OpenShift Container Platform 4.5

Installing on vSphere

Installing OpenShift Container Platform vSphere clusters

Last Updated: 2020-08-27

OpenShift Container Platform 4.5 Installing on vSphere

Installing OpenShift Container Platform vSphere clusters

Legal Notice

Copyright © 2020 Red Hat, Inc.

The text of and illustrations in this document are licensed by Red Hat under a Creative Commons Attribution–Share Alike 3.0 Unported license ("CC-BY-SA"). An explanation of CC-BY-SA is available at
http://creativecommons.org/licenses/by-sa/3.0/
. In accordance with CC-BY-SA, if you distribute this document or an adaptation of it, you must provide the URL for the original version.

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

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

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

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

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

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

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

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

All other trademarks are the property of their respective owners.

Abstract

This document provides instructions for installing OpenShift Container Platform clusters on VMware vSphere.

Table of Contents

CHAPTER 1. INSTALLING ON VSPHERE 6
1.1. INSTALLING A CLUSTER ON VSPHERE 6
1.1.1. Prerequisites 6
1.1.2. Internet and Telemetry access for OpenShift Container Platform 6
1.1.3. VMware vSphere infrastructure requirements 6
1.1.4. vCenter requirements 8
Required vCenter account privileges 8
Cluster resources 8
Cluster limits 9
Networking requirements 9
Required IP Addresses 9
DNS records 9
1.1.5. Generating an SSH private key and adding it to the agent 10
1.1.6. Obtaining the installation program 11
1.1.7. Adding vCenter root CA certificates to your system trust 12
1.1.8. Deploying the cluster 13
1.1.9. Installing the CLI by downloading the binary 15
1.1.9.1. Installing the CLI on Linux 15
1.1.9.2. Installing the CLI on Windows 15
1.1.9.3. Installing the CLI on macOS 16
1.1.10. Logging in to the cluster 16
1.1.11. Creating registry storage 17
1.1.11.1. Image registry removed during installation 17
1.1.11.2. Image registry storage configuration 17
1.1.11.2.1. Configuring registry storage for VMware vSphere 17
1.1.11.2.2. Configuring block registry storage for VMware vSphere 20
1.1.12. Next steps 20
1.2. INSTALLING A CLUSTER ON VSPHERE WITH CUSTOMIZATIONS 20
1.2.1. Prerequisites 20
1.2.2. Internet and Telemetry access for OpenShift Container Platform 21
1.2.3. VMware vSphere infrastructure requirements 21
1.2.4. vCenter requirements 23
Required vCenter account privileges 23
Cluster resources 23
Cluster limits 24
Networking requirements 24
Required IP Addresses 24
DNS records 24
1.2.5. Generating an SSH private key and adding it to the agent 25
1.2.6. Obtaining the installation program 26
1.2.7. Adding vCenter root CA certificates to your system trust 27
1.2.8. Creating the installation configuration file 28
1.2.8.1. Installation configuration parameters 29
1.2.8.2. Sample install-config.yaml file for an installer-provisioned VMware vSphere cluster 34
1.2.9. Deploying the cluster 35
1.2.10. Installing the CLI by downloading the binary 36
1.2.10.1. Installing the CLI on Linux 36
1.2.10.2. Installing the CLI on Windows 36
1.2.10.3. Installing the CLI on macOS 37
1.2.11. Logging in to the cluster 37
1.2.12. Creating registry storage 38

1.2.12.1. Image registry removed during installation 38
1.2.12.2. Image registry storage configuration 38
1.2.12.2.1. Configuring registry storage for VMware vSphere 39
1.2.12.2.2. Configuring block registry storage for VMware vSphere 41
1.2.13. Next steps 42
1.3. INSTALLING A CLUSTER ON VSPHERE 42
1.3.1. Prerequisites 42
1.3.2. Internet and Telemetry access for OpenShift Container Platform 42
1.3.3. VMware vSphere infrastructure requirements 43
1.3.4. Machine requirements for a cluster with user-provisioned infrastructure 44
1.3.4.1. Required machines 44
1.3.4.2. Network connectivity requirements 44
1.3.4.3. Minimum resource requirements 44
1.3.4.4. Certificate signing requests management 45
1.3.5. Creating the user-provisioned infrastructure 45
1.3.5.1. Networking requirements for user-provisioned infrastructure 45
Network topology requirements 46
Load balancers 47
Ethernet adaptor hardware address requirements 48
1.3.5.2. User-provisioned DNS requirements 48
1.3.6. Generating an SSH private key and adding it to the agent 49
1.3.7. Obtaining the installation program 50
1.3.8. Manually creating the installation configuration file 51
1.3.8.1. Sample install-config.yaml file for VMware vSphere 52
1.3.8.2. Configuring the cluster-wide proxy during installation 53
1.3.9. Creating the Kubernetes manifest and Ignition config files 55
1.3.10. Creating Red Hat Enterprise Linux CoreOS (RHCOS) machines in vSphere 56
1.3.11. Installing the CLI by downloading the binary 59
1.3.11.1. Installing the CLI on Linux 60
1.3.11.2. Installing the CLI on Windows 60
1.3.11.3. Installing the CLI on macOS 61
1.3.12. Creating the cluster 61
1.3.13. Logging in to the cluster 62
1.3.14. Approving the CSRs for your machines 62
1.3.15. Initial Operator configuration 64
1.3.15.1. Image registry removed during installation 65
1.3.15.2. Image registry storage configuration 65
1.3.15.2.1. Configuring registry storage for VMware vSphere 65
1.3.15.2.2. Configuring storage for the image registry in non-production clusters 68
1.3.15.2.3. Configuring block registry storage for VMware vSphere 69
1.3.16. Completing installation on user-provisioned infrastructure 69
1.3.17. Next steps 71
1.4. INSTALLING A CLUSTER ON VSPHERE WITH NETWORK CUSTOMIZATIONS 71
1.4.1. Prerequisites 72
1.4.2. Internet and Telemetry access for OpenShift Container Platform 72
1.4.3. VMware vSphere infrastructure requirements 72
1.4.4. Machine requirements for a cluster with user-provisioned infrastructure 74
1.4.4.1. Required machines 74
1.4.4.2. Network connectivity requirements 74
1.4.4.3. Minimum resource requirements 74
1.4.4.4. Certificate signing requests management 74
1.4.5. Creating the user-provisioned infrastructure 75
1.4.5.1. Networking requirements for user-provisioned infrastructure 75

Network topology requirements 76
Load balancers 76
1.4.5.2. User-provisioned DNS requirements 78
1.4.6. Generating an SSH private key and adding it to the agent 79
1.4.7. Obtaining the installation program 80
1.4.8. Manually creating the installation configuration file 80
1.4.8.1. Sample install-config.yaml file for VMware vSphere 81
1.4.8.2. Network configuration parameters 83
1.4.9. Modifying advanced network configuration parameters 84
1.4.10. Cluster Network Operator configuration 85
1.4.10.1. Configuration parameters for the OpenShift SDN default CNI network provider 86
1.4.10.2. Cluster Network Operator example configuration 87
1.4.11. Creating the Ignition config files 88
1.4.12. Creating Red Hat Enterprise Linux CoreOS (RHCOS) machines in vSphere 88
1.4.13. Installing the CLI by downloading the binary 91
1.4.13.1. Installing the CLI on Linux 92
1.4.13.2. Installing the CLI on Windows 92
1.4.13.3. Installing the CLI on macOS 93
1.4.14. Creating the cluster 93
1.4.15. Logging in to the cluster 94
1.4.16. Approving the CSRs for your machines 94
1.4.17. Initial Operator configuration 96
1.4.17.1. Image registry removed during installation 97
1.4.17.2. Image registry storage configuration 97
1.4.17.2.1. Configuring block registry storage for VMware vSphere 97
1.4.18. Completing installation on user-provisioned infrastructure 98
1.4.19. Next steps 100
1.5. INSTALLING A CLUSTER ON VSPHERE IN A RESTRICTED NETWORK 100
1.5.1. Prerequisites 100
1.5.2. About installations in restricted networks 101
1.5.2.1. Additional limits 101
1.5.3. Internet and Telemetry access for OpenShift Container Platform 101
1.5.4. VMware vSphere infrastructure requirements 102
1.5.5. Machine requirements for a cluster with user-provisioned infrastructure 103
1.5.5.1. Required machines 103
1.5.5.2. Network connectivity requirements 104
1.5.5.3. Minimum resource requirements 104
1.5.5.4. Certificate signing requests management 104
1.5.6. Creating the user-provisioned infrastructure 104
1.5.6.1. Networking requirements for user-provisioned infrastructure 105
Network topology requirements 106
Load balancers 106
1.5.6.2. User-provisioned DNS requirements 108
1.5.7. Generating an SSH private key and adding it to the agent 108
1.5.8. Manually creating the installation configuration file 109
1.5.8.1. Sample install-config.yaml file for VMware vSphere 110
1.5.8.2. Configuring the cluster-wide proxy during installation 112
1.5.9. Creating the Kubernetes manifest and Ignition config files 114
1.5.10. Creating Red Hat Enterprise Linux CoreOS (RHCOS) machines in vSphere 115
1.5.11. Creating the cluster 118
1.5.12. Logging in to the cluster 119
1.5.13. Approving the CSRs for your machines 120
1.5.14. Initial Operator configuration 121

1.5.14.1. Image registry storage configuration 122
1.5.14.1.1. Configuring registry storage for VMware vSphere 122
1.5.14.1.2. Configuring storage for the image registry in non-production clusters 125
1.5.14.1.3. Configuring block registry storage for VMware vSphere 125
1.5.15. Completing installation on user-provisioned infrastructure 126
1.5.16. Next steps 128
1.6. UNINSTALLING A CLUSTER ON VSPHERE THAT USES INSTALLER-PROVISIONED INFRASTRUCTURE 128
1.6.1. Removing a cluster that uses installer-provisioned infrastructure 128

CHAPTER 1. INSTALLING ON VSPHERE

1.1. INSTALLING A CLUSTER ON VSPHERE

In OpenShift Container Platform version 4.5, you can install a cluster on your VMware vSphere instance by using installer-provisioned infrastructure.

1.1.1. Prerequisites

- Provision persistent storage for your cluster. To deploy a private image registry, your storage must provide ReadWriteMany access modes.
- Review details about the OpenShift Container Platform installation and update processes.
- If you use a firewall, you must configure it to allow the sites that your cluster requires access to.

NOTE

Be sure to also review this site list if you are configuring a proxy.

1.1.2. Internet and Telemetry access for OpenShift Container Platform

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

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

You must have internet access to:

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

IMPORTANT

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

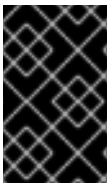
1.1.3. VMware vSphere infrastructure requirements

You must install the OpenShift Container Platform cluster on a VMware vSphere version 6 instance that meets the requirements for the components that you use.

Table 1.1. Minimum supported vSphere version for VMware components

Component	Minimum supported versions	Description
Hypervisor	vSphere 6.5 with HW version 13	This version is the minimum version that Red Hat Enterprise Linux CoreOS (RHCOS) supports. See the Red Hat Enterprise Linux 8 supported hypervisors list.
Networking (NSX-T)	n/a	vSphere 6.5U3 or vSphere 6.7U2+ are required for OpenShift Container Platform. Because previous versions of vSphere with NSX-T are not currently compatible with OpenShift Container Platform, NSX-T is not supported. NSX-T certification is in process and will be supported in a future release.
Storage with in-tree drivers	vSphere 6.5 or vSphere 6.7	This plug-in creates vSphere storage by using the in-tree storage drivers for vSphere included in OpenShift Container Platform and can be used when vSphere CSI drivers are not available.
Storage with vSphere CSI driver	vSphere 6.7U3 and later	This plug-in creates vSphere storage by using the standard Container Storage Interface. The vSphere CSI driver is provided and supported by VMware.

If you use a vSphere version 6.5 instance, consider upgrading to 6.7U2 before you install OpenShift Container Platform.



IMPORTANT

You must ensure that the time on your ESXi hosts is synchronized before you install OpenShift Container Platform. See Edit Time Configuration for a Host in the VMware documentation.

IMPORTANT

A limitation of using VPC is that the Storage Distributed Resource Scheduler (SDRS) is not supported. See link:https://vmware.github.io/vsphere-storage-for-kubernetes/documentation/faqs.html[vSphere Storage for Kubernetes FAQs] in the VMware documentation.

1.1.4. vCenter requirements

Before you install an OpenShift Container Platform cluster on your vCenter that uses infrastructure that the installer provisions, you must prepare your environment.

Required vCenter account privileges
To install an OpenShift Container Platform cluster in a vCenter, the installation program requires access to an account with privileges to read and create the required resources. Using an account that has administrative privileges is the simplest way to access all of the necessary permissions.

A user requires the following privileges to install an OpenShift Container Platform cluster:

- Datastore
  - **Allocate space**
- Folder
  - **Create folder**
  - **Delete folder**
- vSphere Tagging
  - All privileges
- Network
  - **Assign network**
- Resource
  - **Assign virtual machine to resource pool**
- Profile-driven storage
  - All privileges
- vApp
  - All privileges
- Virtual machine
  - All privileges

For more information about creating an account with only the required privileges, see vSphere Permissions and User Management Tasks in the vSphere documentation.

Cluster resources

When you deploy an OpenShift Container Platform cluster that uses installer-provisioned infrastructure, the installation program must be able to create several resources in your vCenter instance.

A standard OpenShift Container Platform installation creates the following vCenter resources:

- 1 Folder
- 1 Tag category
- 1 Tag
- Virtual machines:
  - 1 template
  - 1 temporary bootstrap node
  - 3 control plane nodes
  - 3 compute machines

Although these resources use 856 GB of storage, the bootstrap node is destroyed during the cluster installation process. A minimum of 800 GB of storage is required to use a standard cluster.

If you deploy more compute machines, the OpenShift Container Platform cluster will use more storage.

Cluster limits
Available resources vary between clusters. The number of possible clusters within a vCenter is limited primarily by available storage space and any limitations on the number of required resources. Be sure to consider both limitations to the vCenter resources that the cluster creates and the resources that you require to deploy a cluster, such as IP addresses and networks.

Networking requirements
In addition to using DHCP for the network, you must create the following networking resources before you install the OpenShift Container Platform cluster:

Required IP Addresses
An installer-provisioned vSphere installation requires two static IP addresses:

- The **API** address is used to access the cluster API.
- The **Ingress** address is used for cluster ingress traffic.

You must provide these IP addresses to the installation program when you install the OpenShift Container Platform cluster.

DNS records
You must create DNS records for two static IP addresses in the appropriate DNS server for the vCenter instance that hosts your OpenShift Container Platform cluster. In each record, **<cluster_name>** is the cluster name and **<base_domain>** is the cluster base domain that you specify when you install the cluster. A complete DNS record takes the form: **<component>.<cluster_name>.<base_domain>.**.

Table 1.2. Required DNS records

Component	Record	Description
API VIP	**api.<cluster_name>.<base_domain>.**	This DNS A/AAAA or CNAME record must point to the load balancer for the control plane machines. This record must be resolvable by both clients external to the cluster and from all the nodes within the cluster.
Ingress VIP	***.apps.<cluster_name>.<base_domain>.**	A wildcard DNS A/AAAA or CNAME record that points to the load balancer that targets the machines that run the Ingress router pods, which are the worker nodes by default. This record must be resolvable by both clients external to the cluster and from all the nodes within the cluster.

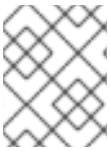
1.1.5. Generating an SSH private key and adding it to the agent

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 to the installation program.

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:

```
$ ssh-keygen -t rsa -b 4096 -N '' \
    -f <path>/<file_name> 1
```

1 Specify the path and file name, such as **~/.ssh/id_rsa**, of the SSH key. Do not specify an existing SSH key, as it will be overwritten.

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

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

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

Example output

```
Agent pid 31874
```

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

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

Example output

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

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

Next steps

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

1.1.6. Obtaining the installation program

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

Prerequisites

- You must install the cluster from a computer that uses Linux or macOS.
- You need 500 MB of local disk space to download the installation program.

Procedure

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

IMPORTANT

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

IMPORTANT

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

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

```
$ tar xvf <installation_program>.tar.gz
```

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

1.1.7. Adding vCenter root CA certificates to your system trust

Because the installation program requires access to your vCenter's API, you must add your vCenter's trusted root CA certificates to your system trust before you install an OpenShift Container Platform cluster.

Procedure

1. From the vCenter home page, download the vCenter's root CA certificates. Click **Download trusted root CA certificates** in the vSphere Web Services SDK section. The **<vCenter>/certs/download.zip** file downloads.

2. Extract the compressed file that contains the vCenter root CA certificates. The contents of the compressed file resemble the following file structure:

```
certs
├── lin
│   ├── 108f4d17.0
│   ├── 108f4d17.r1
│   ├── 7e757f6a.0
│   ├── 8e4f8471.0
│   └── 8e4f8471.r0
├── mac
│   ├── 108f4d17.0
│   ├── 108f4d17.r1
│   ├── 7e757f6a.0
│   ├── 8e4f8471.0
│   └── 8e4f8471.r0
└── win
    ├── 108f4d17.0.crt
    ├── 108f4d17.r1.crl
    ├── 7e757f6a.0.crt
    ├── 8e4f8471.0.crt
    └── 8e4f8471.r0.crl

3 directories, 15 files
```

3. Add the files for your operating system to the system trust. For example, on a Fedora operating system, run the following command:

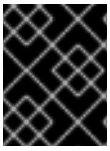
```
# cp certs/lin/* /etc/pki/ca-trust/source/anchors
```

4. Update your system trust. For example, on a Fedora operating system, run the following command:

```
# update-ca-trust extract
```

1.1.8. Deploying the cluster

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



IMPORTANT

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

Prerequisites

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

Procedure

1. Run the installation program:

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

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

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

IMPORTANT

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

Provide values at the prompts:

a. Optional: Select an SSH key to use to access your cluster machines.

NOTE

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

b. Select **vsphere** as the platform to target.

c. Specify the name of your vCenter instance.

d. Specify the user name and password for the vCenter account that has the required permissions to create the cluster.
The installation program connects to your vCenter instance.

e. Select the datacenter in your vCenter instance to connect to.

f. Select the default vCenter datastore to use.

g. Select the vCenter cluster to install the OpenShift Container Platform cluster in.

h. Select the network in the vCenter instance that contains the virtual IP addresses and DNS records that you configured.

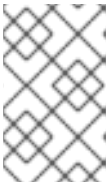
i. Enter the virtual IP address that you configured for control plane API access.

j. Enter the virtual IP address that you configured for cluster ingress.

k. Enter the base domain. This base domain must be the same one that you used in the DNS records that you configured.

l. Enter a descriptive name for your cluster. The cluster name must be the same one that you used in the DNS records that you configured.

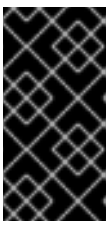
m. Paste the pull secret that you obtained from the Pull Secret page on the Red Hat OpenShift Cluster Manager site.



NOTE

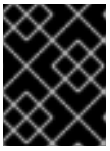
If the cloud provider account that you configured on your host does not have sufficient permissions to deploy the cluster, the installation process stops, and the missing permissions are displayed.

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



IMPORTANT

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



IMPORTANT

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

1.1.9. Installing the CLI by downloading the binary

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

IMPORTANT

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

1.1.9.1. Installing the CLI on Linux

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

Procedure

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

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

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

```
$ echo $PATH
```

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

```
$ oc <command>
```

1.1.9.2. Installing the CLI on Windows

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

Procedure

1. Navigate to the Infrastructure Provider page on the Red Hat OpenShift Cluster Manager site.
2. Select your infrastructure provider, and, if applicable, your installation type.
3. In the **Command-line interface** section, select **Windows** from the drop-down menu and click **Download command-line tools**.

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

```
C:\> path
```

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

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

1.1.9.3. Installing the CLI on macOS

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

Procedure

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

```
$ echo $PATH
```

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

```
$ oc <command>
```

1.1.10. Logging in to the cluster

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

Prerequisites

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

Procedure

1. Export the **kubeadmin** credentials:

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

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

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

```
$ oc whoami
```

Example output

```
system:admin
```

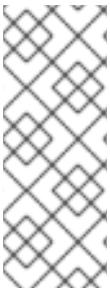
1.1.11. Creating registry storage

After you install the cluster, you must create storage for the registry Operator.

1.1.11.1. Image registry removed during installation

On platforms that do not provide shareable object storage, the OpenShift Image Registry Operator bootstraps itself as **Removed**. This allows **openshift-installer** to complete installations on these platform types.

After installation, you must edit the Image Registry Operator configuration to switch the **ManagementState** from **Removed** to **Managed**.



NOTE

The Prometheus console provides an **ImageRegistryRemoved** alert, for example:

"Image Registry has been removed. **ImageStreamTags**, **BuildConfigs** and **DeploymentConfigs** which reference **ImageStreamTags** may not work as expected. Please configure storage and update the config to **Managed** state by editing configs.imageregistry.operator.openshift.io."

1.1.11.2. Image registry storage configuration

The **image-registry** Operator is not initially available for platforms that do not provide default storage. After installation, you must configure your registry to use storage so that the Registry Operator is made available.

Instructions are shown for configuring a PersistentVolume, which is required for production clusters. Where applicable, instructions are shown for configuring an empty directory as the storage location, which is available for only non-production clusters.

