSA or ECDSA.
3. Add your SSH private key to the **ssh-agent**:

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

**Example output**

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

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.

1.2.8. **Obtaining the installation program**

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

**Prerequisites**

- You have 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. Select your infrastructure provider.

3. 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 the installation program and the files that the installation program creates after you finish installing the cluster. Both files are required to delete 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.

4. 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  
   ```
5. 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.

### 1.2.9. 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. Change to the directory that contains the installation program and run the following command:

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

   **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:

   ```text
   https://<engine-fqdn>/ovirt-engine/api
   ```
1. For `<engine-fqdn>`, specify the fully qualified domain name of the RHV environment.

For example:

```sh
curl -k -u ocpadmin@internal:pw123 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:

```bash
<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:

```bash
ocpadmin@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 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 **Ingress virtual IP**, enter the static IP address you reserved for the wildcard apps domain.

xiii. 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

xiv. 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.
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.

1.2.9.1. 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

```yaml
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:
```
Example: A minimal `install-config.yaml` file

```yaml
apiVersion: v1
baseDomain: example.com
metadata:
  name: test-cluster
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
  pullSecret: '{"auths": ...}'
  sshKey: ssh-ed12345 AAAA...
```

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

```yaml
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
```

```yaml
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
  pullSecret: '{"auths": ...}'
  sshKey: ssh-ed12345 AAAA...
```
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-8ff-279213ee1468
    ovirt_network_name: ovirtmgmt
    vnicProfileID: 3fa86930-0be5-4052-b667-b79f0a729692
    pullSecret: '{"auths": ...}'
  sshKey: ssh-ed25519 AAAA...

1.2.9.2. 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.

**IMPORTANT**

The `openshift-install` command does not validate field names for parameters. If an incorrect name is specified, the related file or object is not created, and no error is reported. Ensure that the field names for any parameters that are specified are correct.

1.2.9.2.1. Required configuration parameters

Required installation configuration parameters are described in the following table:

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
<th>Values</th>
</tr>
</thead>
<tbody>
<tr>
<td>apiVersion</td>
<td>The API version for the <code>install-config.yaml</code> content. The current version is <code>v1</code>. The installer may also support older API versions.</td>
<td>String</td>
</tr>
</tbody>
</table>
### Table of Parameters

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
<th>Values</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>baseDomain</strong></td>
<td>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 <code>baseDomain</code> and <code>metadata.name</code> parameter values that uses the <code>metadata.name</code>. <code>baseDomain</code> format.</td>
<td>A fully-qualified domain or subdomain name, such as example.com.</td>
</tr>
<tr>
<td><strong>metadata</strong></td>
<td>Kubernetes resource <code>ObjectMeta</code>, from which only the <code>name</code> parameter is consumed.</td>
<td>Object</td>
</tr>
<tr>
<td><strong>metadata.name</strong></td>
<td>The name of the cluster. DNS records for the cluster are all subdomains of <code>{{.metadata.name}}.{{.baseDomain}}</code>.</td>
<td>String of lowercase letters, hyphens (-), and periods (.), such as dev.</td>
</tr>
<tr>
<td><strong>platform</strong></td>
<td>The configuration for the specific platform upon which to perform the installation: <code>aws, baremetal, azure, openstack, ovirt, vsphere</code>. For additional information about <code>platform.&lt;platform&gt;</code> parameters, consult the following table for your specific platform.</td>
<td>Object</td>
</tr>
</tbody>
</table>
**pullSecret**

Get a pull secret from [https://console.redhat.com/openshift/install/pull-secret](https://console.redhat.com/openshift/install/pull-secret) to authenticate downloading container images for OpenShift Container Platform components from services such as Quay.io.

```json
{
  "auths": {
    "cloud.openshift.com": {
      "auth": "b3Blb=",
      "email": "you@example.com"
    },
    "quay.io": {
      "auth": "b3Blb=",
      "email": "you@example.com"
    }
  }
}
```

### 1.2.9.2.2. Network configuration parameters

You can customize your installation configuration based on the requirements of your existing network infrastructure. For example, you can expand the IP address block for the cluster network or provide different IP address blocks than the defaults.

Only IPv4 addresses are supported.

#### Table 1.2. Network parameters

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
<th>Values</th>
</tr>
</thead>
<tbody>
<tr>
<td>networking</td>
<td>The configuration for the cluster network.</td>
<td>Object</td>
</tr>
<tr>
<td>networking.clusterNetwork</td>
<td>The IP address blocks for pods. The default value is <strong>10.128.0.0/14</strong> with a host prefix of <strong>/23</strong>. If you specify multiple IP address blocks, the blocks must not overlap.</td>
<td>An array of objects. For example: <code>{ networking: clusterNetwork: - cidr: 10.128.0.0/14 hostPrefix: 23 }</code></td>
</tr>
<tr>
<td>networking.network.Type</td>
<td>The cluster network provider Container Network Interface (CNI) plug-in to install.</td>
<td>Either <strong>OpenShiftSDN</strong> or <strong>OVNKubernetes</strong>. The default value is <strong>OpenShiftSDN</strong>.</td>
</tr>
</tbody>
</table>