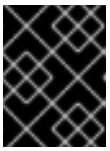
Additional instructions are provided for allowing the image registry to use block storage types by using the **Recreate** rollout strategy during upgrades.

1.1.11.2.1. Configuring registry storage for VMware vSphere

As a cluster administrator, following installation you must configure your registry to use storage.

Prerequisites

- Cluster administrator permissions.
- A cluster on VMware vSphere.
- Provision persistent storage for your cluster. To deploy a private image registry, your storage must provide ReadWriteMany access mode.



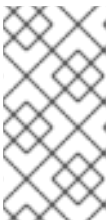
IMPORTANT

vSphere volumes do not support the **ReadWriteMany** access mode. You must use a different storage backend, such as **NFS**, to configure the registry storage.

- Must have "100Gi" capacity.

Procedure

1. To configure your registry to use storage, change the **spec.storage.pvc** in the **configs.imageregistry/cluster** resource.



NOTE

When using shared storage such as NFS, it is strongly recommended to use the **supplementalGroups** strategy, which dictates the allowable supplemental groups for the Security Context, rather than the **fsGroup** ID. Refer to the NFS **Group IDs** documentation for details.

2. Verify you do not have a registry Pod:

```
$ oc get pod -n openshift-image-registry
```

NOTE

- If the storage type is **emptyDIR**, the replica number cannot be greater than **1**.
- If the storage type is **NFS**, you must enable the **no_wdelay** and **root_squash** mount options. For example:

```
# cat /etc/exports
```

Example output

```
/mnt/data *(rw,sync,no_wdelay,root_squash,insecure,fsid=0)
```

```
sh-4.2# exportfs -rv
```

Example output

```
exporting *:/mnt/data
```

3. Check the registry configuration:

```
$ oc edit configs.imageregistry.operator.openshift.io
```

Example registry configuration

```
storage:
  pvc:
    claim:
```

Leave the **claim** field blank to allow the automatic creation of an **image-registry-storage** PVC.

4. Optional: Add a new storage class to a PV:

   a. Create the PV:

```
$ oc create -f -
```

```
apiVersion: v1
kind: PersistentVolume
metadata:
  name: image-registry-pv
spec:
  accessModes:
    - ReadWriteMany
  capacity:
      storage: 100Gi
  nfs:
    path: /registry
    server: 172.16.231.181
  persistentVolumeReclaimPolicy: Retain
  storageClassName: nfs01
```

```
$ oc get pv
```

   b. Create the PVC:

```
$ oc create -n openshift-image-registry -f -
```

```
apiVersion: "v1"
kind: "PersistentVolumeClaim"
metadata:
  name: "image-registry-pvc"
spec:
  accessModes:
    - ReadWriteMany
  resources:
    requests:
      storage: 100Gi
  storageClassName: nfs01
  volumeMode: Filesystem
```

```
$ oc get pvc -n openshift-image-registry
```

-

Finally, add the name of your PVC:

```
$ oc edit configs.imageregistry.operator.openshift.io -o yaml
```

```
storage:
  pvc:
    claim: image-registry-pvc 1
```

1 Creating a custom PVC allows you to leave the **claim** field blank for default automatic creation of an **image-registry-storage** PVC.

5. Check the **clusteroperator** status:

```
$ oc get clusteroperator image-registry
```

1.1.11.2.2. Configuring block registry storage for VMware vSphere

To allow the image registry to use block storage types such as vSphere Virtual Machine Disk (VMDK) during upgrades as a cluster administrator, you can use the **Recreate** rollout strategy.

Procedure

1. To set the image registry storage as a block storage type, patch the registry so that it uses the **Recreate** rollout strategy and runs with only **1** replica:

```
$ oc patch config.imageregistry.operator.openshift.io/cluster --type=merge -p '{"spec":
{"rolloutStrategy":"Recreate","replicas":1}}'
```

2. Provision the PV for the block storage device, and create a PVC for that volume. The requested block volume uses the ReadWriteOnce (RWO) access mode.

3. Edit the registry configuration so that it references the correct PVC.

For instructions about configuring registry storage so that it references the correct PVC, see Configuring the registry for vSphere.

1.1.12. Next steps

- Customize your cluster.
- If necessary, you can opt out of remote health reporting .
- Set up your registry and configure registry storage .

1.2. INSTALLING A CLUSTER ON VSPHERE WITH CUSTOMIZATIONS

In OpenShift Container Platform version 4.5, you can install a cluster on your VMware vSphere instance by using installer-provisioned infrastructure. To customize the installation, you modify parameters in the **install-config.yaml** file before you install the cluster.

1.2.1. Prerequisites

- Provision persistent storage for your cluster. To deploy a private image registry, your storage must provide ReadWriteMany access modes.
- Review details about the OpenShift Container Platform installation and update processes.
- If you use a firewall, you must configure it to allow the sites that your cluster requires access to.

NOTE

Be sure to also review this site list if you are configuring a proxy.

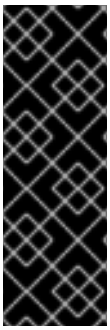
1.2.2. Internet and Telemetry access for OpenShift Container Platform

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

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

You must have internet access to:

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



IMPORTANT

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

1.2.3. VMware vSphere infrastructure requirements

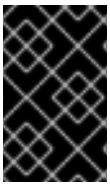
You must install the OpenShift Container Platform cluster on a VMware vSphere version 6 instance that meets the requirements for the components that you use.

Table 1.3. Minimum supported vSphere version for VMware components

Component	Minimum supported versions	Description

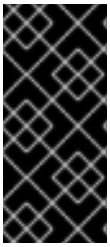
Component	Minimum supported versions	Description
Hypervisor	vSphere 6.5 with HW version 13	This version is the minimum version that Red Hat Enterprise Linux CoreOS (RHCOS) supports. See the Red Hat Enterprise Linux 8 supported hypervisors list.
Networking (NSX-T)	n/a	vSphere 6.5U3 or vSphere 6.7U2+ are required for OpenShift Container Platform. Because previous versions of vSphere with NSX-T are not currently compatible with OpenShift Container Platform, NSX-T is not supported. NSX-T certification is in process and will be supported in a future release.
Storage with in-tree drivers	vSphere 6.5 or vSphere 6.7	This plug-in creates vSphere storage by using the in-tree storage drivers for vSphere included in OpenShift Container Platform and can be used when vSphere CSI drivers are not available.
Storage with vSphere CSI driver	vSphere 6.7U3 and later	This plug-in creates vSphere storage by using the standard Container Storage Interface. The vSphere CSI driver is provided and supported by VMware.

If you use a vSphere version 6.5 instance, consider upgrading to 6.7U2 before you install OpenShift Container Platform.



IMPORTANT

You must ensure that the time on your ESXi hosts is synchronized before you install OpenShift Container Platform. See Edit Time Configuration for a Host in the VMware documentation.



IMPORTANT

A limitation of using VPC is that the Storage Distributed Resource Scheduler (SDRS) is not supported. See link:https://vmware.github.io/vsphere-storage-for-kubernetes/documentation/faqs.html[vSphere Storage for Kubernetes FAQs] in the VMware documentation.

1.2.4. vCenter requirements

Before you install an OpenShift Container Platform cluster on your vCenter that uses infrastructure that the installer provisions, you must prepare your environment.

Required vCenter account privileges
To install an OpenShift Container Platform cluster in a vCenter, the installation program requires access to an account with privileges to read and create the required resources. Using an account that has administrative privileges is the simplest way to access all of the necessary permissions.

A user requires the following privileges to install an OpenShift Container Platform cluster:

- Datastore
  - **Allocate space**
- Folder
  - **Create folder**
  - **Delete folder**
- vSphere Tagging
  - All privileges
- Network
  - **Assign network**
- Resource
  - **Assign virtual machine to resource pool**
- Profile-driven storage
  - All privileges
- vApp
  - All privileges
- Virtual machine
  - All privileges

For more information about creating an account with only the required privileges, see vSphere Permissions and User Management Tasks in the vSphere documentation.

Cluster resources
When you deploy an OpenShift Container Platform cluster that uses installer-provisioned infrastructure, the installation program must be able to create several resources in your vCenter instance.

A standard OpenShift Container Platform installation creates the following vCenter resources:

- 1 Folder
- 1 Tag category

- 1 Tag
- Virtual machines:
  - 1 template
  - 1 temporary bootstrap node
  - 3 control plane nodes
  - 3 compute machines

Although these resources use 856 GB of storage, the bootstrap node is destroyed during the cluster installation process. A minimum of 800 GB of storage is required to use a standard cluster.

If you deploy more compute machines, the OpenShift Container Platform cluster will use more storage.

Cluster limits
Available resources vary between clusters. The number of possible clusters within a vCenter is limited primarily by available storage space and any limitations on the number of required resources. Be sure to consider both limitations to the vCenter resources that the cluster creates and the resources that you require to deploy a cluster, such as IP addresses and networks.

Networking requirements
In addition to using DHCP for the network, you must create the following networking resources before you install the OpenShift Container Platform cluster:

Required IP Addresses
An installer-provisioned vSphere installation requires two static IP addresses:

- The **API** address is used to access the cluster API.
- The **Ingress** address is used for cluster ingress traffic.

You must provide these IP addresses to the installation program when you install the OpenShift Container Platform cluster.

DNS records
You must create DNS records for two static IP addresses in the appropriate DNS server for the vCenter instance that hosts your OpenShift Container Platform cluster. In each record, **<cluster_name>** is the cluster name and **<base_domain>** is the cluster base domain that you specify when you install the cluster. A complete DNS record takes the form: **<component>.<cluster_name>.<base_domain>.**.

Table 1.4. Required DNS records

Component	Record	Description
API VIP	**api.<cluster_name>.<base_domain>.**	This DNS A/AAAA or CNAME record must point to the load balancer for the control plane machines. This record must be resolvable by both clients external to the cluster and from all the nodes within the cluster.

Component	Record	Description
Ingress VIP	***.apps.<cluster_name>.<base_domain>.**	A wildcard DNS A/AAAA or CNAME record that points to the load balancer that targets the machines that run the Ingress router pods, which are the worker nodes by default. This record must be resolvable by both clients external to the cluster and from all the nodes within the cluster.

1.2.5. 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 to the installation program.

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:

```
$ ssh-keygen -t rsa -b 4096 -N '' \
    -f <path>/<file_name> 1
```

1 Specify the path and file name, such as **~/.ssh/id_rsa**, of the SSH key. Do not specify an existing SSH key, as it will be overwritten.

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

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

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

   Example output

```
Agent pid 31874
```

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

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

Example output

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

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

Next steps

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

1.2.6. Obtaining the installation program

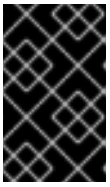
Before you install OpenShift Container Platform, download the installation file on a local computer.

Prerequisites

- You must install the cluster from a computer that uses Linux or macOS.
- You need 500 MB of local disk space to download the installation program.

Procedure

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



IMPORTANT

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

IMPORTANT

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

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

```
$ tar xvf <installation_program>.tar.gz
```

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

1.2.7. Adding vCenter root CA certificates to your system trust

Because the installation program requires access to your vCenter's API, you must add your vCenter's trusted root CA certificates to your system trust before you install an OpenShift Container Platform cluster.

Procedure

1. From the vCenter home page, download the vCenter's root CA certificates. Click **Download trusted root CA certificates** in the vSphere Web Services SDK section. The **<vCenter>/certs/download.zip** file downloads.

2. Extract the compressed file that contains the vCenter root CA certificates. The contents of the compressed file resemble the following file structure:

```
certs
├── lin
│   ├── 108f4d17.0
│   ├── 108f4d17.r1
│   ├── 7e757f6a.0
│   ├── 8e4f8471.0
│   └── 8e4f8471.r0
├── mac
│   ├── 108f4d17.0
│   ├── 108f4d17.r1
│   ├── 7e757f6a.0
│   ├── 8e4f8471.0
│   └── 8e4f8471.r0
└── win
    ├── 108f4d17.0.crt
    ├── 108f4d17.r1.crl
    ├── 7e757f6a.0.crt
    ├── 8e4f8471.0.crt
    └── 8e4f8471.r0.crl

3 directories, 15 files
```

3. Add the files for your operating system to the system trust. For example, on a Fedora operating system, run the following command:

```
# cp certs/lin/* /etc/pki/ca-trust/source/anchors
```

4. Update your system trust. For example, on a Fedora operating system, run the following command:

```
# update-ca-trust extract
```

1.2.8. Creating the installation configuration file

You can customize the OpenShift Container Platform cluster you install on VMware vSphere.

Prerequisites

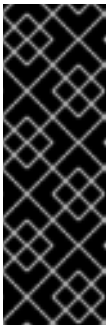
- Download the OpenShift Container Platform installation program and the pull secret for your cluster.

Procedure

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

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

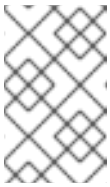
   1 For **<installation_directory>**, specify the directory name to store the files that the installation program creates.



IMPORTANT

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

   b. At the prompts, provide the configuration details for your cloud:
      i. Optional: Select an SSH key to use to access your cluster machines.



NOTE

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

      ii. Select **vsphere** as the platform to target.
      iii. Specify the name of your vCenter instance.
      iv. Specify the user name and password for the vCenter account that has the required permissions to create the cluster.
      The installation program connects to your vCenter instance.
      v. Select the datacenter in your vCenter instance to connect to.
      vi. Select the default vCenter datastore to use.

vii. Select the vCenter cluster to install the OpenShift Container Platform cluster in.

viii. Select the network in the vCenter instance that contains the virtual IP addresses and DNS records that you configured.

ix. Enter the virtual IP address that you configured for control plane API access.

x. Enter the virtual IP address that you configured for cluster ingress.

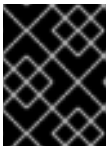
xi. Enter the base domain. This base domain must be the same one that you used in the DNS records that you configured.

xii. Enter a descriptive name for your cluster. The cluster name must be the same one that you used in the DNS records that you configured.

xiii. Paste the pull secret that you obtained 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.

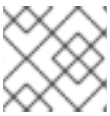


IMPORTANT

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

1.2.8.1. 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

You cannot modify these parameters in the **install-config.yaml** file after installation.

Table 1.5. Required parameters

Parameter	Description	Values

Parameter	Description	Values
baseDomain	The base domain of your cloud provider. This value is used to create routes to your OpenShift Container Platform cluster components. The full DNS name for your cluster is a combination of the **baseDomain** and **metadata.name** parameter values that uses the **<metadata.name>.<baseDomain>** format.	A fully-qualified domain or subdomain name, such as **example.com**.
controlPlane.platform	The cloud provider to host the control plane machines. This parameter value must match the **compute.platform** parameter value.	**aws**, **azure**, **gcp**, **openstack**, **vsphere**, or **{}**
compute.platform	The cloud provider to host the worker machines. This parameter value must match the **controlPlane.platform** parameter value.	**aws**, **azure**, **gcp**, **openstack**, **vsphere**, or **{}**
metadata.name	The name of your cluster.	A string that contains lowercase letters, hyphens (-), or periods (.), such as **dev**.
platform.<platform>.region	The region to deploy your cluster in.	A valid region for your cloud, such as **us-east-1** for AWS, **centralus** for Azure. Red Hat OpenStack Platform (RHOSP) and vSphere do not use this parameter.

pullSecret	The pull secret that you obtained from the Pull Secret page on the Red Hat OpenShift Cluster Manager site. You use this pull secret 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.	<pre>{
		  "auths":{
		  "cloud.openshift.com":{
		  "auth":"b3Blb=",
		  "email":"you@example.com"
		  },
		  "quay.io":{
		  "auth":"b3Blb=",
		  "email":"you@example.com"
		  }
		  }
		}</pre>

Table 1.6. Optional parameters

Parameter	Description	Values
sshKey	The SSH key to use to access your cluster machines. **NOTE** For production OpenShift Container Platform clusters on which you want to perform installation debugging or disaster recovery, specify an SSH key that your **ssh-agent** process uses.	A valid, local public SSH key that you added to the **ssh-agent** process.
fips	Whether to enable or disable FIPS mode. By default, FIPS mode is not enabled. If FIPS mode is enabled, the Red Hat Enterprise Linux CoreOS (RHCOS) machines that OpenShift Container Platform runs on bypass the default Kubernetes cryptography suite and use the cryptography modules that are provided with RHCOS instead.	**false** or **true**
publish	How to publish the user-facing endpoints of your cluster.	**Internal** or **External**. Set **publish** to **Internal** to deploy a private cluster, which cannot be accessed from the internet. The default value is **External**.
compute.hyperthreading	Whether to enable or disable simultaneous multithreading, or **hyperthreading**, on compute machines. By default, simultaneous multithreading is enabled to increase the performance of your machines' cores. **IMPORTANT** If you disable simultaneous multithreading, ensure that your capacity planning accounts for the dramatically decreased machine performance.	**Enabled** or **Disabled**

Parameter	Description	Values
compute.replicas	The number of compute machines, which are also known as worker machines, to provision.	A positive integer greater than or equal to **2**. The default value is **3**.
controlPlane.hyperthreading	Whether to enable or disable simultaneous multithreading, or **hyperthreading**, on control plane machines. By default, simultaneous multithreading is enabled to increase the performance of your machines' cores. **IMPORTANT** If you disable simultaneous multithreading, ensure that your capacity planning accounts for the dramatically decreased machine performance.	**Enabled** or **Disabled**
controlPlane.replicas	The number of control plane machines to provision.	A positive integer greater than or equal to **3**. The default value is **3**.

Table 1.7. Additional VMware vSphere cluster parameters

Parameter	Description	Values
platform.vsphere.vCenter	The fully-qualified host name or IP address of the vCenter server.	String
platform.vsphere.username	The user name to use to connect to the vCenter instance with. This user must have at least the roles and privileges that are required for static or dynamic persistent volume provisioning in vSphere.	String.
platform.vsphere.password	The password for the vCenter user name.	String.
platform.vsphere.datacenter	The name of the datacenter to use in the vCenter instance.	String.

Parameter	Description	Values
platform.vsphere.defaultDatastore	The name of the default datastore to use for provisioning volumes.	String.
platform.vsphere.folder	*Optional*. The absolute path of an existing folder where the installation program creates the virtual machines. create VMs. If you do not provide this value, the installation program creates a folder that is named with the cluster ID is created in the datacenter virtual machine folder.	String, for example, **/<datacenter_name>/vm/<folder_name>/<subfolder_name>**.
platform.vsphere.network	The network in the vCenter instance that contains the virtual IP addresses and DNS records that you configured.	String.
platform.vsphere.cluster	The vCenter cluster to install the OpenShift Container Platform cluster in.	String.
platform.vsphere.apiVIP	The virtual IP address that you configured for control plane API access.	An IP address, for example **128.0.0.1**.
platform.vsphere.ingressVIP	The virtual IP address that you configured for cluster ingress.	An IP address, for example **128.0.0.1**.

Table 1.8. Optional VMware vSphere machine pool parameters

Parameter	Description	Values
platform.vsphere.osDisk.diskSizeGB	The size of the disk in gigabytes.	Integer.
platform.vsphere.cpus	The total number of virtual processor cores to assign a virtual machine.	Integer.
platform.vsphere.coresPerSocket	The number of cores per socket in a virtual machine. The number of vCPUs on the virtual machine is will be **platform.vsphere.cpus**/**platform.vsphere.coresPerSocket**. The default value is **1**	Integer.
platform.vsphere.memoryMB	The size of a virtual machine's memory in megabytes..	Integer.

1.2.8.2. Sample install-config.yaml file for an installer-provisioned VMware vSphere cluster

You can customize the **install-config.yaml** file to specify more details about your OpenShift Container Platform cluster's platform or modify the values of the required parameters.

```
apiVersion: v1
baseDomain: example.com (1)
compute: (2)
- hyperthreading: Enabled (3)
  name: worker
  replicas: 3
  platform:
    vsphere: (4)
      cpus: 2
      coresPerSocket: 2
      memoryMB: 8196
      osDisk:
        diskSizeGB: 120
controlPlane: (5)
  hyperthreading: Enabled (6)
  name: master
  replicas: 3
  platform:
    vsphere: (7)
      cpus: 4
      coresPerSocket: 2
      memoryMB: 16384
      osDisk:
        diskSizeGB: 120
metadata:
  name: cluster (8)
platform:
  vsphere:
    vcenter: your.vcenter.server
    username: username
    password: password
    datacenter: datacenter
    defaultDatastore: datastore
    folder: folder
    network: VM_Network
    cluster: vsphere_cluster_name
    apiVIP: api_vip
    ingressVIP: ingress_vip
fips: false
pullSecret: '{"auths":{"<mirror_registry>": {"auth": "<credentials>","email": "you@example.com"}}}'
sshKey: 'ssh-ed25519 AAAA...'
```

(1) The base domain of the cluster. All DNS records must be sub-domains of this base and include the cluster name.

(2) (5) The **controlPlane** section is a single mapping, but the compute section is a sequence of mappings. To meet the requirements of the different data structures, the first line of the **compute** section must begin with a hyphen, **-**, and the first line of the **controlPlane** section must not. Although both sections currently define a single machine pool, it is possible that future versions of OpenShift Container Platform will support defining multiple compute pools during installation. Only one

control plane pool is used.

3 6 Whether to enable or disable simultaneous multithreading, or **hyperthreading**. By default, simultaneous multithreading is enabled to increase the performance of your machines' cores. You can disable it by setting the parameter value to **Disabled**. If you disable simultaneous multithreading in some cluster machines, you must disable it in all cluster machines.

IMPORTANT

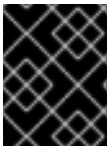
If you disable simultaneous multithreading, ensure that your capacity planning accounts for the dramatically decreased machine performance. Your machines must use at least 8 CPUs and 32 GB of RAM if you disable simultaneous multithreading.

4 7 Optional: Provide additional configuration for the machine pool parameters for the compute and control plane machines.

8 The cluster name that you specified in your DNS records.

1.2.9. Deploying the cluster

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



IMPORTANT

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

Prerequisites

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

Procedure

1. Run the installation program:

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

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

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

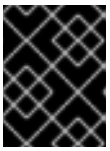
NOTE

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

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

IMPORTANT

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

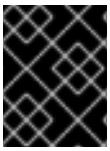


IMPORTANT

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

1.2.10. Installing the CLI by downloading the binary

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



IMPORTANT

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

1.2.10.1. Installing the CLI on Linux

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

Procedure

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

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

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

```
$ echo $PATH
```

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

```
$ oc <command>
```

1.2.10.2. Installing the CLI on Windows

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

Procedure

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

```
C:\> path
```

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

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

1.2.10.3. Installing the CLI on macOS

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

Procedure

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

```
$ echo $PATH
```

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

```
$ oc <command>
```

1.2.11. Logging in to the cluster

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

Prerequisites

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

Procedure

1. Export the **kubeadmin** credentials:

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

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

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

```
$ oc whoami
```

Example output

```
system:admin
```

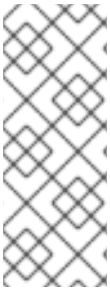
1.2.12. Creating registry storage

After you install the cluster, you must create storage for the registry Operator.

1.2.12.1. Image registry removed during installation

On platforms that do not provide shareable object storage, the OpenShift Image Registry Operator bootstraps itself as **Removed**. This allows **openshift-installer** to complete installations on these platform types.

After installation, you must edit the Image Registry Operator configuration to switch the **ManagementState** from **Removed** to **Managed**.



NOTE

The Prometheus console provides an **ImageRegistryRemoved** alert, for example:

"Image Registry has been removed. **ImageStreamTags**, **BuildConfigs** and **DeploymentConfigs** which reference **ImageStreamTags** may not work as expected. Please configure storage and update the config to **Managed** state by editing configs.imageregistry.operator.openshift.io."

1.2.12.2. Image registry storage configuration

The **image-registry** Operator is not initially available for platforms that do not provide default storage. After installation, you must configure your registry to use storage so that the Registry Operator is made available.

Instructions are shown for configuring a PersistentVolume, which is required for production clusters. Where applicable, instructions are shown for configuring an empty directory as the storage location, which is available for only non-production clusters.

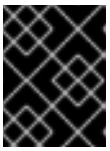
Additional instructions are provided for allowing the image registry to use block storage types by using the **Recreate** rollout strategy during upgrades.

1.2.12.2.1. Configuring registry storage for VMware vSphere

As a cluster administrator, following installation you must configure your registry to use storage.

Prerequisites

- Cluster administrator permissions.
- A cluster on VMware vSphere.
- Provision persistent storage for your cluster. To deploy a private image registry, your storage must provide ReadWriteMany access mode.



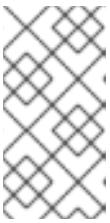
IMPORTANT

vSphere volumes do not support the **ReadWriteMany** access mode. You must use a different storage backend, such as **NFS**, to configure the registry storage.

- Must have "100Gi" capacity.

Procedure

1. To configure your registry to use storage, change the **spec.storage.pvc** in the **configs.imageregistry/cluster** resource.



NOTE

When using shared storage such as NFS, it is strongly recommended to use the **supplementalGroups** strategy, which dictates the allowable supplemental groups for the Security Context, rather than the **fsGroup** ID. Refer to the NFS **Group IDs** documentation for details.

2. Verify you do not have a registry Pod:

```
$ oc get pod -n openshift-image-registry
```

NOTE

- If the storage type is **emptyDIR**, the replica number cannot be greater than **1**.
- If the storage type is **NFS**, you must enable the **no_wdelay** and **root_squash** mount options. For example:

```
# cat /etc/exports
```

Example output

```
/mnt/data *(rw,sync,no_wdelay,root_squash,insecure,fsid=0)
```

```
sh-4.2# exportfs -rv
```

Example output

```
exporting *:/mnt/data
```

3. Check the registry configuration:

```
$ oc edit configs.imageregistry.operator.openshift.io
```

Example registry configuration

```
storage:
  pvc:
    claim:
```

Leave the **claim** field blank to allow the automatic creation of an **image-registry-storage** PVC.

4. Optional: Add a new storage class to a PV:

   a. Create the PV:

```
$ oc create -f -
```

```
apiVersion: v1
kind: PersistentVolume
metadata:
  name: image-registry-pv
spec:
  accessModes:
    - ReadWriteMany
  capacity:
      storage: 100Gi
  nfs:
    path: /registry
    server: 172.16.231.181
  persistentVolumeReclaimPolicy: Retain
  storageClassName: nfs01
```

```
$ oc get pv
```

b. Create the PVC:

```
$ oc create -n openshift-image-registry -f -
```

```
apiVersion: "v1"
kind: "PersistentVolumeClaim"
metadata:
  name: "image-registry-pvc"
spec:
  accessModes:
    - ReadWriteMany
  resources:
    requests:
      storage: 100Gi
  storageClassName: nfs01
  volumeMode: Filesystem
```

```
$ oc get pvc -n openshift-image-registry
```

Finally, add the name of your PVC:

```
$ oc edit configs.imageregistry.operator.openshift.io -o yaml
```

```
storage:
  pvc:
    claim: image-registry-pvc (1)
```

(1) Creating a custom PVC allows you to leave the **claim** field blank for default automatic creation of an **image-registry-storage** PVC.

5. Check the **clusteroperator** status:

```
$ oc get clusteroperator image-registry
```

1.2.12.2.2. Configuring block registry storage for VMware vSphere

To allow the image registry to use block storage types such as vSphere Virtual Machine Disk (VMDK) during upgrades as a cluster administrator, you can use the **Recreate** rollout strategy.

Procedure

1. To set the image registry storage as a block storage type, patch the registry so that it uses the **Recreate** rollout strategy and runs with only **1** replica:

```
$ oc patch config.imageregistry.operator.openshift.io/cluster --type=merge -p '{"spec":
{"rolloutStrategy":"Recreate","replicas":1}}'
```

2. Provision the PV for the block storage device, and create a PVC for that volume. The requested block volume uses the ReadWriteOnce (RWO) access mode.

3. Edit the registry configuration so that it references the correct PVC.

For instructions about configuring registry storage so that it references the correct PVC, see Configuring the registry for vSphere.

1.2.13. Next steps

- Customize your cluster.
- If necessary, you can opt out of remote health reporting .
- Set up your registry and configure registry storage .

1.3. INSTALLING A CLUSTER ON VSPHERE

In OpenShift Container Platform version 4.5, you can install a cluster on VMware vSphere infrastructure that you provision.

1.3.1. Prerequisites

- Provision persistent storage for your cluster. To deploy a private image registry, your storage must provide ReadWriteMany access modes.
- Review details about the OpenShift Container Platform installation and update processes.
- If you use a firewall, you must configure it to allow the sites that your cluster requires access to.

NOTE

Be sure to also review this site list if you are configuring a proxy.

1.3.2. Internet and Telemetry access for OpenShift Container Platform

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

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

You must have internet access to:

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

IMPORTANT

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

1.3.3. VMware vSphere infrastructure requirements

You must install the OpenShift Container Platform cluster on a VMware vSphere version 6 instance that meets the requirements for the components that you use.

Table 1.9. Minimum supported vSphere version for VMware components

Component	Minimum supported versions	Description
Hypervisor	vSphere 6.5 with HW version 13	This version is the minimum version that Red Hat Enterprise Linux CoreOS (RHCOS) supports. See the Red Hat Enterprise Linux 8 supported hypervisors list.
Networking (NSX-T)	n/a	vSphere 6.5U3 or vSphere 6.7U2+ are required for OpenShift Container Platform. Because previous versions of vSphere with NSX-T are not currently compatible with OpenShift Container Platform, NSX-T is not supported. NSX-T certification is in process and will be supported in a future release.
Storage with in-tree drivers	vSphere 6.5 or vSphere 6.7	This plug-in creates vSphere storage by using the in-tree storage drivers for vSphere included in OpenShift Container Platform and can be used when vSphere CSI drivers are not available.
Storage with vSphere CSI driver	vSphere 6.7U3 and later	This plug-in creates vSphere storage by using the standard Container Storage Interface. The vSphere CSI driver is provided and supported by VMware.

If you use a vSphere version 6.5 instance, consider upgrading to 6.7U2 before you install OpenShift Container Platform.

IMPORTANT

You must ensure that the time on your ESXi hosts is synchronized before you install OpenShift Container Platform. See Edit Time Configuration for a Host in the VMware documentation.

IMPORTANT

> A limitation of using VPC is that the Storage Distributed Resource Scheduler (SDRS) is not supported. See link:https://vmware.github.io/vsphere-storage-for-kubernetes/documentation/faqs.html[vSphere Storage for Kubernetes FAQs] in the VMware documentation.

1.3.4. Machine requirements for a cluster with user-provisioned infrastructure

For a cluster that contains user-provisioned infrastructure, you must deploy all of the required machines.

1.3.4.1. Required machines

The smallest OpenShift Container Platform clusters require the following hosts:

- One temporary bootstrap machine
- Three control plane, or master, machines
- At least two compute machines, which are also known as worker machines

NOTE

The cluster requires the bootstrap machine to deploy the OpenShift Container Platform cluster on the three control plane machines. You can remove the bootstrap machine after you install the cluster.

IMPORTANT

To maintain high availability of your cluster, use separate physical hosts for these cluster machines.

The bootstrap, control plane, and compute machines must use the Red Hat Enterprise Linux CoreOS (RHCOS) as the operating system.

Note that RHCOS is based on Red Hat Enterprise Linux (RHEL) 8 and inherits all of its hardware certifications and requirements. See Red Hat Enterprise Linux technology capabilities and limits .

1.3.4.2. Network connectivity requirements

All the Red Hat Enterprise Linux CoreOS (RHCOS) machines require network in **initramfs** during boot to fetch Ignition config files from the Machine Config Server. During the initial boot, the machines require either a DHCP server or that static IP addresses be set in order to establish a network connection to download their Ignition config files.

1.3.4.3. Minimum resource requirements

Each cluster machine must meet the following minimum requirements:

Machine	Operating System	vCPU	Virtual RAM	Storage
Bootstrap	RHCOS	4	16 GB	120 GB
Control plane	RHCOS	4	16 GB	120 GB
Compute	RHCOS or RHEL 7.6	2	8 GB	120 GB

1.3.4.4. Certificate signing requests management

Because your cluster has limited access to automatic machine management when you use infrastructure that you provision, you must provide a mechanism for approving cluster certificate signing requests (CSRs) after installation. The **kube-controller-manager** only approves the kubelet client CSRs. The **machine-approver** cannot guarantee the validity of a serving certificate that is requested by using kubelet credentials because it cannot confirm that the correct machine issued the request. You must determine and implement a method of verifying the validity of the kubelet serving certificate requests and approving them.

1.3.5. Creating the user-provisioned infrastructure

Before you deploy an OpenShift Container Platform cluster that uses user-provisioned infrastructure, you must create the underlying infrastructure.

Prerequistes

- Review the OpenShift Container Platform 4.x Tested Integrations page before you create the supporting infrastructure for your cluster.

Procedure

1. Configure DHCP or set static IP addresses on each node.
2. Provision the required load balancers.
3. Configure the ports for your machines.
4. Configure DNS.
5. Ensure network connectivity.

1.3.5.1. 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, which runs on each master node after a successful cluster installation, 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.

Table 1.10. All machines to all machines

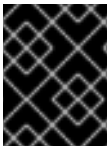
Protocol	Port	Description
ICMP	N/A	Network reachability tests
TCP	**9000**-**9999**	Host level services, including the node exporter on ports **9100**-**9101** and the Cluster Version Operator on port**9099**.
	10250-**10259**	The default ports that Kubernetes reserves
	10256	openshift-sdn
UDP	**4789**	VXLAN and GENEVE
	6081	VXLAN and GENEVE
	9000-**9999**	Host level services, including the node exporter on ports **9100**-**9101**.
TCP/UDP	**30000**-**32767**	Kubernetes NodePort

Table 1.11. All machines to control plane

Protocol	Port	Description
TCP	**2379**-**2380**	etcd server, peer, and metrics ports
	6443	Kubernetes API

Network topology requirements

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



IMPORTANT

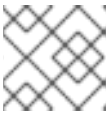
OpenShift Container Platform requires all nodes to have internet access to pull images for platform containers and provide telemetry data to Red Hat.

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.
   - A stateless load balancing algorithm. The options vary based on the load balancer implementation.

   NOTE

   Session persistence is not required for the API load balancer to function properly.

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

   Table 1.12. API load balancer

Port	Back-end machines (pool members)	Internal	External	Description
6443	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.	X	X	Kubernetes API server
22623	Bootstrap and control plane. You remove the bootstrap machine from the load balancer after the bootstrap machine initializes the cluster control plane.	X		Machine Config server

   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.
- 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 1.13. Application Ingress load balancer

Port	Back-end machines (pool members)	Internal	External	Description
443	The machines that run the Ingress router pods, compute, or worker, by default.	X	X	HTTPS traffic
80	The machines that run the Ingress router pods, compute, or worker by default.	X	X	HTTP traffic

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.

Ethernet adaptor hardware address requirements
When provisioning VMs for the cluster, the ethernet interfaces configured for each VM must use a MAC address from the VMware Organizationally Unique Identifier (OUI) allocation ranges:

- **00:05:69:00:00:00** to **00:05:69:FF:FF:FF**
- **00:0c:29:00:00:00** to **00:0c:29:FF:FF:FF**
- **00:1c:14:00:00:00** to **00:1c:14:FF:FF:FF**
- **00:50:56:00:00:00** to **00:50:56:FF:FF:FF**

If a MAC address outside the VMware OUI is used, the cluster installation will not succeed.

1.3.5.2. User-provisioned DNS requirements

The following DNS records are required for an OpenShift Container Platform cluster that uses user-provisioned infrastructure. In each record, **<cluster_name>** is the cluster name and **<base_domain>** is the cluster base domain that you specify in the **install-config.yaml** file. A complete DNS record takes the form: **<component>.<cluster_name>.<base_domain>.**.

Table 1.14. Required DNS records

Component	Record	Description
Kubernetes API	**api.<cluster_name>.<base_domain>.**	This DNS A/AAAA or CNAME record must point to the load balancer for the control plane machines. This record must be resolvable by both clients external to the cluster and from all the nodes within the cluster.
	api-int.<cluster_name>.<base_domain>.	This DNS A/AAAA or CNAME record must point to the load balancer for the control plane machines. This record must be resolvable from all the nodes within the cluster. **IMPORTANT** The API server must be able to resolve the worker nodes by the host names that are recorded in Kubernetes. If it cannot resolve the node names, proxied API calls can fail, and you cannot retrieve logs from Pods.
Routes	***.apps.<cluster_name>.<base_domain>.**	A wildcard DNS A/AAAA or CNAME record that points to the load balancer that targets the machines that run the Ingress router pods, which are the worker nodes by default. This record must be resolvable by both clients external to the cluster and from all the nodes within the cluster.

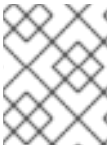
1.3.6. Generating an SSH private key and adding it to the agent

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 to the installation program.

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:

```
$ ssh-keygen -t rsa -b 4096 -N '' \
    -f <path>/<file_name> 1
```

1 Specify the path and file name, such as ~/**.ssh/id_rsa**, of the SSH key. Do not specify an existing SSH key, as it will be overwritten.

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

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

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

Example output

```
Agent pid 31874
```

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

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

Example output

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

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

Next steps

- When you install OpenShift Container Platform, provide the SSH public key to the installation program. If you install a cluster on infrastructure that you provision, you must provide this key to your cluster's machines.

1.3.7. Obtaining the installation program

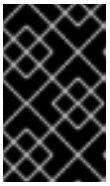
Before you install OpenShift Container Platform, download the installation file on a local computer.

Prerequisites

- You must install the cluster from a computer that uses Linux or macOS.
- You need 500 MB of local disk space to download the installation program.

Procedure

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



IMPORTANT

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

IMPORTANT

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

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

```
$ tar xvf <installation_program>.tar.gz
```

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

1.3.8. Manually creating the installation configuration file

For installations of OpenShift Container Platform that use user-provisioned infrastructure, you manually generate your installation configuration file.

Prerequisites

- Obtain the OpenShift Container Platform installation program and the access token for your cluster.

Procedure

1. Create an installation directory to store your required installation assets in:

```
$ mkdir <installation_directory>
```

IMPORTANT

You must create a 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.

2. Customize the following **install-config.yaml** file template and save it in the **<installation_directory>**.

NOTE

You must name this configuration file **install-config.yaml**.

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 next step of the installation process. You must back it up now.

1.3.8.1. Sample install-config.yaml file for VMware vSphere

You can customize the **install-config.yaml** file to specify more details about your OpenShift Container Platform cluster's platform or modify the values of the required parameters.

```
apiVersion: v1
baseDomain: example.com (1)
compute:
- hyperthreading: Enabled (2) (3)
  name: worker
  replicas: 0 (4)
controlPlane:
  hyperthreading: Enabled (5) (6)
  name: master
  replicas: 3 (7)
metadata:
  name: test (8)
platform:
  vsphere:
    vcenter: your.vcenter.server (9)
    username: username (10)
    password: password (11)
    datacenter: datacenter (12)
    defaultDatastore: datastore (13)
fips: false (14)
pullSecret: '{"auths": ...}' (15)
sshKey: 'ssh-ed25519 AAAA...' (16)
```

(1) The base domain of the cluster. All DNS records must be sub-domains of this base and include the cluster name.

(2) (5) The **controlPlane** section is a single mapping, but the compute section is a sequence of mappings. To meet the requirements of the different data structures, the first line of the **compute** section must begin with a hyphen, **-**, and the first line of the **controlPlane** section must not. Although both sections currently define a single machine pool, it is possible that future versions of OpenShift Container Platform will support defining multiple compute pools during installation. Only one control plane pool is used.

(3) (6) Whether to enable or disable simultaneous multithreading, or **hyperthreading**. By default, simultaneous multithreading is enabled to increase the performance of your machines' cores. You can disable it by setting the parameter value to **Disabled**. If you disable simultaneous multithreading in some cluster machines, you must disable it in all cluster machines.

IMPORTANT

If you disable simultaneous multithreading, ensure that your capacity planning accounts for the dramatically decreased machine performance. Your machines must use at least 8 CPUs and 32 GB of RAM if you disable simultaneous multithreading.

4 You must set the value of the **replicas** parameter to **0**. This parameter controls the number of workers that the cluster creates and manages for you, which are functions that the cluster does not perform when you use user-provisioned infrastructure. You must manually deploy worker machines for the cluster to use before you finish installing OpenShift Container Platform.

7 The number of control plane machines that you add to the cluster. Because the cluster uses this values as the number of etcd endpoints in the cluster, the value must match the number of control plane machines that you deploy.

8 The cluster name that you specified in your DNS records.

9 The fully-qualified host name or IP address of the vCenter server.

10 The name of the user for accessing the server. This user must have at least the roles and privileges that are required for static or dynamic persistent volume provisioning in vSphere.

11 The password associated with the vSphere user.

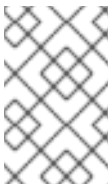
12 The vSphere datacenter.

13 The default vSphere datastore to use.

14 Whether to enable or disable FIPS mode. By default, FIPS mode is not enabled. 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.

15 The pull secret that you obtained from the Pull Secret page on the Red Hat OpenShift Cluster Manager site. 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.

16 The public portion of the default SSH key for the **core** user in Red Hat Enterprise Linux CoreOS (RHCOS).



NOTE

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

1.3.8.2. Configuring the cluster-wide proxy during installation

Production environments can deny direct access to the Internet and instead have an HTTP or HTTPS proxy available. You can configure a new OpenShift Container Platform cluster to use a proxy by configuring the proxy settings in the **install-config.yaml** file.

Prerequisites

- An existing **install-config.yaml** file.
- Review the sites that your cluster requires access to and determine whether any need to bypass the proxy. By default, all cluster egress traffic is proxied, including calls to hosting cloud provider APIs. Add sites to the Proxy object's **spec.noProxy** field to bypass the proxy if necessary.

NOTE

The Proxy object's **status.noProxy** field is populated by default with the instance metadata endpoint (**169.254.169.254**) and with the values of the **networking.machineNetwork[].cidr**, **networking.clusterNetwork[].cidr**, and **networking.serviceNetwork[]** fields from your installation configuration.