**NOTE**

You cannot modify parameters specified by the `networking` object after installation.
<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
<th>Values</th>
</tr>
</thead>
<tbody>
<tr>
<td>networking.clusterNetwork.cidr</td>
<td>Required if you use networking.clusterNetwork. An IP address block.</td>
<td>An IP address block in Classless Inter-Domain Routing (CIDR) notation. The prefix length for an IPv4 block is between 0 and 32.</td>
</tr>
<tr>
<td></td>
<td>An IPv4 network.</td>
<td></td>
</tr>
<tr>
<td>networking.clusterNetwork.hostPrefix</td>
<td>The subnet prefix length to assign to each individual node. For example,</td>
<td>A subnet prefix.</td>
</tr>
<tr>
<td></td>
<td>if hostPrefix is set to 23 then each node is assigned a /23 subnet out of</td>
<td>The default value is 23.</td>
</tr>
<tr>
<td></td>
<td>the given cidr. A hostPrefix value of 23 provides 510 (2^(32 - 23) - 2) pod</td>
<td></td>
</tr>
<tr>
<td></td>
<td>IP addresses.</td>
<td></td>
</tr>
<tr>
<td>networking.serviceNetwork</td>
<td>The IP address block for services. The default value is 172.30.0.0/16.</td>
<td>An array with an IP address block in CIDR format. For example:</td>
</tr>
</tbody>
</table>
|                                 | The OpenShift SDN and OVN-Kubernetes network providers support only a single  | networking:
|                                 | IP address block for the service network.                                    | serviceNetwork:
|                                 |                                                                           | - 172.30.0.0/16                                                       |
| networking.machineNetwork       | The IP address blocks for machines.                                          | An array of objects. For example:                                     |
|                                 | If you specify multiple IP address blocks, the blocks must not overlap.       | networking:
|                                 |                                                                           | machineNetwork:
|                                 |                                                                           | - cidr: 10.0.0.0/16                                                   |
| networking.machineNetwork.cidr  | Required if you use networking.machineNetwork. An IP address block.           | An IP network block in CIDR notation.                                 |
|                                 | The default value is 10.0.0.0/16 for all platforms other than libvirt. For    | For example, 10.0.0.0/16.                                             |
|                                 | libvirt, the default value is 192.168.126.0/24.                               |                                                                      |
|                                 |                                                                           | **NOTE**                                                              |
|                                 |                                                                           | Set the networking.machineNetwork to match the CIDR that the preferred NIC resides in. |

1.2.9.2.3. Optional configuration parameters

Optional installation configuration parameters are described in the following table:

Table 1.3. Optional parameters
<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
<th>Values</th>
</tr>
</thead>
<tbody>
<tr>
<td>additionalTrustBundle</td>
<td>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.</td>
<td>String</td>
</tr>
<tr>
<td>compute</td>
<td>The configuration for the machines that comprise the compute nodes.</td>
<td>Array of machine-pool objects. For details, see the following “Machine-pool” table.</td>
</tr>
<tr>
<td>compute.architecture</td>
<td>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 <code>amd64</code> (the default).</td>
<td>String</td>
</tr>
<tr>
<td>compute.hyperthreading</td>
<td>Whether to enable or disable simultaneous multithreading, or <code>hyperthreading</code>, on compute machines. By default, simultaneous multithreading is enabled to increase the performance of your machines’ cores.</td>
<td>Enabled or Disabled</td>
</tr>
<tr>
<td></td>
<td><strong>IMPORTANT</strong> If you disable simultaneous multithreading, ensure that your capacity planning accounts for the dramatically decreased machine performance.</td>
<td></td>
</tr>
<tr>
<td>compute.name</td>
<td>Required if you use <code>compute</code>. The name of the machine pool.</td>
<td>worker</td>
</tr>
<tr>
<td>compute.platform</td>
<td>Required if you use <code>compute</code>. Use this parameter to specify the cloud provider to host the worker machines. This parameter value must match the <code>controlPlane.platform</code> parameter value.</td>
<td><code>aws</code>, <code>azure</code>, <code>gcp</code>, <code>openstack</code>, <code>ovirt</code>, <code>vsphere</code>, or <code>{}</code></td>
</tr>
<tr>
<td>compute.replicas</td>
<td>The number of compute machines, which are also known as worker machines, to provision.</td>
<td>A positive integer greater than or equal to 2. The default value is 3.</td>
</tr>
<tr>
<td>Parameter</td>
<td>Description</td>
<td>Values</td>
</tr>
<tr>
<td>-------------------------------</td>
<td>-----------------------------------------------------------------------------</td>
<td>---------------------------------------</td>
</tr>
<tr>
<td>controlPlane</td>
<td>The configuration for the machines that comprise the control plane.</td>
<td>Array of <strong>MachinePool</strong> objects. For details, see the following &quot;Machine-pool&quot; table.</td>
</tr>
<tr>
<td>controlPlane.architecture</td>
<td>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 <strong>amd64</strong> (the default).</td>
<td>String</td>
</tr>
<tr>
<td>controlPlane.hypertreading</td>
<td>Whether to enable or disable simultaneous multithreading, or <strong>hypertreading</strong>, on control plane machines. By default, simultaneous multithreading is enabled to increase the performance of your machines' cores. <strong>IMPORTANT</strong> If you disable simultaneous multithreading, ensure that your capacity planning accounts for the dramatically decreased machine performance.</td>
<td><strong>Enabled</strong> or <strong>Disabled</strong></td>
</tr>
<tr>
<td>controlPlane.name</td>
<td>Required if you use <strong>controlPlane</strong>. The name of the machine pool.</td>
<td><strong>master</strong></td>
</tr>
<tr>
<td>controlPlane.platform</td>
<td>Required if you use <strong>controlPlane</strong>. Use this parameter to specify the cloud provider that hosts the control plane machines. This parameter value must match the <strong>compute.platform</strong> parameter value.</td>
<td><strong>aws, azure, gcp, openstack, ovirt, vsphere, or{}</strong></td>
</tr>
<tr>
<td>controlPlane.replicas</td>
<td>The number of control plane machines to provision.</td>
<td>The only supported value is <strong>3</strong>, which is the default value.</td>
</tr>
<tr>
<td>Parameter</td>
<td>Description</td>
<td>Values</td>
</tr>
<tr>
<td>------------------------</td>
<td>-----------------------------------------------------------------------------</td>
<td>---------------------------------------------</td>
</tr>
<tr>
<td>credentialsMode</td>
<td>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.</td>
<td>Mint, Passthrough, Manual, or an empty string (“”).</td>
</tr>
<tr>
<td></td>
<td><strong>NOTE</strong></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Not all CCO modes are supported for all cloud providers. For more information on CCO modes, see the Cloud Credential Operator entry in the Red Hat Operators reference content.</td>
<td></td>
</tr>
<tr>
<td>fips</td>
<td>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.</td>
<td>false or true</td>
</tr>
<tr>
<td></td>
<td><strong>IMPORTANT</strong></td>
<td></td>
</tr>
<tr>
<td></td>
<td>The use of FIPS Validated / Modules in Process cryptographic libraries is only supported on OpenShift Container Platform deployments on the x86_64 architecture.</td>
<td></td>
</tr>
<tr>
<td></td>
<td><strong>NOTE</strong></td>
<td></td>
</tr>
<tr>
<td></td>
<td>If you are using Azure File storage, you cannot enable FIPS mode.</td>
<td></td>
</tr>
<tr>
<td>imageContentSources</td>
<td>Sources and repositories for the release-image content.</td>
<td>Array of objects. Includes a source and, optionally, mirrors, as described in the following rows of this table.</td>
</tr>
</tbody>
</table>
### Parameter Description Values

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
<th>Values</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>imageContentSources.source</code></td>
<td>Required if you use <code>imageContentSources</code>. Specify the repository that users refer to, for example, in image pull specifications.</td>
<td>String</td>
</tr>
<tr>
<td><code>imageContentSources.mirrors</code></td>
<td>Specify one or more repositories that may also contain the same images.</td>
<td>Array of strings</td>
</tr>
<tr>
<td><code>publish</code></td>
<td>How to publish or expose the user-facing endpoints of your cluster, such as the Kubernetes API, OpenShift routes.</td>
<td><strong>Internal</strong> or <strong>External</strong>. The default value is <strong>External</strong>. Setting this field to <strong>Internal</strong> is not supported on non-cloud platforms. IMPORTANT If the value of the field is set to <strong>Internal</strong>, the cluster will become non-functional. For more information, refer to <a href="https://issues.redhat.com/browse/BZ1953035">BZ#1953035</a>.</td>
</tr>
<tr>
<td><code>sshKey</code></td>
<td>The SSH key or keys to authenticate access your cluster machines.</td>
<td>One or more keys. For example:</td>
</tr>
<tr>
<td></td>
<td><strong>NOTE</strong></td>
<td></td>
</tr>
<tr>
<td></td>
<td>For production OpenShift Container Platform clusters on which you want to perform installation debugging or disaster recovery, specify an SSH key that your <code>ssh-agent</code> process uses.</td>
<td></td>
</tr>
</tbody>
</table>

### 1.2.9.2.4. Additional Red Hat Virtualization (RHV) configuration parameters

Additional RHV configuration parameters are described in the following table:

#### Table 1.4. Additional RHV parameters for clusters
<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
<th>Values</th>
</tr>
</thead>
<tbody>
<tr>
<td>platform.ovirt.ovirt_cluster_id</td>
<td>Required. The Cluster where the VMs will be created.</td>
<td>String. For example: 68833f9f-e89c-4891-b768-e2ba0815b76b</td>
</tr>
<tr>
<td>platform.ovirt.ovirt_storage_domain_id</td>
<td>Required. The Storage Domain ID where the VM disks will be created.</td>
<td>String. For example: ed7b0f4e-0e96-492a-8fff-279213ee1468</td>
</tr>
<tr>
<td>platform.ovirt.ovirt_network_name</td>
<td>Required. The network name where the VM nics will be created.</td>
<td>String. For example: ocpcluster</td>
</tr>
<tr>
<td>platform.ovirt.vnicProfileID</td>
<td>Required. The vNIC profile ID of the VM network interfaces. This can be</td>
<td>String. For example: 3fa86930-0be5-4052-b667-b79f0a729692</td>
</tr>
<tr>
<td></td>
<td>inferred if the cluster network has a single profile.</td>
<td></td>
</tr>
<tr>
<td>platform.ovirt.api_vip</td>
<td>Required. An IP address on the machine network that will be assigned to the</td>
<td>String. Example: 10.46.8.230</td>
</tr>
<tr>
<td></td>
<td>API virtual IP (VIP). You can access the OpenShift API at this endpoint.</td>
<td></td>
</tr>
<tr>
<td>platform.ovirt.ingress_vip</td>
<td>Required. An IP address on the machine network that will be assigned to the</td>
<td>String. Example: 10.46.8.232</td>
</tr>
<tr>
<td></td>
<td>Ingress virtual IP (VIP).</td>
<td></td>
</tr>
</tbody>
</table>

1.2.9.2.5. Additional RHV parameters for machine pools

Additional RHV configuration parameters for machine pools are described in the following table:

Table 1.5. Additional RHV parameters for machine pools

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
<th>Values</th>
</tr>
</thead>
<tbody>
<tr>
<td>&lt;machine-pool&gt;.platform.ovirt.cpu</td>
<td>Optional. Defines the CPU of the VM.</td>
<td>Object</td>
</tr>
<tr>
<td>&lt;machine-pool&gt;.platform.ovirt.cpu.cores</td>
<td>Required if you use &lt;machine-pool&gt;.platform.ovirt.cpu. The number of cores. Total virtual</td>
<td>Integer</td>
</tr>
<tr>
<td></td>
<td>CPUs (vCPUs) is cores * sockets.</td>
<td></td>
</tr>
<tr>
<td>&lt;machine-pool&gt;.platform.ovirt.cpusockets</td>
<td>Required if you use &lt;machine-pool&gt;.platform.ovirt.cpu. The number of sockets per core. Total</td>
<td>Integer</td>
</tr>
</tbody>
</table>
### Parameter Description Values

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
<th>Values</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>&lt;machine-pool&gt;.platform.ovirt.memoryMB</code></td>
<td>Optional. Memory of the VM in MiB.</td>
<td>Integer</td>
</tr>
<tr>
<td><code>&lt;machine-pool&gt;.platform.ovirt.instanceTypeID</code></td>
<td>Optional. An instance type UUID, such as <code>00000009-0009-0009-0009-0000000000f1</code>, which you can get from the <code>https://&lt;engine-fqdn&gt;/ovirt-engine/api/instancetypes</code> endpoint.</td>
<td>String of UUID</td>
</tr>
<tr>
<td><code>&lt;machine-pool&gt;.platform.ovirt.osDisk</code></td>
<td>Optional. Defines the first and bootable disk of the VM.</td>
<td>String</td>
</tr>
<tr>
<td><code>&lt;machine-pool&gt;.platform.ovirt.osDisk.sizeGB</code></td>
<td>Required if you use <code>&lt;machine-pool&gt;.platform.ovirt.osDisk</code>. Size of the disk in GiB.</td>
<td>Number</td>
</tr>
<tr>
<td><code>&lt;machine-pool&gt;.platform.ovirt.vmType</code></td>
<td>Optional. The VM workload type, such as <code>high-performance</code>, <code>server</code>, or <code>desktop</code>.</td>
<td>String</td>
</tr>
</tbody>
</table>

**NOTE**

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

### 1.2.10. 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**

- Open the `ovirt-imageio` port to the Manager from the machine running the installer. By default, the port is `54322`.
- Obtain the OpenShift Container Platform installation program and the pull secret for your cluster.

**Procedure**
1. Change to the directory that contains the installation program and initialize the cluster deployment:

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

   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.

   **Example output**

   
   INFO Install complete!
   INFO To access the cluster as the system:admin user when using 'oc', run 'export KUBECONFIG=/home/myuser/install_dir/auth/kubeconfig'
   INFO Access the OpenShift web-console here: https://console-openshift-console.apps.mycluster.example.com
   INFO Login to the console with user: "kubeadmin", and password: "4vYBz-Ee6gm-ymBZj-Wt5AL"
   INFO Time elapsed: 36m22s

   **NOTE**

   The cluster access and credential information also outputs to `<installation_directory>/openshift_install.log` when an installation succeeds.