Procedure

1. Edit your **install-config.yaml** file and add the proxy settings. For example:

```
apiVersion: v1
baseDomain: my.domain.com
proxy:
  httpProxy: http://<username>:<pswd>@<ip>:<port> 1
  httpsProxy: http://<username>:<pswd>@<ip>:<port> 2
  noProxy: example.com 3
additionalTrustBundle: | 4
    -----BEGIN CERTIFICATE-----
    <MY_TRUSTED_CA_CERT>
    -----END CERTIFICATE-----
...
```

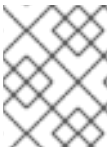
1. A proxy URL to use for creating HTTP connections outside the cluster. The URL scheme must be **http**. If you use an MITM transparent proxy network that does not require additional proxy configuration but requires additional CAs, you must not specify an **httpProxy** value.
2. A proxy URL to use for creating HTTPS connections outside the cluster. If this field is not specified, then **httpProxy** is used for both HTTP and HTTPS connections. The URL scheme must be **http**; **https** is currently not supported. If you use an MITM transparent proxy network that does not require additional proxy configuration but requires additional CAs, you must not specify an **httpsProxy** value.
3. A comma-separated list of destination domain names, domains, IP addresses, or other network CIDRs to exclude proxying. Preface a domain with **.** to include all subdomains of that domain. Use ***** to bypass proxy for all destinations. You must include vCenter's IP address and the IP range that you use for its machines.
4. If provided, the installation program generates a ConfigMap that is named **user-ca-bundle** in the **openshift-config** namespace that contains one or more additional CA certificates that are required for proxying HTTPS connections. The Cluster Network Operator then creates a **trusted-ca-bundle** ConfigMap that merges these contents with the Red Hat Enterprise Linux CoreOS (RHCOS) trust bundle, and this ConfigMap is referenced in the Proxy object's **trustedCA** field. The **additionalTrustBundle** field is required unless the proxy's identity certificate is signed by an authority from the RHCOS trust bundle. If you use an MITM transparent proxy network that does not require additional proxy configuration but requires additional CAs, you must provide the MITM CA certificate.

NOTE

The installation program does not support the proxy **readinessEndpoints** field.

2. Save the file and reference it when installing OpenShift Container Platform.

The installation program creates a cluster-wide proxy that is named **cluster** that uses the proxy settings in the provided **install-config.yaml** file. If no proxy settings are provided, a **cluster** Proxy object is still created, but it will have a nil **spec**.

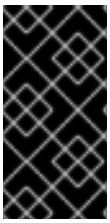


NOTE

Only the Proxy object named **cluster** is supported, and no additional proxies can be created.

1.3.9. Creating the Kubernetes manifest and Ignition config files

Because you must modify some cluster definition files and manually start the cluster machines, you must generate the Kubernetes manifest and Ignition config files that the cluster needs to make its machines.



IMPORTANT

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

Prerequisites

- Obtain the OpenShift Container Platform installation program.
- Create the **install-config.yaml** installation configuration file.

Procedure

1. Generate the Kubernetes manifests for the cluster:

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

Example output

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

1 For **<installation_directory>**, specify the installation directory that contains the **install-config.yaml** file you created.

Because you create your own compute machines later in the installation process, you can safely ignore this warning.

2. Remove the Kubernetes manifest files that define the control plane machines and compute machineSets:

```
$ rm -f openshift/99_openshift-cluster-api_master-machines-*.yaml openshift/99_openshift-cluster-api_worker-machineset-*.yaml
```

Because you create and manage these resources yourself, you do not have to initialize them.

- You can preserve the MachineSet files to create compute machines by using the machine API, but you must update references to them to match your environment.

3. Modify the **<installation_directory>/manifests/cluster-scheduler-02-config.yml** Kubernetes manifest file to prevent Pods from being scheduled on the control plane machines:

   a. Open the **<installation_directory>/manifests/cluster-scheduler-02-config.yml** file.

   b. Locate the **mastersSchedulable** parameter and set its value to **False**.

   c. Save and exit the file.

NOTE

Currently, due to a Kubernetes limitation, router Pods running on control plane machines will not be reachable by the ingress load balancer. This step might not be required in a future minor version of OpenShift Container Platform.

4. Obtain the Ignition config files:

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

1 For **<installation_directory>**, specify the same installation directory.

The following files are generated in the directory:

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

1.3.10. Creating Red Hat Enterprise Linux CoreOS (RHCOS) machines in vSphere

Before you install a cluster that contains user-provisioned infrastructure on VMware vSphere, you must create RHCOS machines on vSphere hosts for it to use.

Prerequisites

- Obtain the Ignition config files for your cluster.
- Have access to an HTTP server that you can access from your computer and that the machines that you create can access.
- Create a vSphere cluster.

Procedure

1. Upload the bootstrap Ignition config file, which is named **<installation_directory>/bootstrap.ign**, that the installation program created to your HTTP server. Note the URL of this file.
   You must host the bootstrap Ignition config file because it is too large to fit in a vApp property.

2. Save the following secondary Ignition config file for your bootstrap node to your computer as **<installation_directory>/append-bootstrap.ign**.

```
{
  "ignition": {
    "config": {
      "append": [
        {
          "source": "<bootstrap_ignition_config_url>", 1
          "verification": {}
        }
      ]
    },
    "timeouts": {},
    "version": "2.1.0"
  },
  "networkd": {},
  "passwd": {},
  "storage": {},
  "systemd": {}
}
```

   1 Specify the URL of the bootstrap Ignition config file that you hosted.

   When you create the Virtual Machine (VM) for the bootstrap machine, you use this Ignition config file.

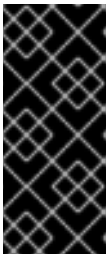
3. Convert the master, worker, and secondary bootstrap Ignition config files to Base64 encoding. For example, if you use a Linux operating system, you can use the **base64** command to encode the files.

```
$ base64 -w0 <installation_directory>/master.ign > <installation_directory>/master.64
```

```
$ base64 -w0 <installation_directory>/worker.ign > <installation_directory>/worker.64
```

```
$ base64 -w0 <installation_directory>/append-bootstrap.ign >
<installation_directory>/append-bootstrap.64
```

4. Obtain the RHCOS OVA image from the Product Downloads page on the Red Hat customer portal or the RHCOS image mirror page.

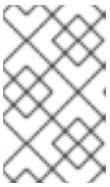


IMPORTANT

The RHCOS images might not change with every release of OpenShift Container Platform. You must download an image with the highest version that is less than or equal to the OpenShift Container Platform version that you install. Use the image version that matches your OpenShift Container Platform version if it is available.

The file name contains the OpenShift Container Platform version number in the format **rhcos-<version>-vmware.<architecture>.ova**.

5. In the vSphere Client, create a folder in your datacenter to store your VMs.
   a. Click the **VMs and Templates** view.
   b. Right-click the name of your datacenter.
   c. Click **New Folder → New VM and Template Folder**.
   d. In the window that is displayed, enter the folder name. The folder name must match the cluster name that you specified in the **install-config.yaml** file.
6. In the vSphere Client, create a template for the OVA image.



NOTE

In the following steps, you use the same template for all of your cluster machines and provide the location for the Ignition config file for that machine type when you provision the VMs.

   a. From the **Hosts and Clusters** tab, right-click your cluster's name and click **Deploy OVF Template**.
   b. On the **Select an OVF** tab, specify the name of the RHCOS OVA file that you downloaded.
   c. On the **Select a name and folder** tab, set a **Virtual machine name**, such as RHCOS, click the name of your vSphere cluster, and select the folder you created in the previous step.
   d. On the **Select a compute resource** tab, click the name of your vSphere cluster.
   e. On the **Select storage** tab, configure the storage options for your VM.
      - Select **Thin Provision** or **Thick Provision**, based on your storage preferences.
      - Select the datastore that you specified in your **install-config.yaml** file.
   f. On the **Select network** tab, specify the network that you configured for the cluster, if available.
   g. If you plan to use the same template for all cluster machine types, do not specify values on the **Customize template** tab.
7. After the template deploys, deploy a VM for a machine in the cluster.
   a. Right-click the template's name and click **Clone → Clone to Virtual Machine**

b. On the **Select a name and folder** tab, specify a name for the VM. You might include the machine type in the name, such as **control-plane-0** or **compute-1**.

c. On the **Select a name and folder** tab, select the name of the folder that you created for the cluster.

d. On the **Select a compute resource** tab, select the name of a host in your datacenter.

e. Optional: On the **Select storage** tab, customize the storage options.

f. On the **Select clone options**, select **Customize this virtual machine's hardware**.

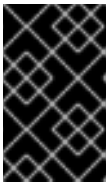
g. On the **Customize hardware** tab, click **VM Options → Advanced**.

- Optional: In the event of cluster performance issues, from the **Latency Sensitivity** list, select **High**.
- Click **Edit Configuration**, and on the **Configuration Parameters** window, click **Add Configuration Params**. Define the following parameter names and values:
  - **guestinfo.ignition.config.data**: Paste the contents of the base64-encoded Ignition config file for this machine type.
  - **guestinfo.ignition.config.data.encoding**: Specify **base64**.
  - **disk.EnableUUID**: Specify **TRUE**.
- Alternatively, prior to powering on the virtual machine add via vApp properties:
  - Navigate to a virtual machine from the vCenter Server inventory.
  - On the **Configure** tab, expand **Settings** and select **vApp options.**
  - Scroll down and under **Properties** apply the configurations from above.

h. In the **Virtual Hardware** panel of the **Customize hardware** tab, modify the specified values as required. Ensure that the amount of RAM, CPU, and disk storage meets the minimum requirements for the machine type.

i. Complete the configuration and power on the VM.

8. Create the rest of the machines for your cluster by following the preceding steps for each machine.

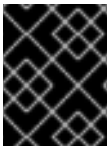


IMPORTANT

You must create the bootstrap and control plane machines at this time. Because some pods are deployed on compute machines by default, also create at least two compute machine before you install the cluster.

1.3.11. Installing the CLI by downloading the binary

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



IMPORTANT

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

1.3.11.1. Installing the CLI on Linux

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

Procedure

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

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

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

```
$ echo $PATH
```

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

```
$ oc <command>
```

1.3.11.2. Installing the CLI on Windows

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

Procedure

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

```
C:\> path
```

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

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

1.3.11.3. Installing the CLI on macOS

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

Procedure

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

   ```
   $ echo $PATH
   ```

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

```
$ oc <command>
```

1.3.12. Creating the cluster

To create the OpenShift Container Platform cluster, you wait for the bootstrap process to complete on the machines that you provisioned by using the Ignition config files that you generated with the installation program.

Prerequisites

- Create the required infrastructure for the cluster.
- You obtained the installation program and generated the Ignition config files for your cluster.
- You used the Ignition config files to create RHCOS machines for your cluster.
- Your machines have direct internet access.

Procedure

1. Monitor the bootstrap process:

   ```
   $ ./openshift-install --dir=<installation_directory> wait-for bootstrap-complete \ 1
       --log-level=info 2
   ```

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

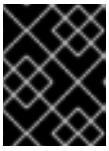
   2 To view different installation details, specify **warn**, **debug**, or **error** instead of **info**.

Example output

```
INFO Waiting up to 30m0s for the Kubernetes API at https://api.test.example.com:6443...
INFO API v1.18.3 up
INFO Waiting up to 30m0s for bootstrapping to complete...
INFO It is now safe to remove the bootstrap resources
```

The command succeeds when the Kubernetes API server signals that it has been bootstrapped on the control plane machines.

2. After bootstrap process is complete, remove the bootstrap machine from the load balancer.



IMPORTANT

You must remove the bootstrap machine from the load balancer at this point. You can also remove or reformat the machine itself.

1.3.13. Logging in to the cluster

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

Prerequisites

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

Procedure

1. Export the **kubeadmin** credentials:

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

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

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

```
$ oc whoami
```

Example output

```
system:admin
```

1.3.14. Approving the CSRs for your machines

When you add machines to a cluster, two pending certificates signing request (CSRs) are generated for each machine that you added. You must confirm that these CSRs are approved or, if necessary, approve them yourself.

Prerequisites

- You added machines to your cluster.

Procedure

1. Confirm that the cluster recognizes the machines:

```
$ oc get nodes
```

Example output

```
NAME      STATUS    ROLES   AGE  VERSION
master-0  Ready     master  63m  v1.18.3
master-1  Ready     master  63m  v1.18.3
master-2  Ready     master  64m  v1.18.3
worker-0  NotReady  worker  76s  v1.18.3
worker-1  NotReady  worker  70s  v1.18.3
```

The output lists all of the machines that you created.

2. Review the pending certificate signing requests (CSRs) and ensure that the you see a client and server request with **Pending** or **Approved** status for each machine that you added to the cluster:

```
$ oc get csr
```

Example output

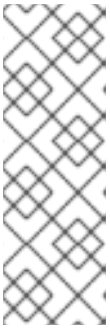
```
NAME        AGE     REQUESTOR                                                   CONDITION
csr-8b2br   15m     system:serviceaccount:openshift-machine-config-operator:node-
bootstrapper   Pending (1)
csr-8vnps   15m     system:serviceaccount:openshift-machine-config-operator:node-
bootstrapper   Pending
csr-bfd72   5m26s   system:node:ip-10-0-50-126.us-east-2.compute.internal
Pending (2)
csr-c57lv   5m26s   system:node:ip-10-0-95-157.us-east-2.compute.internal
Pending
...
```

(1) A client request CSR.

(2) A server request CSR.

In this example, two machines are joining the cluster. You might see more approved CSRs in the list.

3. If the CSRs were not approved, after all of the pending CSRs for the machines you added are in **Pending** status, approve the CSRs for your cluster machines:



NOTE

Because the CSRs rotate automatically, approve your CSRs within an hour of adding the machines to the cluster. If you do not approve them within an hour, the certificates will rotate, and more than two certificates will be present for each node. You must approve all of these certificates. After you approve the initial CSRs, the subsequent node client CSRs are automatically approved by the cluster **kube-controller-manager**. You must implement a method of automatically approving the kubelet serving certificate requests.

- To approve them individually, run the following command for each valid CSR:

```
$ oc adm certificate approve <csr_name> 1
```

1 **<csr_name>** is the name of a CSR from the list of current CSRs.

- To approve all pending CSRs, run the following command:

```
$ oc get csr -o go-template='{{range .items}}{{if not .status}}{{.metadata.name}}{{"\n"}}
{{end}}{{end}}' | xargs oc adm certificate approve
```

1.3.15. Initial Operator configuration

After the control plane initializes, you must immediately configure some Operators so that they all become available.

Prerequisites

- Your control plane has initialized.

Procedure

1. Watch the cluster components come online:

```
$ watch -n5 oc get clusteroperators
```

Example output

```
NAME                       VERSION   AVAILABLE   PROGRESSING   DEGRADED
SINCE
authentication             4.5.4     True        False         False     69s
cloud-credential           4.5.4     True        False         False     12m
cluster-autoscaler         4.5.4     True        False         False     11m
console                    4.5.4     True        False         False     46s
dns                        4.5.4     True        False         False     11m
image-registry             4.5.4     True        False         False     5m26s
ingress                    4.5.4     True        False         False     5m36s
kube-apiserver             4.5.4     True        False         False     8m53s
kube-controller-manager    4.5.4     True        False         False     7m24s
kube-scheduler             4.5.4     True        False         False     12m
machine-api                4.5.4     True        False         False     12m
machine-config             4.5.4     True        False         False     7m36s
```

```
marketplace                 4.5.4     True        False         False      7m54m
monitoring                  4.5.4     True        False         False      7h54s
network                     4.5.4     True        False         False      5m9s
node-tuning                 4.5.4     True        False         False      11m
openshift-apiserver         4.5.4     True        False         False      11m
openshift-controller-manager        4.5.4     True        False         False      5m943s
openshift-samples           4.5.4     True        False         False      3m55s
operator-lifecycle-manager          4.5.4     True        False         False      11m
operator-lifecycle-manager-catalog  4.5.4     True        False         False      11m
service-ca                  4.5.4     True        False         False      11m
service-catalog-apiserver           4.5.4     True        False         False      5m26s
service-catalog-controller-manager  4.5.4     True        False         False      5m25s
storage                     4.5.4     True        False         False      5m30s
```

2. Configure the Operators that are not available.

1.3.15.1. Image registry removed during installation

On platforms that do not provide shareable object storage, the OpenShift Image Registry Operator bootstraps itself as **Removed**. This allows **openshift-installer** to complete installations on these platform types.

After installation, you must edit the Image Registry Operator configuration to switch the **ManagementState** from **Removed** to **Managed**.

NOTE

The Prometheus console provides an **ImageRegistryRemoved** alert, for example:

"Image Registry has been removed. **ImageStreamTags**, **BuildConfigs** and **DeploymentConfigs** which reference **ImageStreamTags** may not work as expected. Please configure storage and update the config to **Managed** state by editing configs.imageregistry.operator.openshift.io."

1.3.15.2. Image registry storage configuration

The **image-registry** Operator is not initially available for platforms that do not provide default storage. After installation, you must configure your registry to use storage so that the Registry Operator is made available.

Instructions are shown for configuring a PersistentVolume, which is required for production clusters. Where applicable, instructions are shown for configuring an empty directory as the storage location, which is available for only non-production clusters.

Additional instructions are provided for allowing the image registry to use block storage types by using the **Recreate** rollout strategy during upgrades.

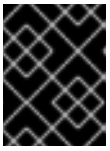
1.3.15.2.1. Configuring registry storage for VMware vSphere

As a cluster administrator, following installation you must configure your registry to use storage.

Prerequisites

- Cluster administrator permissions.

- A cluster on VMware vSphere.
- Provision persistent storage for your cluster. To deploy a private image registry, your storage must provide ReadWriteMany access mode.



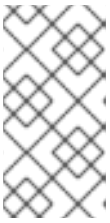
IMPORTANT

vSphere volumes do not support the **ReadWriteMany** access mode. You must use a different storage backend, such as **NFS**, to configure the registry storage.

- Must have "100Gi" capacity.

Procedure

1. To configure your registry to use storage, change the **spec.storage.pvc** in the **configs.imageregistry/cluster** resource.

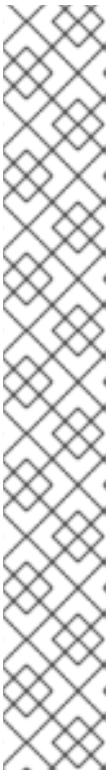


NOTE

When using shared storage such as NFS, it is strongly recommended to use the **supplementalGroups** strategy, which dictates the allowable supplemental groups for the Security Context, rather than the **fsGroup** ID. Refer to the NFS **Group IDs** documentation for details.

2. Verify you do not have a registry Pod:

```
$ oc get pod -n openshift-image-registry
```



NOTE

- If the storage type is **emptyDIR**, the replica number cannot be greater than **1**.
- If the storage type is **NFS**, you must enable the **no_wdelay** and **root_squash** mount options. For example:

```
# cat /etc/exports
```

Example output

```
/mnt/data *(rw,sync,no_wdelay,root_squash,insecure,fsid=0)
```

```
sh-4.2# exportfs -rv
```

Example output

```
exporting *:/mnt/data
```

3. Check the registry configuration:

```
$ oc edit configs.imageregistry.operator.openshift.io
```

- **Example registry configuration**

```
storage:
  pvc:
    claim:
```

Leave the **claim** field blank to allow the automatic creation of an **image-registry-storage** PVC.

4. Optional: Add a new storage class to a PV:

   a. Create the PV:

```
$ oc create -f -
```

```
apiVersion: v1
kind: PersistentVolume
metadata:
  name: image-registry-pv
spec:
  accessModes:
    - ReadWriteMany
  capacity:
      storage: 100Gi
  nfs:
    path: /registry
    server: 172.16.231.181
  persistentVolumeReclaimPolicy: Retain
  storageClassName: nfs01
```

```
$ oc get pv
```

   b. Create the PVC:

```
$ oc create -n openshift-image-registry -f -
```

```
apiVersion: "v1"
kind: "PersistentVolumeClaim"
metadata:
  name: "image-registry-pvc"
spec:
  accessModes:
    - ReadWriteMany
  resources:
    requests:
      storage: 100Gi
  storageClassName: nfs01
  volumeMode: Filesystem
```

```
$ oc get pvc -n openshift-image-registry
```

   Finally, add the name of your PVC:

```
$ oc edit configs.imageregistry.operator.openshift.io -o yaml
```

```
storage:
  pvc:
    claim: image-registry-pvc (1)
```

(1) Creating a custom PVC allows you to leave the **claim** field blank for default automatic creation of an **image-registry-storage** PVC.

5. Check the **clusteroperator** status:

```
$ oc get clusteroperator image-registry
```

1.3.15.2.2. Configuring storage for the image registry in non-production clusters

You must configure storage for the image registry Operator. For non-production clusters, you can set the image registry to an empty directory. If you do so, all images are lost if you restart the registry.

Procedure

1. To set the image registry storage to an empty directory:

```
$ oc patch configs.imageregistry.operator.openshift.io cluster --type merge --patch '{"spec":{"storage":{"emptyDir":{}}}}'
```

WARNING

Configure this option for only non-production clusters.

If you run this command before the Image Registry Operator initializes its components, the **oc patch** command fails with the following error:

```
Error from server (NotFound): configs.imageregistry.operator.openshift.io "cluster" not found
```

Wait a few minutes and run the command again.

2. Ensure that your registry is set to managed to enable building and pushing of images.

- Run:

```
$ oc edit configs.imageregistry/cluster
```

Then, change the line

```
managementState: Removed
```

to

```
managementState: Managed
```

1.3.15.2.3. Configuring block registry storage for VMware vSphere

To allow the image registry to use block storage types such as vSphere Virtual Machine Disk (VMDK) during upgrades as a cluster administrator, you can use the **Recreate** rollout strategy.

Procedure

1. To set the image registry storage as a block storage type, patch the registry so that it uses the **Recreate** rollout strategy and runs with only **1** replica:

```
$ oc patch config.imageregistry.operator.openshift.io/cluster --type=merge -p '{"spec":
{"rolloutStrategy":"Recreate","replicas":1}}'
```

2. Provision the PV for the block storage device, and create a PVC for that volume. The requested block volume uses the ReadWriteOnce (RWO) access mode.

3. Edit the registry configuration so that it references the correct PVC.

For instructions about configuring registry storage so that it references the correct PVC, see Configuring the registry for vSphere.

1.3.16. Completing installation on user-provisioned infrastructure

After you complete the Operator configuration, you can finish installing the cluster on infrastructure that you provide.

Prerequisites

- Your control plane has initialized.
- You have completed the initial Operator configuration.

Procedure

1. If instead of installing a new cluster you are upgrading an existing cluster from version 4.3 or from versions 4.4.0 to 4.4.11, run:

```
$ oc edit configs.samples cluster
```

   Then, change the line

```
managementState: Removed
```

   to

```
managementState: Managed
```

2. Confirm that all the cluster components are online:

```
$ watch -n5 oc get clusteroperators
```

Example output

```
NAME                                       VERSION   AVAILABLE   PROGRESSING   DEGRADED
SINCE
authentication                             4.5.4     True        False         False      7m56s
cloud-credential                           4.5.4     True        False         False      31m
cluster-autoscaler                         4.5.4     True        False         False      16m
console                                    4.5.4     True        False         False      10m
csi-snapshot-controller                    4.5.4     True        False         False      16m
dns                                        4.5.4     True        False         False      22m
etcd                                       4.5.4     False       False         False      25s
image-registry                             4.5.4     True        False         False      16m
ingress                                    4.5.4     True        False         False      16m
insights                                   4.5.4     True        False         False      17m
kube-apiserver                             4.5.4     True        False         False      19m
kube-controller-manager                    4.5.4     True        False         False      20m
kube-scheduler                             4.5.4     True        False         False      20m
kube-storage-version-migrator              4.5.4     True        False         False      16m
machine-api                                4.5.4     True        False         False      22m
machine-config                             4.5.4     True        False         False      22m
marketplace                                4.5.4     True        False         False      16m
monitoring                                 4.5.4     True        False         False      10m
network                                    4.5.4     True        False         False      23m
node-tuning                                4.5.4     True        False         False      23m
openshift-apiserver                        4.5.4     True        False         False      17m
openshift-controller-manager               4.5.4     True        False         False      15m
openshift-samples                          4.5.4     True        False         False      16m
operator-lifecycle-manager                 4.5.4     True        False         False      22m
operator-lifecycle-manager-catalog         4.5.4     True        False         False      22m
operator-lifecycle-manager-packageserver   4.5.4     True        False         False      18m
service-ca                                 4.5.4     True        False         False      23m
service-catalog-apiserver                  4.5.4     True        False         False      23m
service-catalog-controller-manager         4.5.4     True        False         False      23m
storage                                    4.5.4     True        False         False      17m
```

When all of the cluster Operators are **AVAILABLE**, you can complete the installation.

3. Monitor for cluster completion:

```
$ ./openshift-install --dir=<installation_directory> wait-for install-complete 1
```

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

Example output

```
INFO Waiting up to 30m0s for the cluster to initialize...
```

The command succeeds when the Cluster Version Operator finishes deploying the OpenShift Container Platform cluster from Kubernetes API server.

IMPORTANT

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

4. Confirm that the Kubernetes API server is communicating with the Pods.

   a. To view a list of all Pods, use the following command:

   ```
   $ oc get pods --all-namespaces
   ```

   Example output

   ```
   NAMESPACE                        NAME                                        READY   STATUS
   RESTARTS   AGE
   openshift-apiserver-operator     openshift-apiserver-operator-85cb746d55-zqhs8   1/1
   Running   1          9m
   openshift-apiserver              apiserver-67b9g                               1/1     Running     0
   3m
   openshift-apiserver              apiserver-ljcmx                               1/1     Running     0
   1m
   openshift-apiserver              apiserver-z25h4                               1/1     Running     0
   2m
   openshift-authentication-operator authentication-operator-69d5d8bf84-vh2n8       1/1
   Running     0          5m
   ...
   ```

   b. View the logs for a Pod that is listed in the output of the previous command by using the following command:

   ```
   $ oc logs <pod_name> -n <namespace> 1
   ```

   1 Specify the Pod name and namespace, as shown in the output of the previous command.

   If the Pod logs display, the Kubernetes API server can communicate with the cluster machines.

1.3.17. Next steps

- Customize your cluster.
- If necessary, you can opt out of remote health reporting .
- Set up your registry and configure registry storage .

1.4. INSTALLING A CLUSTER ON VSPHERE WITH NETWORK CUSTOMIZATIONS

In OpenShift Container Platform version 4.5, you can install a cluster on VMware vSphere infrastructure that you provision with customized network configuration options. By customizing your network

configuration, your cluster can coexist with existing IP address allocations in your environment and integrate with existing MTU and VXLAN configurations.

You must set most of the network configuration parameters during installation, and you can modify only **kubeProxy** configuration parameters in a running cluster.

1.4.1. Prerequisites

- Review details about the OpenShift Container Platform installation and update processes.
- If you use a firewall, you must configure it to access Red Hat Insights .

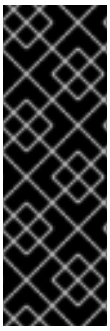
1.4.2. Internet and Telemetry access for OpenShift Container Platform

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

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

You must have internet access to:

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



IMPORTANT

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

1.4.3. VMware vSphere infrastructure requirements

You must install the OpenShift Container Platform cluster on a VMware vSphere version 6 instance that meets the requirements for the components that you use.

Table 1.15. Minimum supported vSphere version for VMware components

Component	Minimum supported versions	Description

Component	Minimum supported versions	Description
Hypervisor	vSphere 6.5 with HW version 13	This version is the minimum version that Red Hat Enterprise Linux CoreOS (RHCOS) supports. See the Red Hat Enterprise Linux 8 supported hypervisors list.
Networking (NSX-T)	n/a	vSphere 6.5U3 or vSphere 6.7U2+ are required for OpenShift Container Platform. Because previous versions of vSphere with NSX-T are not currently compatible with OpenShift Container Platform, NSX-T is not supported. NSX-T certification is in process and will be supported in a future release.
Storage with in-tree drivers	vSphere 6.5 or vSphere 6.7	This plug-in creates vSphere storage by using the in-tree storage drivers for vSphere included in OpenShift Container Platform and can be used when vSphere CSI drivers are not available.
Storage with vSphere CSI driver	vSphere 6.7U3 and later	This plug-in creates vSphere storage by using the standard Container Storage Interface. The vSphere CSI driver is provided and supported by VMware.

If you use a vSphere version 6.5 instance, consider upgrading to 6.7U2 before you install OpenShift Container Platform.

IMPORTANT

You must ensure that the time on your ESXi hosts is synchronized before you install OpenShift Container Platform. See Edit Time Configuration for a Host in the VMware documentation.

IMPORTANT

A limitation of using VPC is that the Storage Distributed Resource Scheduler (SDRS) is not supported. See link:https://vmware.github.io/vsphere-storage-for-kubernetes/documentation/faqs.html[vSphere Storage for Kubernetes FAQs] in the VMware documentation.

1.4.4. Machine requirements for a cluster with user-provisioned infrastructure

For a cluster that contains user-provisioned infrastructure, you must deploy all of the required machines.

1.4.4.1. Required machines

The smallest OpenShift Container Platform clusters require the following hosts:

- One temporary bootstrap machine
- Three control plane, or master, machines
- At least two compute machines, which are also known as worker machines

NOTE

The cluster requires the bootstrap machine to deploy the OpenShift Container Platform cluster on the three control plane machines. You can remove the bootstrap machine after you install the cluster.

IMPORTANT

To maintain high availability of your cluster, use separate physical hosts for these cluster machines.

The bootstrap, control plane, and compute machines must use the Red Hat Enterprise Linux CoreOS (RHCOS) as the operating system.

Note that RHCOS is based on Red Hat Enterprise Linux (RHEL) 8 and inherits all of its hardware certifications and requirements. See Red Hat Enterprise Linux technology capabilities and limits .

1.4.4.2. Network connectivity requirements

All the Red Hat Enterprise Linux CoreOS (RHCOS) machines require network in **initramfs** during boot to fetch Ignition config files from the Machine Config Server. During the initial boot, the machines require either a DHCP server or that static IP addresses be set in order to establish a network connection to download their Ignition config files.

1.4.4.3. Minimum resource requirements

Each cluster machine must meet the following minimum requirements:

Machine	Operating System	vCPU	Virtual RAM	Storage
Bootstrap	RHCOS	4	16 GB	120 GB
Control plane	RHCOS	4	16 GB	120 GB
Compute	RHCOS or RHEL 7.6	2	8 GB	120 GB

1.4.4.4. Certificate signing requests management

Because your cluster has limited access to automatic machine management when you use infrastructure that you provision, you must provide a mechanism for approving cluster certificate signing requests (CSRs) after installation. The **kube-controller-manager** only approves the kubelet client CSRs. The **machine-approver** cannot guarantee the validity of a serving certificate that is requested by using kubelet credentials because it cannot confirm that the correct machine issued the request. You must determine and implement a method of verifying the validity of the kubelet serving certificate requests and approving them.

1.4.5. Creating the user-provisioned infrastructure

Before you deploy an OpenShift Container Platform cluster that uses user-provisioned infrastructure, you must create the underlying infrastructure.

Prerequistes

- Review the OpenShift Container Platform 4.x Tested Integrations page before you create the supporting infrastructure for your cluster.

Procedure

1. Configure DHCP or set static IP addresses on each node.
2. Provision the required load balancers.
3. Configure the ports for your machines.
4. Configure DNS.
5. Ensure network connectivity.

1.4.5.1. 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, which runs on each master node after a successful cluster installation, 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.

Table 1.16. All machines to all machines

Protocol	Port	Description
ICMP	N/A	Network reachability tests
TCP	**9000**-**9999**	Host level services, including the node exporter on ports **9100**-**9101** and the Cluster Version Operator on port**9099**.
	10250-**10259**	The default ports that Kubernetes reserves
	10256	openshift-sdn
UDP	**4789**	VXLAN and GENEVE
	6081	VXLAN and GENEVE
	9000-**9999**	Host level services, including the node exporter on ports **9100**-**9101**.
TCP/UDP	**30000**-**32767**	Kubernetes NodePort

Table 1.17. All machines to control plane

Protocol	Port	Description
TCP	**2379**-**2380**	etcd server, peer, and metrics ports
	6443	Kubernetes API

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.
   - A stateless load balancing algorithm. The options vary based on the load balancer implementation.

NOTE

Session persistence is not required for the API load balancer to function properly.

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

Table 1.18. API load balancer

Port	Back-end machines (pool members)	Internal	External	Description
6443	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.	X	X	Kubernetes API server
22623	Bootstrap and control plane. You remove the bootstrap machine from the load balancer after the bootstrap machine initializes the cluster control plane.	X		Machine Config server

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.
   - 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:

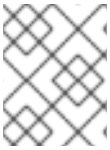
Table 1.19. Application Ingress load balancer

Port	Back-end machines (pool members)	Internal	External	Description
443	The machines that run the Ingress router pods, compute, or worker, by default.	X	X	HTTPS traffic

Port	Back-end machines (pool members)	Internal	External	Description
80	The machines that run the Ingress router pods, compute, or worker by default.	X	X	HTTP traffic

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.



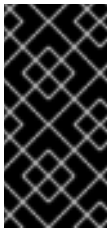
NOTE

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.

1.4.5.2. User-provisioned DNS requirements

The following DNS records are required for an OpenShift Container Platform cluster that uses user-provisioned infrastructure. In each record, **<cluster_name>** is the cluster name and **<base_domain>** is the cluster base domain that you specify in the **install-config.yaml** file. A complete DNS record takes the form: **<component>.<cluster_name>.<base_domain>.**.

Table 1.20. Required DNS records

Component	Record	Description
Kubernetes API	**api.<cluster_name>.<base_domain>.**	This DNS A/AAAA or CNAME record must point to the load balancer for the control plane machines. This record must be resolvable by both clients external to the cluster and from all the nodes within the cluster.
	api-int.<cluster_name>.<base_domain>.	This DNS A/AAAA or CNAME record must point to the load balancer for the control plane machines. This record must be resolvable from all the nodes within the cluster.  **IMPORTANT** The API server must be able to resolve the worker nodes by the host names that are recorded in Kubernetes. If it cannot resolve the node names, proxied API calls can fail, and you cannot retrieve logs from Pods.
Routes	***.apps.<cluster_name>.<base_domain>.**	A wildcard DNS A/AAAA or CNAME record that points to the load balancer that targets the machines that run the Ingress router pods, which are the worker nodes by default. This record must be resolvable by both clients external to the cluster and from all the nodes within the cluster.

1.4.6. 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 to the installation program.

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:

```
$ ssh-keygen -t rsa -b 4096 -N '' \
    -f <path>/<file_name> 1
```

1 Specify the path and file name, such as ~/**.ssh/id_rsa**, of the SSH key. Do not specify an existing SSH key, as it will be overwritten.

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

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

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

   Example output

```
Agent pid 31874
```

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

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

   Example output

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

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

Next steps

Next steps

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

1.4.7. Obtaining the installation program

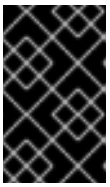
Before you install OpenShift Container Platform, download the installation file on a local computer.

Prerequisites

- You must install the cluster from a computer that uses Linux or macOS.
- You need 500 MB of local disk space to download the installation program.

Procedure

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



IMPORTANT

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

IMPORTANT

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

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

```
$ tar xvf <installation_program>.tar.gz
```

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

1.4.8. Manually creating the installation configuration file

For installations of OpenShift Container Platform that use user-provisioned infrastructure, you manually generate your installation configuration file.

Prerequisites

- Obtain the OpenShift Container Platform installation program and the access token for your cluster.

Procedure

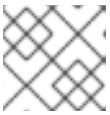
1. Create an installation directory to store your required installation assets in:

```
$ mkdir <installation_directory>
```

IMPORTANT

You must create a 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.

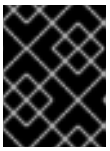
2. Customize the following **install-config.yaml** file template and save it in the **<installation_directory>**.



NOTE

You must name this configuration file **install-config.yaml**.

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 next step of the installation process. You must back it up now.

1.4.8.1. Sample install-config.yaml file for VMware vSphere

You can customize the **install-config.yaml** file to specify more details about your OpenShift Container Platform cluster's platform or modify the values of the required parameters.

```
apiVersion: v1
baseDomain: example.com (1)
compute:
- hyperthreading: Enabled (2) (3)
  name: worker
  replicas: 0 (4)
controlPlane:
  hyperthreading: Enabled (5) (6)
  name: master
  replicas: 3 (7)
metadata:
  name: test (8)
platform:
  vsphere:
    vcenter: your.vcenter.server (9)
```

```
    username: username 10
    password: password 11
    datacenter: datacenter 12
    defaultDatastore: datastore 13
fips: false 14
pullSecret: '{"auths": ...}' 15
sshKey: 'ssh-ed25519 AAAA...' 16
```

1 The base domain of the cluster. All DNS records must be sub-domains of this base and include the cluster name.

2 5 The **controlPlane** section is a single mapping, but the compute section is a sequence of mappings. To meet the requirements of the different data structures, the first line of the **compute** section must begin with a hyphen, **-**, and the first line of the **controlPlane** section must not. Although both sections currently define a single machine pool, it is possible that future versions of OpenShift Container Platform will support defining multiple compute pools during installation. Only one control plane pool is used.

3 6 Whether to enable or disable simultaneous multithreading, or **hyperthreading**. By default, simultaneous multithreading is enabled to increase the performance of your machines' cores. You can disable it by setting the parameter value to **Disabled**. If you disable simultaneous multithreading in some cluster machines, you must disable it in all cluster machines.

IMPORTANT

If you disable simultaneous multithreading, ensure that your capacity planning accounts for the dramatically decreased machine performance. Your machines must use at least 8 CPUs and 32 GB of RAM if you disable simultaneous multithreading.

4 You must set the value of the **replicas** parameter to **0**. This parameter controls the number of workers that the cluster creates and manages for you, which are functions that the cluster does not perform when you use user-provisioned infrastructure. You must manually deploy worker machines for the cluster to use before you finish installing OpenShift Container Platform.

7 The number of control plane machines that you add to the cluster. Because the cluster uses this values as the number of etcd endpoints in the cluster, the value must match the number of control plane machines that you deploy.

8 The cluster name that you specified in your DNS records.

9 The fully-qualified host name or IP address of the vCenter server.

10 The name of the user for accessing the server. This user must have at least the roles and privileges that are required for static or dynamic persistent volume provisioning in vSphere.

11 The password associated with the vSphere user.

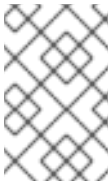
12 The vSphere datacenter.

13 The default vSphere datastore to use.

14 Whether to enable or disable FIPS mode. By default, FIPS mode is not enabled. 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.

15 The pull secret that you obtained from the Pull Secret page on the Red Hat OpenShift Cluster Manager site. This pull secret allows you to authenticate with the services that are provided by the

16 The public portion of the default SSH key for the **core** user in Red Hat Enterprise Linux CoreOS (RHCOS).

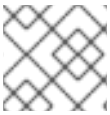


NOTE

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

1.4.8.2. Network configuration parameters

You can modify your cluster network configuration parameters in the **install-config.yaml** configuration file. The following table describes the parameters.



NOTE

You cannot modify these parameters in the **install-config.yaml** file after installation.

Table 1.21. Required network parameters

Parameter	Description	Value
networking.networkType	The default Container Network Interface (CNI) network provider plug-in to deploy. The **OpenShiftSDN** plug-in is the only plug-in supported in OpenShift Container Platform 4.5.	The default value is **OpenShiftSDN**.
networking.clusterNetwork[].cidr	A block of IP addresses from which Pod IP addresses are allocated. The **OpenShiftSDN** network plug-in supports multiple cluster networks. The address blocks for multiple cluster networks must not overlap. Select address pools large enough to fit your anticipated workload.	An IP address allocation in CIDR format. The default value is **10.128.0.0/14**.
networking.clusterNetwork[].hostPrefix	The subnet prefix length to assign to each individual node. For example, if **hostPrefix** is set to **23**, then each node is assigned a /**23** subnet out of the given **cidr**, allowing for 510 (2^(32 - 23) - 2) Pod IP addresses.	A subnet prefix. The default value is **23**.
networking.serviceNetwork[]	A block of IP addresses for services. **OpenShiftSDN** allows only one **serviceNetwork** block. The address block must not overlap with any other network block.	An IP address allocation in CIDR format. The default value is **172.30.0.0/16**.

Parameter	Description	Value
networking.machineNetwork[].cidr	A block of IP addresses assigned to nodes created by the OpenShift Container Platform installation program while installing the cluster. The address block must not overlap with any other network block. Multiple CIDR ranges may be specified.	An IP address allocation in CIDR format. The default value is **10.0.0.0/16**.

1.4.9. Modifying advanced network configuration parameters

You can modify the advanced network configuration parameters only before you install the cluster. Advanced configuration customization lets you integrate your cluster into your existing network environment by specifying an MTU or VXLAN port, by allowing customization of kube-proxy settings, and by specifying a different **mode** for the **openshiftSDNConfig** parameter.

IMPORTANT

Modifying the OpenShift Container Platform manifest files directly is not supported.

Prerequisites

- Create the **install-config.yaml** file and complete any modifications to it.
- Create the Ignition config files for your cluster.

Procedure

1. Use the following command to create manifests:

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

1 For **<installation_directory>**, specify the name of the directory that contains the **install-config.yaml** file for your cluster.

2. Create a file that is named **cluster-network-03-config.yml** in the **<installation_directory>/manifests/** directory:

```
$ touch <installation_directory>/manifests/cluster-network-03-config.yml 1
```

1 For **<installation_directory>**, specify the directory name that contains the **manifests/** directory for your cluster.

After creating the file, several network configuration files are in the **manifests/** directory, as shown:

```
$ ls <installation_directory>/manifests/cluster-network-*
```

Example output

```
cluster-network-01-crd.yml
cluster-network-02-config.yml
cluster-network-03-config.yml
```

3. Open the **cluster-network-03-config.yml** file in an editor and enter a CR that describes the Operator configuration you want:

```
apiVersion: operator.openshift.io/v1
kind: Network
metadata:
  name: cluster
spec: 1
  clusterNetwork:
  - cidr: 10.128.0.0/14
    hostPrefix: 23
  serviceNetwork:
  - 172.30.0.0/16
  defaultNetwork:
    type: OpenShiftSDN
    openshiftSDNConfig:
      mode: NetworkPolicy
      mtu: 1450
      vxlanPort: 4789
```

1 The parameters for the **spec** parameter are only an example. Specify your configuration for the Cluster Network Operator in the CR.