   **IMPORTANT**

   - The Ignition config files that the installation program generates contain certificates that expire after 24 hours, which are then renewed at that time. If the cluster is shut down before renewing the certificates and the cluster is later restarted after the 24 hours have elapsed, the cluster automatically recovers the expired certificates. The exception is that you must manually approve the pending `node-bootstrapper` certificate signing requests (CSRs) to recover kubelet certificates. See the documentation for Recovering from expired control plane certificates for more information.

   - It is recommended that you use Ignition config files within 12 hours after they are generated because the 24-hour certificate rotates from 16 to 22 hours after the cluster is installed. By using the Ignition config files within 12 hours, you can avoid installation failure if the certificate update runs during installation.
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.

1.2.11. Installing the OpenShift 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 4.6. Download and install the new version of oc.

1.2.11.1. Installing the OpenShift CLI on Linux

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

Procedure


2. Select the appropriate version in the Version drop-down menu.

3. Click Download Now next to the OpenShift v4.6 Linux Client entry and save the file.

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 OpenShift CLI, it is available using the oc command:

   $ oc <command>

1.2.11.2. Installing the OpenShift CLI on Windows

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

Procedure

2. Select the appropriate version in the Version drop-down menu.

3. Click Download Now next to the OpenShift v4.6 Windows Client entry and save the file.

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 OpenShift CLI, it is available using the oc command:

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

1.2.11.3. Installing the OpenShift CLI on macOS

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

Procedure


2. Select the appropriate version in the Version drop-down menu.

3. Click Download Now next to the OpenShift v4.6 MacOSX Client entry and save the file.

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 OpenShift CLI, it is available using the oc command:

   ```
   $ oc <command>
   ```

1.2.12. Logging in to the cluster by using the CLI

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

- You deployed an OpenShift Container Platform cluster.
• You installed the oc CLI.

Procedure

1. Export the kubeadmin credentials:

   $ export KUBECONFIG=<installation_directory>/auth/kubeconfig

   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 OpenShift CLI.

1.2.13. 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

   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:
Troubleshooting

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

1.2.14. Accessing the OpenShift Container Platform web console on RHV

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

Procedure


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>`

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

   For example:

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

1.2.15. Telemetry access for OpenShift Container Platform

In OpenShift Container Platform 4.6, the Telemetry service, which runs by default to provide metrics about cluster health and the success of updates, 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).

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

Additional resources

- See About remote health monitoring for more information about the Telemetry service

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

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

```sh
$ oc get pods -A
```
1.2.16.1. 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 control plane nodes (also known as the 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 kubadmin or user with cluster-admin privileges:

```
```

To learn more, see Exploring Application Endpoints for the purposes of Debugging with OpenShift 4.x