The CNO provides default values for the parameters in the CR, so you must specify only the parameters that you want to change.

4. Save the **cluster-network-03-config.yml** file and quit the text editor.

5. Optional: Back up the **manifests/cluster-network-03-config.yml** file. The installation program deletes the **manifests/** directory when creating the cluster.

6. Remove the Kubernetes manifest files that define the control plane machines and compute machineSets:

```
$ rm -f openshift/99_openshift-cluster-api_master-machines-*.yaml openshift/99_openshift-cluster-api_worker-machineset-*.yaml
```

Because you create and manage these resources yourself, you do not have to initialize them.

- You can preserve the MachineSet files to create compute machines by using the machine API, but you must update references to them to match your environment.

1.4.10. Cluster Network Operator configuration

The configuration for the cluster network is specified as part of the Cluster Network Operator (CNO) configuration and stored in a CR object that is named **cluster**. The CR specifies the parameters for the **Network** API in the **operator.openshift.io** API group.

You can specify the cluster network configuration for your OpenShift Container Platform cluster by setting the parameter values for the **defaultNetwork** parameter in the CNO CR. The following CR

displays the default configuration for the CNO and explains both the parameters you can configure and the valid parameter values:

Cluster Network Operator CR

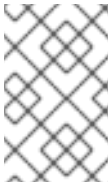
```
apiVersion: operator.openshift.io/v1
kind: Network
metadata:
  name: cluster
spec:
  clusterNetwork: 1
  - cidr: 10.128.0.0/14
    hostPrefix: 23
  serviceNetwork: 2
  - 172.30.0.0/16
  defaultNetwork: 3
    ...
  kubeProxyConfig: 4
    iptablesSyncPeriod: 30s 5
    proxyArguments:
      iptables-min-sync-period: 6
      - 0s
```

1 2 Specified in the **install-config.yaml** file.

3 Configures the default Container Network Interface (CNI) network provider for the cluster network.

4 The parameters for this object specify the **kube-proxy** configuration. If you do not specify the parameter values, the Cluster Network Operator applies the displayed default parameter values. If you are using the OVN-Kubernetes default CNI network provider, the kube-proxy configuration has no effect.

5 The refresh period for **iptables** rules. The default value is **30s**. Valid suffixes include **s**, **m**, and **h** and are described in the Go time package documentation.



NOTE

Because of performance improvements introduced in OpenShift Container Platform 4.3 and greater, adjusting the **iptablesSyncPeriod** parameter is no longer necessary.

6 The minimum duration before refreshing **iptables** rules. This parameter ensures that the refresh does not happen too frequently. Valid suffixes include **s**, **m**, and **h** and are described in the Go time package.

1.4.10.1. Configuration parameters for the OpenShift SDN default CNI network provider

The following YAML object describes the configuration parameters for the OpenShift SDN default Container Network Interface (CNI) network provider.

```
defaultNetwork:
  type: OpenShiftSDN 1
```

```
  openshiftSDNConfig: 2
    mode: NetworkPolicy 3
    mtu: 1450 4
    vxlanPort: 4789 5
```

1 Specified in the **install-config.yaml** file.

2 Specify only if you want to override part of the OpenShift SDN configuration.

3 Configures the network isolation mode for OpenShift SDN. The allowed values are **Multitenant**, **Subnet**, or **NetworkPolicy**. The default value is **NetworkPolicy**.

4 The maximum transmission unit (MTU) for the VXLAN overlay network. This value is normally configured automatically, but if the nodes in your cluster do not all use the same MTU, then you must set this explicitly to 50 less than the smallest node MTU value.

5 The port to use for all VXLAN packets. The default value is **4789**. If you are running in a virtualized environment with existing nodes that are part of another VXLAN network, then you might be required to change this. For example, when running an OpenShift SDN overlay on top of VMware NSX-T, you must select an alternate port for VXLAN, since both SDNs use the same default VXLAN port number.

On Amazon Web Services (AWS), you can select an alternate port for the VXLAN between port **9000** and port **9999**.

1.4.10.2. Cluster Network Operator example configuration

A complete CR object for the CNO is displayed in the following example:

Cluster Network Operator example CR

```
apiVersion: operator.openshift.io/v1
kind: Network
metadata:
  name: cluster
spec:
  clusterNetwork:
  - cidr: 10.128.0.0/14
    hostPrefix: 23
  serviceNetwork:
  - 172.30.0.0/16
  defaultNetwork:
    type: OpenShiftSDN
    openshiftSDNConfig:
      mode: NetworkPolicy
      mtu: 1450
      vxlanPort: 4789
  kubeProxyConfig:
    iptablesSyncPeriod: 30s
    proxyArguments:
      iptables-min-sync-period:
      - 0s
```

1.4.11. Creating the Ignition config files

Because you must manually start the cluster machines, you must generate the Ignition config files that the cluster needs to make its machines.

IMPORTANT

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

Prerequisites

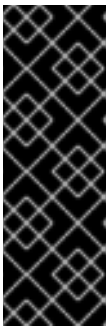
- Obtain the OpenShift Container Platform installation program and the pull secret for your cluster.

Procedure

- Obtain the Ignition config files:

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

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



IMPORTANT

If you created an **install-config.yaml** file, specify the directory that contains it. Otherwise, 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.

The following files are generated in the directory:

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

1.4.12. Creating Red Hat Enterprise Linux CoreOS (RHCOS) machines in vSphere

Before you install a cluster that contains user-provisioned infrastructure on VMware vSphere, you must create RHCOS machines on vSphere hosts for it to use.

Prerequisites

- Obtain the Ignition config files for your cluster.
- Have access to an HTTP server that you can access from your computer and that the machines that you create can access.
- Create a vSphere cluster.

Procedure

1. Upload the bootstrap Ignition config file, which is named **<installation_directory>/bootstrap.ign**, that the installation program created to your HTTP server. Note the URL of this file.
   You must host the bootstrap Ignition config file because it is too large to fit in a vApp property.

2. Save the following secondary Ignition config file for your bootstrap node to your computer as **<installation_directory>/append-bootstrap.ign**.

```
{
  "ignition": {
    "config": {
      "append": [
        {
          "source": "<bootstrap_ignition_config_url>", 1
          "verification": {}
        }
      ]
    },
    "timeouts": {},
    "version": "2.1.0"
  },
  "networkd": {},
  "passwd": {},
  "storage": {},
  "systemd": {}
}
```

   1 Specify the URL of the bootstrap Ignition config file that you hosted.

   When you create the Virtual Machine (VM) for the bootstrap machine, you use this Ignition config file.

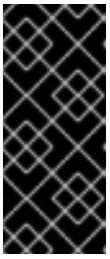
3. Convert the master, worker, and secondary bootstrap Ignition config files to Base64 encoding. For example, if you use a Linux operating system, you can use the **base64** command to encode the files.

```
$ base64 -w0 <installation_directory>/master.ign > <installation_directory>/master.64
```

```
$ base64 -w0 <installation_directory>/worker.ign > <installation_directory>/worker.64
```

```
$ base64 -w0 <installation_directory>/append-bootstrap.ign >
<installation_directory>/append-bootstrap.64
```

4. Obtain the RHCOS OVA image from the Product Downloads page on the Red Hat customer portal or the RHCOS image mirror page.



IMPORTANT

The RHCOS images might not change with every release of OpenShift Container Platform. You must download an image with the highest version that is less than or equal to the OpenShift Container Platform version that you install. Use the image version that matches your OpenShift Container Platform version if it is available.

The file name contains the OpenShift Container Platform version number in the format **rhcos-<version>-vmware.<architecture>.ova**.

5. In the vSphere Client, create a folder in your datacenter to store your VMs.

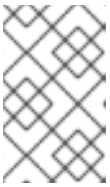
   a. Click the **VMs and Templates** view.

   b. Right-click the name of your datacenter.

   c. Click **New Folder → New VM and Template Folder**.

   d. In the window that is displayed, enter the folder name. The folder name must match the cluster name that you specified in the **install-config.yaml** file.

6. In the vSphere Client, create a template for the OVA image.



NOTE

In the following steps, you use the same template for all of your cluster machines and provide the location for the Ignition config file for that machine type when you provision the VMs.

   a. From the **Hosts and Clusters** tab, right-click your cluster's name and click **Deploy OVF Template**.

   b. On the **Select an OVF** tab, specify the name of the RHCOS OVA file that you downloaded.

   c. On the **Select a name and folder** tab, set a **Virtual machine name**, such as RHCOS, click the name of your vSphere cluster, and select the folder you created in the previous step.

   d. On the **Select a compute resource** tab, click the name of your vSphere cluster.

   e. On the **Select storage** tab, configure the storage options for your VM.

      - Select **Thin Provision** or **Thick Provision**, based on your storage preferences.
      - Select the datastore that you specified in your **install-config.yaml** file.

   f. On the **Select network** tab, specify the network that you configured for the cluster, if available.

   g. If you plan to use the same template for all cluster machine types, do not specify values on the **Customize template** tab.

7. After the template deploys, deploy a VM for a machine in the cluster.

a. Right-click the template's name and click **Clone → Clone to Virtual Machine**

b. On the **Select a name and folder** tab, specify a name for the VM. You might include the machine type in the name, such as **control-plane-0** or **compute-1**.

c. On the **Select a name and folder** tab, select the name of the folder that you created for the cluster.

d. On the **Select a compute resource** tab, select the name of a host in your datacenter.

e. Optional: On the **Select storage** tab, customize the storage options.

f. On the **Select clone options**, select **Customize this virtual machine's hardware**.

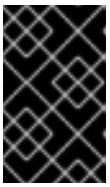
g. On the **Customize hardware** tab, click **VM Options → Advanced**.

- Optional: In the event of cluster performance issues, from the **Latency Sensitivity** list, select **High**.
- Click **Edit Configuration**, and on the **Configuration Parameters** window, click **Add Configuration Params**. Define the following parameter names and values:
  - **guestinfo.ignition.config.data**: Paste the contents of the base64-encoded Ignition config file for this machine type.
  - **guestinfo.ignition.config.data.encoding**: Specify **base64**.
  - **disk.EnableUUID**: Specify **TRUE**.
- Alternatively, prior to powering on the virtual machine add via vApp properties:
  - Navigate to a virtual machine from the vCenter Server inventory.
  - On the **Configure** tab, expand **Settings** and select **vApp options.**
  - Scroll down and under **Properties** apply the configurations from above.

h. In the **Virtual Hardware** panel of the **Customize hardware** tab, modify the specified values as required. Ensure that the amount of RAM, CPU, and disk storage meets the minimum requirements for the machine type.

i. Complete the configuration and power on the VM.

8. Create the rest of the machines for your cluster by following the preceding steps for each machine.

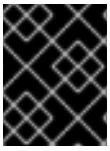


IMPORTANT

You must create the bootstrap and control plane machines at this time. Because some pods are deployed on compute machines by default, also create at least two compute machine before you install the cluster.

1.4.13. Installing the CLI by downloading the binary

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



IMPORTANT

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

1.4.13.1. Installing the CLI on Linux

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

Procedure

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

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

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

```
$ echo $PATH
```

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

```
$ oc <command>
```

1.4.13.2. Installing the CLI on Windows

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

Procedure

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

```
C:\> path
```

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

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

1.4.13.3. Installing the CLI on macOS

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

Procedure

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

   ```
   $ echo $PATH
   ```

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

```
$ oc <command>
```

1.4.14. Creating the cluster

To create the OpenShift Container Platform cluster, you wait for the bootstrap process to complete on the machines that you provisioned by using the Ignition config files that you generated with the installation program.

Prerequisites

- Create the required infrastructure for the cluster.
- You obtained the installation program and generated the Ignition config files for your cluster.
- You used the Ignition config files to create RHCOS machines for your cluster.
- Your machines have direct internet access.

Procedure

1. Monitor the bootstrap process:

   ```
   $ ./openshift-install --dir=<installation_directory> wait-for bootstrap-complete \ 1
       --log-level=info 2
   ```

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

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

Example output

```
INFO Waiting up to 30m0s for the Kubernetes API at https://api.test.example.com:6443...
INFO API v1.18.3 up
INFO Waiting up to 30m0s for bootstrapping to complete...
INFO It is now safe to remove the bootstrap resources
```

The command succeeds when the Kubernetes API server signals that it has been bootstrapped on the control plane machines.

2. After bootstrap process is complete, remove the bootstrap machine from the load balancer.

IMPORTANT

You must remove the bootstrap machine from the load balancer at this point. You can also remove or reformat the machine itself.

1.4.15. Logging in to the cluster

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

Prerequisites

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

Procedure

1. Export the **kubeadmin** credentials:

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

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

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

```
$ oc whoami
```

Example output

```
system:admin
```

1.4.16. Approving the CSRs for your machines

When you add machines to a cluster, two pending certificates signing request (CSRs) are generated for each machine that you added. You must confirm that these CSRs are approved or, if necessary, approve them yourself.

Prerequisites

- You added machines to your cluster.

Procedure

1. Confirm that the cluster recognizes the machines:

```
$ oc get nodes
```

Example output

```
NAME      STATUS    ROLES   AGE  VERSION
master-0  Ready     master  63m  v1.18.3
master-1  Ready     master  63m  v1.18.3
master-2  Ready     master  64m  v1.18.3
worker-0  NotReady  worker  76s  v1.18.3
worker-1  NotReady  worker  70s  v1.18.3
```

The output lists all of the machines that you created.

2. Review the pending certificate signing requests (CSRs) and ensure that the you see a client and server request with **Pending** or **Approved** status for each machine that you added to the cluster:

```
$ oc get csr
```

Example output

```
NAME        AGE     REQUESTOR                                                   CONDITION
csr-8b2br   15m     system:serviceaccount:openshift-machine-config-operator:node-
bootstrapper   Pending (1)
csr-8vnps   15m     system:serviceaccount:openshift-machine-config-operator:node-
bootstrapper   Pending
csr-bfd72   5m26s   system:node:ip-10-0-50-126.us-east-2.compute.internal
Pending (2)
csr-c57lv   5m26s   system:node:ip-10-0-95-157.us-east-2.compute.internal
Pending
...
```

(1) A client request CSR.

(2) A server request CSR.

In this example, two machines are joining the cluster. You might see more approved CSRs in the list.

3. If the CSRs were not approved, after all of the pending CSRs for the machines you added are in **Pending** status, approve the CSRs for your cluster machines:

NOTE

Because the CSRs rotate automatically, approve your CSRs within an hour of adding the machines to the cluster. If you do not approve them within an hour, the certificates will rotate, and more than two certificates will be present for each node. You must approve all of these certificates. After you approve the initial CSRs, the subsequent node client CSRs are automatically approved by the cluster **kube-controller-manager**. You must implement a method of automatically approving the kubelet serving certificate requests.

- To approve them individually, run the following command for each valid CSR:

```
$ oc adm certificate approve <csr_name> 1
```

1 **<csr_name>** is the name of a CSR from the list of current CSRs.

- To approve all pending CSRs, run the following command:

```
$ oc get csr -o go-template='{{range .items}}{{if not .status}}{{.metadata.name}}{{"\n"}}
{{end}}{{end}}' | xargs oc adm certificate approve
```

1.4.17. Initial Operator configuration

After the control plane initializes, you must immediately configure some Operators so that they all become available.

Prerequisites

- Your control plane has initialized.

Procedure

1. Watch the cluster components come online:

```
$ watch -n5 oc get clusteroperators
```

Example output

```
NAME                       VERSION   AVAILABLE   PROGRESSING   DEGRADED
SINCE
authentication             4.5.4     True        False         False     69s
cloud-credential           4.5.4     True        False         False     12m
cluster-autoscaler         4.5.4     True        False         False     11m
console                    4.5.4     True        False         False     46s
dns                        4.5.4     True        False         False     11m
image-registry             4.5.4     True        False         False     5m26s
ingress                    4.5.4     True        False         False     5m36s
kube-apiserver             4.5.4     True        False         False     8m53s
kube-controller-manager    4.5.4     True        False         False     7m24s
kube-scheduler             4.5.4     True        False         False     12m
machine-api                4.5.4     True        False         False     12m
machine-config             4.5.4     True        False         False     7m36s
```

```
marketplace                 4.5.4     True        False         False      7m54m
monitoring                  4.5.4     True        False         False      7h54s
network                     4.5.4     True        False         False      5m9s
node-tuning                 4.5.4     True        False         False      11m
openshift-apiserver         4.5.4     True        False         False      11m
openshift-controller-manager        4.5.4     True        False         False      5m943s
openshift-samples           4.5.4     True        False         False      3m55s
operator-lifecycle-manager          4.5.4     True        False         False      11m
operator-lifecycle-manager-catalog  4.5.4     True        False         False      11m
service-ca                  4.5.4     True        False         False      11m
service-catalog-apiserver           4.5.4     True        False         False      5m26s
service-catalog-controller-manager  4.5.4     True        False         False      5m25s
storage                     4.5.4     True        False         False      5m30s
```

2. Configure the Operators that are not available.

1.4.17.1. Image registry removed during installation

On platforms that do not provide shareable object storage, the OpenShift Image Registry Operator bootstraps itself as **Removed**. This allows **openshift-installer** to complete installations on these platform types.

After installation, you must edit the Image Registry Operator configuration to switch the **ManagementState** from **Removed** to **Managed**.

NOTE

The Prometheus console provides an **ImageRegistryRemoved** alert, for example:

"Image Registry has been removed. **ImageStreamTags**, **BuildConfigs** and **DeploymentConfigs** which reference **ImageStreamTags** may not work as expected. Please configure storage and update the config to **Managed** state by editing configs.imageregistry.operator.openshift.io."

1.4.17.2. Image registry storage configuration

The **image-registry** Operator is not initially available for platforms that do not provide default storage. After installation, you must configure your registry to use storage so that the Registry Operator is made available.

Instructions are shown for configuring a PersistentVolume, which is required for production clusters. Where applicable, instructions are shown for configuring an empty directory as the storage location, which is available for only non-production clusters.

Additional instructions are provided for allowing the image registry to use block storage types by using the **Recreate** rollout strategy during upgrades.

1.4.17.2.1. Configuring block registry storage for VMware vSphere

To allow the image registry to use block storage types such as vSphere Virtual Machine Disk (VMDK) during upgrades as a cluster administrator, you can use the **Recreate** rollout strategy.

Procedure

1. To set the image registry storage as a block storage type, patch the registry so that it uses the **Recreate** rollout strategy and runs with only **1** replica:

```
$ oc patch config.imageregistry.operator.openshift.io/cluster --type=merge -p '{"spec":
{"rolloutStrategy":"Recreate","replicas":1}}'
```

2. Provision the PV for the block storage device, and create a PVC for that volume. The requested block volume uses the ReadWriteOnce (RWO) access mode.

3. Edit the registry configuration so that it references the correct PVC.

For instructions about configuring registry storage so that it references the correct PVC, see Configuring the registry for vSphere.

1.4.18. Completing installation on user-provisioned infrastructure

After you complete the Operator configuration, you can finish installing the cluster on infrastructure that you provide.

Prerequisites

- Your control plane has initialized.
- You have completed the initial Operator configuration.

Procedure

1. If instead of installing a new cluster you are upgrading an existing cluster from version 4.3 or from versions 4.4.0 to 4.4.11, run:

```
$ oc edit configs.samples cluster
```

Then, change the line

```
managementState: Removed
```

to

```
managementState: Managed
```

2. Confirm that all the cluster components are online:

```
$ watch -n5 oc get clusteroperators
```

Example output

```
NAME                       VERSION   AVAILABLE   PROGRESSING   DEGRADED
SINCE
authentication             4.5.4     True        False         False      7m56s
cloud-credential           4.5.4     True        False         False      31m
cluster-autoscaler         4.5.4     True        False         False      16m
console                    4.5.4     True        False         False      10m
csi-snapshot-controller    4.5.4     True        False         False      16m
```

```
dns                                        4.5.4     True        False       False      22m
etcd                                       4.5.4     False       False       False      25s
image-registry                             4.5.4     True        False       False      16m
ingress                                    4.5.4     True        False       False      16m
insights                                   4.5.4     True        False       False      17m
kube-apiserver                             4.5.4     True        False       False      19m
kube-controller-manager                    4.5.4     True        False       False      20m
kube-scheduler                             4.5.4     True        False       False      20m
kube-storage-version-migrator              4.5.4     True        False       False      16m
machine-api                                4.5.4     True        False       False      22m
machine-config                             4.5.4     True        False       False      22m
marketplace                                4.5.4     True        False       False      16m
monitoring                                 4.5.4     True        False       False      10m
network                                    4.5.4     True        False       False      23m
node-tuning                                4.5.4     True        False       False      23m
openshift-apiserver                        4.5.4     True        False       False      17m
openshift-controller-manager               4.5.4     True        False       False      15m
openshift-samples                          4.5.4     True        False       False      16m
operator-lifecycle-manager                 4.5.4     True        False       False      22m
operator-lifecycle-manager-catalog         4.5.4     True        False       False      22m
operator-lifecycle-manager-packageserver   4.5.4     True        False       False      18m
service-ca                                 4.5.4     True        False       False      23m
service-catalog-apiserver                  4.5.4     True        False       False      23m
service-catalog-controller-manager         4.5.4     True        False       False      23m
storage                                    4.5.4     True        False       False      17m
```

When all of the cluster Operators are **AVAILABLE**, you can complete the installation.

3. Monitor for cluster completion:

```
$ ./openshift-install --dir=<installation_directory> wait-for install-complete 1
```

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

Example output

```
INFO Waiting up to 30m0s for the cluster to initialize...
```

The command succeeds when the Cluster Version Operator finishes deploying the OpenShift Container Platform cluster from Kubernetes API server.

IMPORTANT

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

4. Confirm that the Kubernetes API server is communicating with the Pods.

   a. To view a list of all Pods, use the following command:

```
$ oc get pods --all-namespaces
```

Example output

```
NAMESPACE                      NAME                                        READY   STATUS
RESTARTS   AGE
openshift-apiserver-operator   openshift-apiserver-operator-85cb746d55-zqhs8   1/1
Running   1          9m
openshift-apiserver            apiserver-67b9g                             1/1     Running   0
3m
openshift-apiserver            apiserver-ljcmx                             1/1     Running   0
1m
openshift-apiserver            apiserver-z25h4                             1/1     Running   0
2m
openshift-authentication-operator authentication-operator-69d5d8bf84-vh2n8        1/1
Running   0          5m
...
```

b. View the logs for a Pod that is listed in the output of the previous command by using the following command:

```
$ oc logs <pod_name> -n <namespace> 1
```

1 Specify the Pod name and namespace, as shown in the output of the previous command.

If the Pod logs display, the Kubernetes API server can communicate with the cluster machines.

1.4.19. Next steps

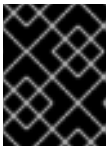
- Customize your cluster.
- If necessary, you can opt out of remote health reporting .
- Set up your registry and configure registry storage .

1.5. INSTALLING A CLUSTER ON VSPHERE IN A RESTRICTED NETWORK

In OpenShift Container Platform version 4.5, you can install a cluster on VMware vSphere infrastructure that you provision in a restricted network.

1.5.1. Prerequisites

- Create a registry on your mirror host and obtain the **imageContentSources** data for your version of OpenShift Container Platform.



IMPORTANT

Because the installation media is on the mirror host, you can use that computer to complete all installation steps.

- Provision persistent storage for your cluster. To deploy a private image registry, your storage must provide ReadWriteMany access modes.
- Review details about the OpenShift Container Platform installation and update processes.
- If you use a firewall and plan to use telemetry, you must configure the firewall to allow the sites that your cluster requires access to.

NOTE

Be sure to also review this site list if you are configuring a proxy.

1.5.2. About installations in restricted networks

In OpenShift Container Platform 4.5, you can perform an installation that does not require an active connection to the internet to obtain software components. You complete an installation in a restricted network on only infrastructure that you provision, not infrastructure that the installation program provisions, so your platform selection is limited.

If you choose to perform a restricted network installation on a cloud platform, you still require access to its cloud APIs. Some cloud functions, like Amazon Web Service's IAM service, require internet access, so you might still require internet access. Depending on your network, you might require less internet access for an installation on bare metal hardware or on VMware vSphere.

To complete a restricted network installation, you must create a registry that mirrors the contents of the OpenShift Container Platform registry and contains the installation media. You can create this registry on a mirror host, which can access both the internet and your closed network, or by using other methods that meet your restrictions.

IMPORTANT

Restricted network installations always use user-provisioned infrastructure. Because of the complexity of the configuration for user-provisioned installations, consider completing a standard user-provisioned infrastructure installation before you attempt a restricted network installation. Completing this test installation might make it easier to isolate and troubleshoot any issues that might arise during your installation in a restricted network.

1.5.2.1. Additional limits

Clusters in restricted networks have the following additional limitations and restrictions:

- The ClusterVersion status includes an **Unable to retrieve available updates** error.
- By default, you cannot use the contents of the Developer Catalog because you cannot access the required ImageStreamTags.

1.5.3. Internet and Telemetry access for OpenShift Container Platform

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

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

You must have internet access to:

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

IMPORTANT

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

1.5.4. VMware vSphere infrastructure requirements

You must install the OpenShift Container Platform cluster on a VMware vSphere version 6 instance that meets the requirements for the components that you use.

Table 1.22. Minimum supported vSphere version for VMware components

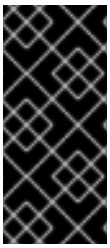
Component	Minimum supported versions	Description
Hypervisor	vSphere 6.5 with HW version 13	This version is the minimum version that Red Hat Enterprise Linux CoreOS (RHCOS) supports. See the Red Hat Enterprise Linux 8 supported hypervisors list.
Networking (NSX-T)	n/a	vSphere 6.5U3 or vSphere 6.7U2+ are required for OpenShift Container Platform. Because previous versions of vSphere with NSX-T are not currently compatible with OpenShift Container Platform, NSX-T is not supported. NSX-T certification is in process and will be supported in a future release.

Component	Minimum supported versions	Description
Storage with in-tree drivers	vSphere 6.5 or vSphere 6.7	This plug-in creates vSphere storage by using the in-tree storage drivers for vSphere included in OpenShift Container Platform and can be used when vSphere CSI drivers are not available.
Storage with vSphere CSI driver	vSphere 6.7U3 and later	This plug-in creates vSphere storage by using the standard Container Storage Interface. The vSphere CSI driver is provided and supported by VMware.

If you use a vSphere version 6.5 instance, consider upgrading to 6.7U2 before you install OpenShift Container Platform.

IMPORTANT

You must ensure that the time on your ESXi hosts is synchronized before you install OpenShift Container Platform. See Edit Time Configuration for a Host in the VMware documentation.



IMPORTANT

A limitation of using VPC is that the Storage Distributed Resource Scheduler (SDRS) is not supported. See link:https://vmware.github.io/vsphere-storage-for-kubernetes/documentation/faqs.html[vSphere Storage for Kubernetes FAQs] in the VMware documentation.

1.5.5. Machine requirements for a cluster with user-provisioned infrastructure

For a cluster that contains user-provisioned infrastructure, you must deploy all of the required machines.

1.5.5.1. Required machines

The smallest OpenShift Container Platform clusters require the following hosts:

- One temporary bootstrap machine
- Three control plane, or master, machines
- At least two compute machines, which are also known as worker machines

NOTE

The cluster requires the bootstrap machine to deploy the OpenShift Container Platform cluster on the three control plane machines. You can remove the bootstrap machine after you install the cluster.

IMPORTANT

To maintain high availability of your cluster, use separate physical hosts for these cluster machines.

The bootstrap, control plane, and compute machines must use the Red Hat Enterprise Linux CoreOS (RHCOS) as the operating system.

Note that RHCOS is based on Red Hat Enterprise Linux (RHEL) 8 and inherits all of its hardware certifications and requirements. See Red Hat Enterprise Linux technology capabilities and limits .

1.5.5.2. Network connectivity requirements

All the Red Hat Enterprise Linux CoreOS (RHCOS) machines require network in **initramfs** during boot to fetch Ignition config files from the Machine Config Server. During the initial boot, the machines require either a DHCP server or that static IP addresses be set in order to establish a network connection to download their Ignition config files.

1.5.5.3. Minimum resource requirements

Each cluster machine must meet the following minimum requirements:

Machine	Operating System	vCPU	Virtual RAM	Storage
Bootstrap	RHCOS	4	16 GB	120 GB
Control plane	RHCOS	4	16 GB	120 GB
Compute	RHCOS or RHEL 7.6	2	8 GB	120 GB

1.5.5.4. Certificate signing requests management

Because your cluster has limited access to automatic machine management when you use infrastructure that you provision, you must provide a mechanism for approving cluster certificate signing requests (CSRs) after installation. The **kube-controller-manager** only approves the kubelet client CSRs. The **machine-approver** cannot guarantee the validity of a serving certificate that is requested by using kubelet credentials because it cannot confirm that the correct machine issued the request. You must determine and implement a method of verifying the validity of the kubelet serving certificate requests and approving them.

1.5.6. Creating the user-provisioned infrastructure

Before you deploy an OpenShift Container Platform cluster that uses user-provisioned infrastructure, you must create the underlying infrastructure.

Prerequistes

- Review the OpenShift Container Platform 4.x Tested Integrations page before you create the supporting infrastructure for your cluster.

Procedure

1. Configure DHCP or set static IP addresses on each node.
2. Provision the required load balancers.
3. Configure the ports for your machines.
4. Configure DNS.
5. Ensure network connectivity.

1.5.6.1. 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, which runs on each master node after a successful cluster installation, 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.

Table 1.23. All machines to all machines

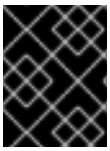
Protocol	Port	Description
ICMP	N/A	Network reachability tests
TCP	**9000**-**9999**	Host level services, including the node exporter on ports **9100**-**9101** and the Cluster Version Operator on port**9099**.
	10250-**10259**	The default ports that Kubernetes reserves
	10256	openshift-sdn
UDP	**4789**	VXLAN and GENEVE
	6081	VXLAN and GENEVE
	9000-**9999**	Host level services, including the node exporter on ports **9100**-**9101**.
TCP/UDP	**30000**-**32767**	Kubernetes NodePort

Table 1.24. All machines to control plane

Protocol	Port	Description
TCP	**2379-2380**	etcd server, peer, and metrics ports
	6443	Kubernetes API

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.
   - A stateless load balancing algorithm. The options vary based on the load balancer implementation.

NOTE

Session persistence is not required for the API load balancer to function properly.

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

Table 1.25. API load balancer

Port	Back-end machines (pool members)	Internal	External	Description
6443	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.	X	X	Kubernetes API server

Port	Back-end machines (pool members)	Internal	External	Description
22623	Bootstrap and control plane. You remove the bootstrap machine from the load balancer after the bootstrap machine initializes the cluster control plane.	X		Machine Config server

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.
   - 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 1.26. Application Ingress load balancer

Port	Back-end machines (pool members)	Internal	External	Description
443	The machines that run the Ingress router pods, compute, or worker, by default.	X	X	HTTPS traffic
80	The machines that run the Ingress router pods, compute, or worker by default.	X	X	HTTP traffic

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.

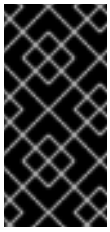
NOTE

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.

1.5.6.2. User-provisioned DNS requirements

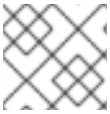
The following DNS records are required for an OpenShift Container Platform cluster that uses user-provisioned infrastructure. In each record, **<cluster_name>** is the cluster name and **<base_domain>** is the cluster base domain that you specify in the **install-config.yaml** file. A complete DNS record takes the form: **<component>.<cluster_name>.<base_domain>.**.

Table 1.27. Required DNS records

Component	Record	Description
Kubernetes API	**api.<cluster_name>.<base_domain>.**	This DNS A/AAAA or CNAME record must point to the load balancer for the control plane machines. This record must be resolvable by both clients external to the cluster and from all the nodes within the cluster.
	api-int.<cluster_name>.<base_domain>.	This DNS A/AAAA or CNAME record must point to the load balancer for the control plane machines. This record must be resolvable from all the nodes within the cluster.  **IMPORTANT** The API server must be able to resolve the worker nodes by the host names that are recorded in Kubernetes. If it cannot resolve the node names, proxied API calls can fail, and you cannot retrieve logs from Pods.
Routes	***.apps.<cluster_name>.<base_domain>.**	A wildcard DNS A/AAAA or CNAME record that points to the load balancer that targets the machines that run the Ingress router pods, which are the worker nodes by default. This record must be resolvable by both clients external to the cluster and from all the nodes within the cluster.

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

If you want to perform installation debugging or disaster recovery on your cluster, you must provide an SSH key to both your **ssh-agent** and to the installation program.



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:

```
$ ssh-keygen -t rsa -b 4096 -N '' \
    -f <path>/<file_name> 1
```

1 Specify the path and file name, such as ~/**.ssh/id_rsa**, of the SSH key. Do not specify an existing SSH key, as it will be overwritten.

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

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

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

   Example output

```
Agent pid 31874
```

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

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

   Example output

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

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

Next steps

- When you install OpenShift Container Platform, provide the SSH public key to the installation program. If you install a cluster on infrastructure that you provision, you must provide this key to your cluster's machines.

1.5.8. Manually creating the installation configuration file

For installations of OpenShift Container Platform that use user-provisioned infrastructure, you manually generate your installation configuration file.

Prerequisites

- Obtain the OpenShift Container Platform installation program and the access token for your cluster.
- Obtain the **imageContentSources** section from the output of the command to mirror the repository.

- Obtain the contents of the certificate for your mirror registry.

Procedure

1. Create an installation directory to store your required installation assets in:

```
$ mkdir <installation_directory>
```

IMPORTANT

You must create a 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.

2. Customize the following **install-config.yaml** file template and save it in the **<installation_directory>**.

NOTE

You must name this configuration file **install-config.yaml**.

- Unless you use a registry that RHCOS trusts by default, such as **docker.io**, you must provide the contents of the certificate for your mirror repository in the **additionalTrustBundle** section. In most cases, you must provide the certificate for your mirror.
- You must include the **imageContentSources** section from the output of the command to mirror the repository.

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 next step of the installation process. You must back it up now.

1.5.8.1. Sample install-config.yaml file for VMware vSphere

You can customize the **install-config.yaml** file to specify more details about your OpenShift Container Platform cluster's platform or modify the values of the required parameters.

```
apiVersion: v1
baseDomain: example.com (1)
compute:
- hyperthreading: Enabled (2) (3)
  name: worker
  replicas: 0 (4)
controlPlane:
  hyperthreading: Enabled (5) (6)
  name: master
```

```
  replicas: 3 7
metadata:
  name: test 8
platform:
  vsphere:
    vcenter: your.vcenter.server 9
    username: username 10
    password: password 11
    datacenter: datacenter 12
    defaultDatastore: datastore 13
fips: false 14
pullSecret: '{"auths":{"<local_registry>": {"auth": "<credentials>","email": "you@example.com"}}}' 15
sshKey: 'ssh-ed25519 AAAA...' 16
additionalTrustBundle: | 17
  -----BEGIN CERTIFICATE-----
  ZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZ
  -----END CERTIFICATE-----
imageContentSources: 18
- mirrors:
  - <local_registry>/<local_repository_name>/release
  source: quay.io/openshift-release-dev/ocp-release
- mirrors:
  - <local_registry>/<local_repository_name>/release
  source: registry.svc.ci.openshift.org/ocp/release
```

1 The base domain of the cluster. All DNS records must be sub-domains of this base and include the cluster name.

2 5 The **controlPlane** section is a single mapping, but the compute section is a sequence of mappings. To meet the requirements of the different data structures, the first line of the **compute** section must begin with a hyphen, **-**, and the first line of the **controlPlane** section must not. Although both sections currently define a single machine pool, it is possible that future versions of OpenShift Container Platform will support defining multiple compute pools during installation. Only one control plane pool is used.

3 6 Whether to enable or disable simultaneous multithreading, or **hyperthreading**. By default, simultaneous multithreading is enabled to increase the performance of your machines' cores. You can disable it by setting the parameter value to **Disabled**. If you disable simultaneous multithreading in some cluster machines, you must disable it in all cluster machines.

IMPORTANT

If you disable simultaneous multithreading, ensure that your capacity planning accounts for the dramatically decreased machine performance. Your machines must use at least 8 CPUs and 32 GB of RAM if you disable simultaneous multithreading.

4 You must set the value of the **replicas** parameter to **0**. This parameter controls the number of workers that the cluster creates and manages for you, which are functions that the cluster does not perform when you use user-provisioned infrastructure. You must manually deploy worker machines for the cluster to use before you finish installing OpenShift Container Platform.

7 The number of control plane machines that you add to the cluster. Because the cluster uses this values as the number of etcd endpoints in the cluster, the value must match the number of control plane machines that you deploy.

8. The cluster name that you specified in your DNS records.
9. The fully-qualified host name or IP address of the vCenter server.
10. The name of the user for accessing the server. This user must have at least the roles and privileges that are required for static or dynamic persistent volume provisioning in vSphere.
11. The password associated with the vSphere user.
12. The vSphere datacenter.
13. The default vSphere datastore to use.
14. Whether to enable or disable FIPS mode. By default, FIPS mode is not enabled. 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.
15. For **<local_registry>**, specify the registry domain name, and optionally the port, that your mirror registry uses to serve content. For example **registry.example.com** or **registry.example.com:5000**. For **<credentials>**, specify the base64-encoded user name and password for your mirror registry.
16. The public portion of the default SSH key for the **core** user in Red Hat Enterprise Linux CoreOS (RHCOS).

NOTE

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

17. Provide the contents of the certificate file that you used for your mirror registry.
18. Provide the **imageContentSources** section from the output of the command to mirror the repository.

1.5.8.2. Configuring the cluster-wide proxy during installation

Production environments can deny direct access to the Internet and instead have an HTTP or HTTPS proxy available. You can configure a new OpenShift Container Platform cluster to use a proxy by configuring the proxy settings in the **install-config.yaml** file.

Prerequisites

- An existing **install-config.yaml** file.
- Review the sites that your cluster requires access to and determine whether any need to bypass the proxy. By default, all cluster egress traffic is proxied, including calls to hosting cloud provider APIs. Add sites to the Proxy object's **spec.noProxy** field to bypass the proxy if necessary.

NOTE

The Proxy object's **status.noProxy** field is populated by default with the instance metadata endpoint (**169.254.169.254**) and with the values of the **networking.machineNetwork[].cidr**, **networking.clusterNetwork[].cidr**, and **networking.serviceNetwork[]** fields from your installation configuration.

Procedure

1. Edit your **install-config.yaml** file and add the proxy settings. For example:

```
apiVersion: v1
baseDomain: my.domain.com
proxy:
  httpProxy: http://<username>:<pswd>@<ip>:<port> 1
  httpsProxy: http://<username>:<pswd>@<ip>:<port> 2
  noProxy: example.com 3
additionalTrustBundle: | 4
    -----BEGIN CERTIFICATE-----
    <MY_TRUSTED_CA_CERT>
    -----END CERTIFICATE-----
...
```

1. A proxy URL to use for creating HTTP connections outside the cluster. The URL scheme must be **http**. If you use an MITM transparent proxy network that does not require additional proxy configuration but requires additional CAs, you must not specify an **httpProxy** value.

2. A proxy URL to use for creating HTTPS connections outside the cluster. If this field is not specified, then **httpProxy** is used for both HTTP and HTTPS connections. The URL scheme must be **http**; **https** is currently not supported. If you use an MITM transparent proxy network that does not require additional proxy configuration but requires additional CAs, you must not specify an **httpsProxy** value.

3. A comma-separated list of destination domain names, domains, IP addresses, or other network CIDRs to exclude proxying. Preface a domain with **.** to include all subdomains of that domain. Use ***** to bypass proxy for all destinations.

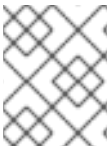
4. If provided, the installation program generates a ConfigMap that is named **user-ca-bundle** in the **openshift-config** namespace that contains one or more additional CA certificates that are required for proxying HTTPS connections. The Cluster Network Operator then creates a **trusted-ca-bundle** ConfigMap that merges these contents with the Red Hat Enterprise Linux CoreOS (RHCOS) trust bundle, and this ConfigMap is referenced in the Proxy object's **trustedCA** field. The **additionalTrustBundle** field is required unless the proxy's identity certificate is signed by an authority from the RHCOS trust bundle. If you use an MITM transparent proxy network that does not require additional proxy configuration but requires additional CAs, you must provide the MITM CA certificate.

NOTE

The installation program does not support the proxy **readinessEndpoints** field.

2. Save the file and reference it when installing OpenShift Container Platform.

The installation program creates a cluster-wide proxy that is named **cluster** that uses the proxy settings in the provided **install-config.yaml** file. If no proxy settings are provided, a **cluster** Proxy object is still created, but it will have a nil **spec**.



NOTE

Only the Proxy object named **cluster** is supported, and no additional proxies can be created.

1.5.9. Creating the Kubernetes manifest and Ignition config files

Because you must modify some cluster definition files and manually start the cluster machines, you must generate the Kubernetes manifest and Ignition config files that the cluster needs to make its machines.

IMPORTANT

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

Prerequisites

- Obtain the OpenShift Container Platform installation program. For a restricted network installation, these files are on your mirror host.
- Create the **install-config.yaml** installation configuration file.

Procedure

1. Generate the Kubernetes manifests for the cluster:

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

Example output

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

1 For **<installation_directory>**, specify the installation directory that contains the **install-config.yaml** file you created.

Because you create your own compute machines later in the installation process, you can safely ignore this warning.

2. Remove the Kubernetes manifest files that define the control plane machines and compute machineSets:

```
$ rm -f openshift/99_openshift-cluster-api_master-machines-*.yaml openshift/99_openshift-
cluster-api_worker-machineset-*.yaml
```

Because you create and manage these resources yourself, you do not have to initialize them.

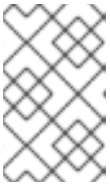
- You can preserve the MachineSet files to create compute machines by using the machine API, but you must update references to them to match your environment.

3. Modify the **<installation_directory>/manifests/cluster-scheduler-02-config.yml** Kubernetes manifest file to prevent Pods from being scheduled on the control plane machines:

   a. Open the **<installation_directory>/manifests/cluster-scheduler-02-config.yml** file.

   b. Locate the **mastersSchedulable** parameter and set its value to **False**.

   c. Save and exit the file.



NOTE

Currently, due to a Kubernetes limitation, router Pods running on control plane machines will not be reachable by the ingress load balancer. This step might not be required in a future minor version of OpenShift Container Platform.