1.2.16.2. 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
```

1.2.17. 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.
1.2.18. Next steps

- Customize your cluster.
- If necessary, you can opt out of remote health reporting.

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

In OpenShift Container Platform version 4.6, 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.

![Diagram of OpenShift Container Platform cluster on RHV]

The RHV hosts run virtual machines that contain both control plane and compute pods. One of the hosts also runs a Manager virtual machine and a bootstrap virtual machine that contains a temporary control plane pod.

1.3.1. 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 Red Hat Virtualization (RHV).

You are familiar with the OpenShift Container Platform installation and update processes.

### 1.3.2. Internet access for OpenShift Container Platform

In OpenShift Container Platform 4.6, you require access to the Internet to install your cluster.

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.

### 1.3.3. 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. Ensure that DHCP is enabled on this network, because the VMs that the installer creates obtain their IP address by using DHCP.

- 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.
1.3.4. 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 4.6 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 ocpadmin@internal:pw123 \  
   https://rhv-env.virtlab.example.com/ovirt-engine/api

1.3.5. Networking requirements for user-provisioned infrastructure

All the Red Hat Enterprise Linux CoreOS (RHCOS) machines require network in initramfs during boot to fetch Ignition config from the machine config server.

During the initial boot, the machines require either a DHCP server or that static IP addresses be set on each host in the cluster in order to establish a network connection, which allows them to download their Ignition config files.

It is recommended to use the DHCP server to manage the machines for the cluster long-term. Ensure that the DHCP server is configured to provide persistent IP addresses and host names to the cluster machines.

The Kubernetes API server must be able to resolve the node names of the cluster machines. If the API
servers and worker nodes are in different zones, you can configure a default DNS search zone to allow the API server to resolve the node names. Another supported approach is to always refer to hosts by their fully-qualified domain names in both the node objects and all DNS requests.

You must configure the network connectivity between machines to allow cluster components to communicate. Each machine must be able to resolve the host names of all other machines in the cluster.

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

See also:
- Red Hat Virtualization Manager firewall requirements
- Host firewall requirements

Load balancers
Configure one or preferably two 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, which are usually compute 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. For example, ports 443 and 80 of the compute machines.

Table 1.6. All machines to all machines

<table>
<thead>
<tr>
<th>Protocol</th>
<th>Port</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ICMP</td>
<td>N/A</td>
<td>Network reachability tests</td>
</tr>
<tr>
<td>TCP</td>
<td>1936</td>
<td>Metrics</td>
</tr>
<tr>
<td></td>
<td>9000-9999</td>
<td>Host level services, including the node exporter on ports 9100-9101 and the Cluster Version Operator on port 9099.</td>
</tr>
<tr>
<td></td>
<td>10250-10259</td>
<td>The default ports that Kubernetes reserves</td>
</tr>
<tr>
<td></td>
<td>10256</td>
<td>openshift-sdn</td>
</tr>
<tr>
<td>Protocol</td>
<td>Port</td>
<td>Description</td>
</tr>
<tr>
<td>----------</td>
<td>--------</td>
<td>--------------------------------------------------</td>
</tr>
<tr>
<td>UDP</td>
<td>4789</td>
<td>VXLAN and Geneve</td>
</tr>
<tr>
<td></td>
<td>6081</td>
<td>VXLAN and Geneve</td>
</tr>
<tr>
<td></td>
<td>9000-9999</td>
<td>Host level services, including the node exporter on ports 9100-9101.</td>
</tr>
<tr>
<td>TCP/UDP</td>
<td>30000-32767</td>
<td>Kubernetes node port</td>
</tr>
</tbody>
</table>

Table 1.7. All machines to control plane

<table>
<thead>
<tr>
<th>Protocol</th>
<th>Port</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>TCP</td>
<td>6443</td>
<td>Kubernetes API</td>
</tr>
</tbody>
</table>

Table 1.8. Control plane machines to control plane machines

<table>
<thead>
<tr>
<th>Protocol</th>
<th>Port</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>TCP</td>
<td>2379-2380</td>
<td>etcd server and peer ports</td>
</tr>
</tbody>
</table>

Network topology requirements
The infrastructure that you provision for your cluster must meet the following network topology requirements.

**IMPORTANT**

OpenShift Container Platform requires all nodes to have internet access to pull images for platform containers and provide telemetry data to Red Hat.

Load balancers
Before you install OpenShift Container Platform, you must provision two load balancers that meet the following requirements:

1. **API load balancer**. Provides a common endpoint for users, both human and machine, to interact with and configure the platform. Configure the following conditions:
   - Layer 4 load balancing only. This can be referred to as Raw TCP, SSL Passthrough, or SSL Bridge mode. If you use SSL Bridge mode, you must enable Server Name Indication (SNI) for the API routes.
   - A stateless load balancing algorithm. The options vary based on the load balancer implementation.

**IMPORTANT**

Do not configure session persistence for an API load balancer.
Configure the following ports on both the front and back of the load balancers:

### Table 1.9. API load balancer

<table>
<thead>
<tr>
<th>Port</th>
<th>Back-end machines (pool members)</th>
<th>Internal</th>
<th>External</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>6443</td>
<td>Bootstrap and control plane. You remove the bootstrap machine from the load balancer after the bootstrap machine initializes the cluster control plane. You must configure the /readyz endpoint for the API server health check probe.</td>
<td>X</td>
<td>X</td>
<td>Kubernetes API server</td>
</tr>
<tr>
<td>22623</td>
<td>Bootstrap and control plane. You remove the bootstrap machine from the load balancer after the bootstrap machine initializes the cluster control plane.</td>
<td>X</td>
<td></td>
<td>Machine config server</td>
</tr>
</tbody>
</table>

**NOTE**

The load balancer must be configured to take a maximum of 30 seconds from the time the API server turns off the /readyz endpoint to the removal of the API server instance from the pool. Within the time frame after /readyz returns an error or becomes healthy, the endpoint must have been removed or added. Probing every 5 or 10 seconds, with two successful requests to become healthy and three to become unhealthy, are well-tested values.

2. **Application Ingress load balancer**: Provides an Ingress point for application traffic flowing in from outside the cluster. Configure the following conditions:

- Layer 4 load balancing only. This can be referred to as Raw TCP, SSL Passthrough, or SSL Bridge mode. If you use SSL Bridge mode, you must enable Server Name Indication (SNI) for the Ingress routes.

- A connection-based or session-based persistence is recommended, based on the options available and types of applications that will be hosted on the platform.

Configure the following ports on both the front and back of the load balancers:

### Table 1.10. Application Ingress load balancer

<table>
<thead>
<tr>
<th>Port</th>
<th>Back-end machines (pool members)</th>
<th>Internal</th>
<th>External</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>443</td>
<td>The machines that run the Ingress router pods, compute, or worker, by default.</td>
<td>X</td>
<td>X</td>
<td>HTTPS traffic</td>
</tr>
<tr>
<td>80</td>
<td>The machines that run the Ingress router pods, compute, or worker, by default.</td>
<td>X</td>
<td>X</td>
<td>HTTP traffic</td>
</tr>
</tbody>
</table>
TIP

If the true IP address of the client can be seen by the load balancer, enabling source IP-based session persistence can improve performance for applications that use end-to-end TLS encryption.

NOTE

A working configuration for the Ingress router is required for an OpenShift Container Platform cluster. You must configure the Ingress router after the control plane initializes.

NTP configuration

OpenShift Container Platform clusters are configured to use a public Network Time Protocol (NTP) server by default. If you want to use a local enterprise NTP server, or if your cluster is being deployed in a disconnected network, you can configure the cluster to use a specific time server. For more information, see the documentation for Configuring chrony time service.

If a DHCP server provides NTP server information, the chrony time service on the Red Hat Enterprise Linux CoreOS (RHCOS) machines read the information and can sync the clock with the NTP servers.

1.3.6. 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 Manager.

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 the 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 the 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.

1.3.7. 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:
     $$
     \texttt{curl} -k 'https://<engine-fqdn>/ovirt-engine/services/pki-resource?resource=ca-certificate&format=X509-PEM-CA' -o /tmp/ca.pem
     $$

     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--`):

   $$
   \texttt{chmod} 0644 /tmp/ca.pem
   $$

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

   $$
   \texttt{cp} -p /tmp/ca.pem /etc/pki/ca-trust/source/anchors/ca.pem
   $$

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:

     $$
     \texttt{update-ca-trust}
     $$

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.

1.3.8. 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**

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:

   ```bash
   $ ssh-keygen -t ed25519 -N '' -f <path>/<file_name>
   ```

   Specify the path and file name, such as ~/.ssh/id_rsa, of the new SSH key. If you have an existing key pair, ensure your public key is in the your ~/.ssh directory.

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

   **NOTE**

   If you plan to install an OpenShift Container Platform cluster that uses FIPS Validated / Modules in Process cryptographic libraries on the x86_64 architecture, do not create a key that uses the ed25519 algorithm. Instead, create a key that uses the rsa or ecdsa algorithm.

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

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

   **Example output**

   Agent pid 31874
NOTE

If your cluster is in FIPS mode, only use FIPS-compliant algorithms to generate the SSH key. The key must be either RSA or ECDSA.

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

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

Example output

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

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.

1.3.9. Obtaining the installation program

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

Prerequisites

- You have 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. Select your infrastructure provider.

3. 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 the installation program and the files that the installation program creates after you finish installing the cluster. Both files are required to delete 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.
4. 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
```

5. 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.

### 1.3.10. Downloading the Ansible playbooks

Download the Ansible playbooks for installing OpenShift Container Platform version 4.6 on RHV.

**Procedure**

- On your installation machine, run the following commands:

```
$ mkdir playbooks
$ cd playbooks
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.

### 1.3.11. 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**

```yaml
---
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/4.6/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
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 OpenShift Container Platform 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 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`.

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

### 1.3.12. 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/4.6/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:

   ```yaml
   rhcos:
   "https://mirror.openshift.com/pub/openshift-v4/dependencies/rhcos/4.6/latest/rhcos-openstack.x86_64.qcow2.gz"
   ```
1.3.13. 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 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>
? Enter ovirt-engine username <ocpadmin@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://console.redhat.com/openshift/install/pull-secret](https://console.redhat.com/openshift/install/pull-secret).
1.3.14. 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))'
```

1.3.15. 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:
   ```bash
   $ cp install-config.yaml install-config.yaml.backup
   ```

2. Generate a set of manifests in your assets directory:
   ```bash
   $ 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
  │   ├── etcd-serving-ca-configmap.yaml
  │   ├── etcd-signer-secret.yaml
  │   ├── kube-cloud-config.yaml
  │   ├── kube-system-configmap-root-ca.yaml
  │   ├── machine-config-server-tls-secret.yaml
  │   └── openshift-config-secret-pull-secret.yaml
```
Next steps

- Make control plane nodes non-schedulable.

### 1.3.16. Making control-plane nodes non-schedulable

Because you are manually creating and deploying the control plane machines, you must configure a manifest file to make the control plane nodes non-schedulable.

**Procedure**

1. To make the control plane nodes non-schedulable, enter:

   ```bash
   $ python3 -c 'import os, yaml
   path = "%s/manifests/cluster-scheduler-02-config.yml" % os.environ["ASSETS_DIR"]
data = yaml.safe_load(open(path))
data["spec"]['mastersSchedulable'] = False
   open(path, "w").write(yaml.dump(data, default_flow_style=False))'
   ```

### 1.3.17. Building the Ignition files

To build the Ignition files from the manifest files you just generated and modified, you run the installation program. This action creates a Red Hat Enterprise Linux CoreOS (RHCOS) machine, *initramfs*, which fetches the Ignition files and performs the configurations needed to create a node.

In addition to the Ignition files, the installation program generates the following:

- An **auth** directory that contains the admin credentials for connecting to the cluster with the *oc* and *kubectl* utilities.

- A **metadata.json** file that contains information such as the OpenShift Container Platform cluster name, cluster ID, and infrastructure ID for the current installation.

The Ansible playbooks for this installation process use the value of *infraID* as a prefix for the virtual machines they create. This prevents naming conflicts when there are multiple installations in the same oVirt/RHV cluster.

**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:
1.3.18. 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` file. 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 the `inventory.yml` file, prepend the value of `template` with `infraID`. For example:

   ```yaml
   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:

   ```bash
   $ openshift-install create ignition-configs --dir $ASSETS_DIR
   $ tree
   ...
   ```
1.3.19. 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:

   ```bash
   $ 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 IP address. To use SSH, enter:

   ```bash
   $ ssh core@<bootstrap_ip>
   ```

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

   ```bash
   [core@ocp4-lk6b4-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 control plane nodes (also known as the master nodes). After etcd has started on each control plane node and the nodes have joined the cluster, the errors should stop.

1.3.20. 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:

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

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

   ```bash
   $ 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

1.3.21. 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

1.3.22. 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.

### 1.3.23. 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**

<table>
<thead>
<tr>
<th>NAME</th>
<th>AGE</th>
<th>SIGNERNAME</th>
<th>REQUESTOR</th>
<th>CONDITION</th>
</tr>
</thead>
<tbody>
<tr>
<td>csr-2lnxd</td>
<td>63m</td>
<td>kubernetes.io/kubelet-serving</td>
<td>system:node:ocp4-lk6b4-master0.ocp4.example.org</td>
<td>Approved,Issued</td>
</tr>
<tr>
<td>csr-hff4q</td>
<td>64m</td>
<td>kubernetes.io/kube-apiserver-client-kubelet</td>
<td>system:serviceaccount:openshift-machine-config-operator:node-bootstrapper</td>
<td>Approved,Issued</td>
</tr>
<tr>
<td>csr-hsn96</td>
<td>60m</td>
<td>kubernetes.io/kubelet-serving</td>
<td>system:node:ocp4-lk6b4-master2.ocp4.example.org</td>
<td>Approved,Issued</td>
</tr>
<tr>
<td>csr-m724n</td>
<td>6m2s</td>
<td>kubernetes.io/kube-apiserver-client-kubelet</td>
<td>system:serviceaccount:openshift-machine-config-operator:node-bootstrapper</td>
<td>Pending</td>
</tr>
<tr>
<td>csr-p4dz2</td>
<td>60m</td>
<td>kubernetes.io/kube-apiserver-client-kubelet</td>
<td>system:serviceaccount:openshift-machine-config-operator:node-bootstrapper</td>
<td>Approved,Issued</td>
</tr>
<tr>
<td>csr-t9vfj</td>
<td>60m</td>
<td>kubernetes.io/kubelet-serving</td>
<td>system:node:ocp4-lk6b4-master1.ocp4.example.org</td>
<td>Approved,Issued</td>
</tr>
<tr>
<td>csr-tggtr</td>
<td>61m</td>
<td>kubernetes.io/kube-apiserver-client-kubelet</td>
<td>system:serviceaccount:openshift-machine-config-operator:node-bootstrapper</td>
<td>Approved,Issued</td>
</tr>
<tr>
<td>csr-wcbrf</td>
<td>7m6s</td>
<td>kubernetes.io/kube-apiserver-client-kubelet</td>
<td>system:serviceaccount:openshift-machine-config-operator:node-bootstrapper</td>
<td>Pending</td>
</tr>
</tbody>
</table>

3. 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
```

4. 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
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>
```

5. If the CSR information is correct, approve the request:

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

6. 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.

### 1.3.24. Telemetry access for OpenShift Container Platform

In OpenShift Container Platform 4.6, the Telemetry service, which runs by default to provide metrics about cluster health and the success of updates, 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).
After you confirm that your Red Hat OpenShift Cluster Manager inventory is correct, either maintained automatically by Telemetry or manually by using OCM, use subscription watch to track your OpenShift Container Platform subscriptions at the account or multi-cluster level.

**Additional resources**

- See About remote health monitoring for more information about the Telemetry service

### 1.4. UNINSTALLING A CLUSTER ON RHV

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

#### 1.4.1. Removing a cluster that uses installer-provisioned infrastructure

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

**NOTE**

After uninstallation, check your cloud provider for any resources not removed properly, especially with User Provisioned Infrastructure (UPI) clusters. There might be resources that the installer did not create or that the installer is unable to access.

**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 directory that contains the installation program on the computer that you used to install the cluster, run the following command:

```bash
$ ./openshift-install destroy cluster --dir <installation_directory> --log-level info
```

   **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.

#### 1.4.2. 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 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:

   ```sh
   $ 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.