4. Obtain the Ignition config files:

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

1 For **<installation_directory>**, specify the same installation directory.

The following files are generated in the directory:

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

1.5.10. Creating Red Hat Enterprise Linux CoreOS (RHCOS) machines in vSphere

Before you install a cluster that contains user-provisioned infrastructure on VMware vSphere, you must create RHCOS machines on vSphere hosts for it to use.

Prerequisites

- Obtain the Ignition config files for your cluster.
- Have access to an HTTP server that you can access from your computer and that the machines that you create can access.
- Create a vSphere cluster.

Procedure

1. Upload the bootstrap Ignition config file, which is named **<installation_directory>/bootstrap.ign**, that the installation program created to your HTTP server. Note the URL of this file.
   You must host the bootstrap Ignition config file because it is too large to fit in a vApp property.

2. Save the following secondary Ignition config file for your bootstrap node to your computer as **<installation_directory>/append-bootstrap.ign**.

```
{
  "ignition": {
    "config": {
      "append": [
        {
          "source": "<bootstrap_ignition_config_url>", 1
          "verification": {}
        }
      ]
    },
    "timeouts": {},
    "version": "2.1.0"
  },
  "networkd": {},
  "passwd": {},
  "storage": {},
  "systemd": {}
}
```

1 Specify the URL of the bootstrap Ignition config file that you hosted.

   When you create the Virtual Machine (VM) for the bootstrap machine, you use this Ignition config file.

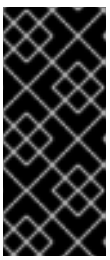
3. Convert the master, worker, and secondary bootstrap Ignition config files to Base64 encoding. For example, if you use a Linux operating system, you can use the **base64** command to encode the files.

```
$ base64 -w0 <installation_directory>/master.ign > <installation_directory>/master.64
```

```
$ base64 -w0 <installation_directory>/worker.ign > <installation_directory>/worker.64
```

```
$ base64 -w0 <installation_directory>/append-bootstrap.ign >
<installation_directory>/append-bootstrap.64
```

4. Obtain the RHCOS OVA image from the Product Downloads page on the Red Hat customer portal or the RHCOS image mirror page.



IMPORTANT

The RHCOS images might not change with every release of OpenShift Container Platform. You must download an image with the highest version that is less than or equal to the OpenShift Container Platform version that you install. Use the image version that matches your OpenShift Container Platform version if it is available.

The file name contains the OpenShift Container Platform version number in the format **rhcos-<version>-vmware.<architecture>.ova**.

5. In the vSphere Client, create a folder in your datacenter to store your VMs.

   a. Click the **VMs and Templates** view.

   b. Right-click the name of your datacenter.

   c. Click **New Folder → New VM and Template Folder**.

   d. In the window that is displayed, enter the folder name. The folder name must match the cluster name that you specified in the **install-config.yaml** file.

6. In the vSphere Client, create a template for the OVA image.

NOTE

In the following steps, you use the same template for all of your cluster machines and provide the location for the Ignition config file for that machine type when you provision the VMs.

   a. From the **Hosts and Clusters** tab, right-click your cluster's name and click **Deploy OVF Template**.

   b. On the **Select an OVF** tab, specify the name of the RHCOS OVA file that you downloaded.

   c. On the **Select a name and folder** tab, set a **Virtual machine name**, such as RHCOS, click the name of your vSphere cluster, and select the folder you created in the previous step.

   d. On the **Select a compute resource** tab, click the name of your vSphere cluster.

   e. On the **Select storage** tab, configure the storage options for your VM.

      - Select **Thin Provision** or **Thick Provision**, based on your storage preferences.
      - Select the datastore that you specified in your **install-config.yaml** file.

   f. On the **Select network** tab, specify the network that you configured for the cluster, if available.

   g. If you plan to use the same template for all cluster machine types, do not specify values on the **Customize template** tab.

7. After the template deploys, deploy a VM for a machine in the cluster.

   a. Right-click the template's name and click **Clone → Clone to Virtual Machine**

   b. On the **Select a name and folder** tab, specify a name for the VM. You might include the machine type in the name, such as **control-plane-0** or **compute-1**.

   c. On the **Select a name and folder** tab, select the name of the folder that you created for the cluster.

   d. On the **Select a compute resource** tab, select the name of a host in your datacenter.

   e. Optional: On the **Select storage** tab, customize the storage options.

f. On the **Select clone options**, select **Customize this virtual machine's hardware**.

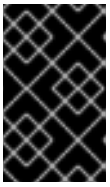
g. On the **Customize hardware** tab, click **VM Options → Advanced**.

- Optional: In the event of cluster performance issues, from the **Latency Sensitivity** list, select **High**.
- Click **Edit Configuration**, and on the **Configuration Parameters** window, click **Add Configuration Params**. Define the following parameter names and values:
  - **guestinfo.ignition.config.data**: Paste the contents of the base64-encoded Ignition config file for this machine type.
  - **guestinfo.ignition.config.data.encoding**: Specify **base64**.
  - **disk.EnableUUID**: Specify **TRUE**.
- Alternatively, prior to powering on the virtual machine add via vApp properties:
  - Navigate to a virtual machine from the vCenter Server inventory.
  - On the **Configure** tab, expand **Settings** and select **vApp options.**
  - Scroll down and under **Properties** apply the configurations from above.

h. In the **Virtual Hardware** panel of the **Customize hardware** tab, modify the specified values as required. Ensure that the amount of RAM, CPU, and disk storage meets the minimum requirements for the machine type.

i. Complete the configuration and power on the VM.

8. Create the rest of the machines for your cluster by following the preceding steps for each machine.



IMPORTANT

You must create the bootstrap and control plane machines at this time. Because some pods are deployed on compute machines by default, also create at least two compute machine before you install the cluster.

1.5.11. Creating the cluster

To create the OpenShift Container Platform cluster, you wait for the bootstrap process to complete on the machines that you provisioned by using the Ignition config files that you generated with the installation program.

Prerequisites

- Create the required infrastructure for the cluster.
- You obtained the installation program and generated the Ignition config files for your cluster.
- You used the Ignition config files to create RHCOS machines for your cluster.

Procedure

1. Monitor the bootstrap process:

```
$ ./openshift-install --dir=<installation_directory> wait-for bootstrap-complete \ 1
    --log-level=info 2
```

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

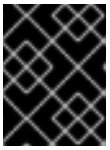
2 To view different installation details, specify **warn**, **debug**, or **error** instead of **info**.

Example output

```
INFO Waiting up to 30m0s for the Kubernetes API at https://api.test.example.com:6443...
INFO API v1.18.3 up
INFO Waiting up to 30m0s for bootstrapping to complete...
INFO It is now safe to remove the bootstrap resources
```

The command succeeds when the Kubernetes API server signals that it has been bootstrapped on the control plane machines.

2. After bootstrap process is complete, remove the bootstrap machine from the load balancer.



IMPORTANT

You must remove the bootstrap machine from the load balancer at this point. You can also remove or reformat the machine itself.

1.5.12. Logging in to the cluster

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

Prerequisites

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

Procedure

1. Export the **kubeadmin** credentials:

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

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

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

```
$ oc whoami
```

-

Example output

```
system:admin
```

1.5.13. Approving the CSRs for your machines

When you add machines to a cluster, two pending certificates signing request (CSRs) are generated for each machine that you added. You must confirm that these CSRs are approved or, if necessary, approve them yourself.

Prerequisites

- You added machines to your cluster.

Procedure

1. Confirm that the cluster recognizes the machines:

```
$ oc get nodes
```

Example output

```
NAME      STATUS    ROLES   AGE  VERSION
master-0  Ready     master  63m  v1.18.3
master-1  Ready     master  63m  v1.18.3
master-2  Ready     master  64m  v1.18.3
worker-0  NotReady  worker  76s  v1.18.3
worker-1  NotReady  worker  70s  v1.18.3
```

The output lists all of the machines that you created.

2. Review the pending certificate signing requests (CSRs) and ensure that the you see a client and server request with **Pending** or **Approved** status for each machine that you added to the cluster:

```
$ oc get csr
```

Example output

```
NAME        AGE     REQUESTOR                                                   CONDITION
csr-8b2br   15m     system:serviceaccount:openshift-machine-config-operator:node-
bootstrapper   Pending (1)
csr-8vnps   15m     system:serviceaccount:openshift-machine-config-operator:node-
bootstrapper   Pending
csr-bfd72   5m26s   system:node:ip-10-0-50-126.us-east-2.compute.internal
Pending (2)
csr-c57lv   5m26s   system:node:ip-10-0-95-157.us-east-2.compute.internal
Pending
...
```

(1) A client request CSR.

2 A server request CSR.

In this example, two machines are joining the cluster. You might see more approved CSRs in the list.

3. If the CSRs were not approved, after all of the pending CSRs for the machines you added are in **Pending** status, approve the CSRs for your cluster machines:

NOTE

Because the CSRs rotate automatically, approve your CSRs within an hour of adding the machines to the cluster. If you do not approve them within an hour, the certificates will rotate, and more than two certificates will be present for each node. You must approve all of these certificates. After you approve the initial CSRs, the subsequent node client CSRs are automatically approved by the cluster **kube-controller-manager**. You must implement a method of automatically approving the kubelet serving certificate requests.

- To approve them individually, run the following command for each valid CSR:

```
$ oc adm certificate approve <csr_name> 1
```

1 **<csr_name>** is the name of a CSR from the list of current CSRs.

- To approve all pending CSRs, run the following command:

```
$ oc get csr -o go-template='{{range .items}}{{if not .status}}{{.metadata.name}}{{"\n"}}
{{end}}{{end}}' | xargs oc adm certificate approve
```

1.5.14. Initial Operator configuration

After the control plane initializes, you must immediately configure some Operators so that they all become available.

Prerequisites

- Your control plane has initialized.

Procedure

1. Watch the cluster components come online:

```
$ watch -n5 oc get clusteroperators
```

Example output

```
NAME                                       VERSION   AVAILABLE   PROGRESSING   DEGRADED
SINCE
authentication                             4.5.4     True        False         False     69s
cloud-credential                           4.5.4     True        False         False     12m
cluster-autoscaler                         4.5.4     True        False         False     11m
```

```
console                                    4.5.4     True        False         False      46s
dns                                        4.5.4     True        False         False      11m
image-registry                             4.5.4     True        False         False      5m26s
ingress                                    4.5.4     True        False         False      5m36s
kube-apiserver                             4.5.4     True        False         False      8m53s
kube-controller-manager                    4.5.4     True        False         False      7m24s
kube-scheduler                             4.5.4     True        False         False      12m
machine-api                                4.5.4     True        False         False      12m
machine-config                             4.5.4     True        False         False      7m36s
marketplace                                4.5.4     True        False         False      7m54m
monitoring                                 4.5.4     True        False         False      7h54s
network                                    4.5.4     True        False         False      5m9s
node-tuning                                4.5.4     True        False         False      11m
openshift-apiserver                        4.5.4     True        False         False      11m
openshift-controller-manager               4.5.4     True        False         False      5m943s
openshift-samples                          4.5.4     True        False         False      3m55s
operator-lifecycle-manager                 4.5.4     True        False         False      11m
operator-lifecycle-manager-catalog         4.5.4     True        False         False      11m
service-ca                                 4.5.4     True        False         False      11m
service-catalog-apiserver                  4.5.4     True        False         False      5m26s
service-catalog-controller-manager         4.5.4     True        False         False      5m25s
storage                                    4.5.4     True        False         False      5m30s
```

2. Configure the Operators that are not available.

1.5.14.1. Image registry storage configuration

The **image-registry** Operator is not initially available for platforms that do not provide default storage. After installation, you must configure your registry to use storage so that the Registry Operator is made available.

Instructions are shown for configuring a PersistentVolume, which is required for production clusters. Where applicable, instructions are shown for configuring an empty directory as the storage location, which is available for only non-production clusters.

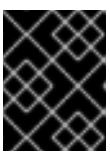
Additional instructions are provided for allowing the image registry to use block storage types by using the **Recreate** rollout strategy during upgrades.

1.5.14.1.1. Configuring registry storage for VMware vSphere

As a cluster administrator, following installation you must configure your registry to use storage.

Prerequisites

- Cluster administrator permissions.
- A cluster on VMware vSphere.
- Provision persistent storage for your cluster. To deploy a private image registry, your storage must provide ReadWriteMany access mode.



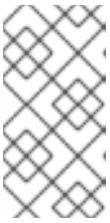
IMPORTANT

vSphere volumes do not support the **ReadWriteMany** access mode. You must use a different storage backend, such as **NFS**, to configure the registry storage.

- Must have "100Gi" capacity.

Procedure

1. To configure your registry to use storage, change the **spec.storage.pvc** in the **configs.imageregistry/cluster** resource.

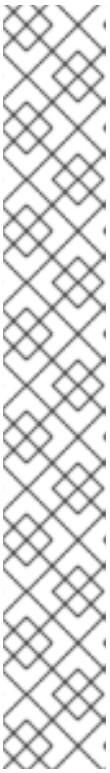


NOTE

When using shared storage such as NFS, it is strongly recommended to use the **supplementalGroups** strategy, which dictates the allowable supplemental groups for the Security Context, rather than the **fsGroup** ID. Refer to the NFS **Group IDs** documentation for details.

2. Verify you do not have a registry Pod:

```
$ oc get pod -n openshift-image-registry
```



NOTE

- If the storage type is **emptyDIR**, the replica number cannot be greater than **1**.
- If the storage type is **NFS**, you must enable the **no_wdelay** and **root_squash** mount options. For example:

```
# cat /etc/exports
```

Example output

```
/mnt/data *(rw,sync,no_wdelay,root_squash,insecure,fsid=0)
```

```
sh-4.2# exportfs -rv
```

Example output

```
exporting *:/mnt/data
```

3. Check the registry configuration:

```
$ oc edit configs.imageregistry.operator.openshift.io
```

Example registry configuration

```
storage:
  pvc:
    claim:
```

Leave the **claim** field blank to allow the automatic creation of an **image-registry-storage** PVC.

4. Optional: Add a new storage class to a PV:

   a. Create the PV:

```
$ oc create -f -
```

```
apiVersion: v1
kind: PersistentVolume
metadata:
  name: image-registry-pv
spec:
  accessModes:
    - ReadWriteMany
  capacity:
      storage: 100Gi
  nfs:
    path: /registry
    server: 172.16.231.181
  persistentVolumeReclaimPolicy: Retain
  storageClassName: nfs01
```

```
$ oc get pv
```

   b. Create the PVC:

```
$ oc create -n openshift-image-registry -f -
```

```
apiVersion: "v1"
kind: "PersistentVolumeClaim"
metadata:
  name: "image-registry-pvc"
spec:
  accessModes:
    - ReadWriteMany
  resources:
    requests:
      storage: 100Gi
  storageClassName: nfs01
  volumeMode: Filesystem
```

```
$ oc get pvc -n openshift-image-registry
```

   Finally, add the name of your PVC:

```
$ oc edit configs.imageregistry.operator.openshift.io -o yaml
```

```
storage:
  pvc:
    claim: image-registry-pvc (1)
```

   (1) Creating a custom PVC allows you to leave the **claim** field blank for default automatic creation of an **image-registry-storage** PVC.

5. Check the **clusteroperator** status:

```
$ oc get clusteroperator image-registry
```

1.5.14.1.2. Configuring storage for the image registry in non-production clusters

You must configure storage for the image registry Operator. For non-production clusters, you can set the image registry to an empty directory. If you do so, all images are lost if you restart the registry.

Procedure

1. To set the image registry storage to an empty directory:

```
$ oc patch configs.imageregistry.operator.openshift.io cluster --type merge --patch '{"spec":{"storage":{"emptyDir":{}}}}'
```

> **WARNING**
>
> Configure this option for only non-production clusters.

If you run this command before the Image Registry Operator initializes its components, the **oc patch** command fails with the following error:

```
Error from server (NotFound): configs.imageregistry.operator.openshift.io "cluster" not found
```

Wait a few minutes and run the command again.

2. Ensure that your registry is set to managed to enable building and pushing of images.

- Run:

```
$ oc edit configs.imageregistry/cluster
```

Then, change the line

```
managementState: Removed
```

to

```
managementState: Managed
```

1.5.14.1.3. Configuring block registry storage for VMware vSphere

To allow the image registry to use block storage types such as vSphere Virtual Machine Disk (VMDK) during upgrades as a cluster administrator, you can use the **Recreate** rollout strategy.

Procedure

1. To set the image registry storage as a block storage type, patch the registry so that it uses the **Recreate** rollout strategy and runs with only **1** replica:

```
$ oc patch config.imageregistry.operator.openshift.io/cluster --type=merge -p '{"spec":
{"rolloutStrategy":"Recreate","replicas":1}}'
```

2. Provision the PV for the block storage device, and create a PVC for that volume. The requested block volume uses the ReadWriteOnce (RWO) access mode.

3. Edit the registry configuration so that it references the correct PVC.

For instructions about configuring registry storage so that it references the correct PVC, see Configuring the registry for vSphere.

1.5.15. Completing installation on user-provisioned infrastructure

After you complete the Operator configuration, you can finish installing the cluster on infrastructure that you provide.

Prerequisites

- Your control plane has initialized.
- You have completed the initial Operator configuration.

Procedure

1. If instead of installing a new cluster you are upgrading an existing cluster from version 4.3 or from versions 4.4.0 to 4.4.11, run:

```
$ oc edit configs.samples cluster
```

Then, change the line

```
managementState: Removed
```

to

```
managementState: Managed
```

2. Confirm that all the cluster components are online:

```
$ watch -n5 oc get clusteroperators
```

Example output

```
NAME                                       VERSION   AVAILABLE   PROGRESSING   DEGRADED
SINCE
authentication                             4.5.4     True        False         False      7m56s
cloud-credential                           4.5.4     True        False         False      31m
cluster-autoscaler                         4.5.4     True        False         False      16m
console                                    4.5.4     True        False         False      10m
csi-snapshot-controller                    4.5.4     True        False         False      16m
```

```
dns                                        4.5.4     True        False       False      22m
etcd                                       4.5.4     False       False       False      25s
image-registry                             4.5.4     True        False       False      16m
ingress                                    4.5.4     True        False       False      16m
insights                                   4.5.4     True        False       False      17m
kube-apiserver                             4.5.4     True        False       False      19m
kube-controller-manager                    4.5.4     True        False       False      20m
kube-scheduler                             4.5.4     True        False       False      20m
kube-storage-version-migrator              4.5.4     True        False       False      16m
machine-api                                4.5.4     True        False       False      22m
machine-config                             4.5.4     True        False       False      22m
marketplace                                4.5.4     True        False       False      16m
monitoring                                 4.5.4     True        False       False      10m
network                                    4.5.4     True        False       False      23m
node-tuning                                4.5.4     True        False       False      23m
openshift-apiserver                        4.5.4     True        False       False      17m
openshift-controller-manager               4.5.4     True        False       False      15m
openshift-samples                          4.5.4     True        False       False      16m
operator-lifecycle-manager                 4.5.4     True        False       False      22m
operator-lifecycle-manager-catalog         4.5.4     True        False       False      22m
operator-lifecycle-manager-packageserver   4.5.4     True        False       False      18m
service-ca                                 4.5.4     True        False       False      23m
service-catalog-apiserver                  4.5.4     True        False       False      23m
service-catalog-controller-manager         4.5.4     True        False       False      23m
storage                                    4.5.4     True        False       False      17m
```

When all of the cluster Operators are **AVAILABLE**, you can complete the installation.

3. Monitor for cluster completion:

```
$ ./openshift-install --dir=<installation_directory> wait-for install-complete 1
```

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

Example output

```
INFO Waiting up to 30m0s for the cluster to initialize...
```

The command succeeds when the Cluster Version Operator finishes deploying the OpenShift Container Platform cluster from Kubernetes API server.

IMPORTANT

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

4. Confirm that the Kubernetes API server is communicating with the Pods.

   a. To view a list of all Pods, use the following command:

```
$ oc get pods --all-namespaces
```

Example output

```
NAMESPACE                      NAME                                        READY   STATUS
RESTARTS   AGE
openshift-apiserver-operator   openshift-apiserver-operator-85cb746d55-zqhs8   1/1
Running   1          9m
openshift-apiserver            apiserver-67b9g                             1/1     Running   0
3m
openshift-apiserver            apiserver-ljcmx                             1/1     Running   0
1m
openshift-apiserver            apiserver-z25h4                             1/1     Running   0
2m
openshift-authentication-operator authentication-operator-69d5d8bf84-vh2n8        1/1
Running   0          5m
...
```

b. View the logs for a Pod that is listed in the output of the previous command by using the following command:

```
$ oc logs <pod_name> -n <namespace> 1
```

1 Specify the Pod name and namespace, as shown in the output of the previous command.

If the Pod logs display, the Kubernetes API server can communicate with the cluster machines.

5. Register your cluster on the Cluster registration page.

1.5.16. Next steps

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

1.6. UNINSTALLING A CLUSTER ON VSPHERE THAT USES INSTALLER-PROVISIONED INFRASTRUCTURE

You can remove a cluster that you deployed in your VMware vSphere instance by using installer-provisioned infrastructure.

1.6.1. Removing a cluster that uses installer-provisioned infrastructure

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

Prerequisites

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

Procedure

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

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

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

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

NOTE

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

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