



OpenShift Container Platform 4.13

Installing on a single node

Installing OpenShift Container Platform on a single node

OpenShift Container Platform 4.13 Installing on a single node

Installing OpenShift Container Platform on a single node

Legal Notice

Copyright © 2024 Red Hat, Inc.

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

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

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

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

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

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

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

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

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

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

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

All other trademarks are the property of their respective owners.

Abstract

This document describes how to install OpenShift Container Platform on a single node.

Table of Contents

CHAPTER 1. PREPARING TO INSTALL ON A SINGLE NODE	3
1.1. PREREQUISITES	3
1.2. ABOUT OPENSIFT ON A SINGLE NODE	3
1.3. REQUIREMENTS FOR INSTALLING OPENSIFT ON A SINGLE NODE	3
CHAPTER 2. INSTALLING OPENSIFT ON A SINGLE NODE	5
2.1. INSTALLING SINGLE-NODE OPENSIFT USING THE ASSISTED INSTALLER	5
2.1.1. Generating the discovery ISO with the Assisted Installer	5
2.1.2. Installing single-node OpenShift with the Assisted Installer	6
2.2. INSTALLING SINGLE-NODE OPENSIFT MANUALLY	6
2.2.1. Generating the installation ISO with coreos-installer	7
2.2.2. Monitoring the cluster installation using openshift-install	9
2.3. INSTALLING SINGLE-NODE OPENSIFT ON AWS	10
2.3.1. Additional requirements for installing on a single node on AWS	10
2.3.2. Installing single-node OpenShift on AWS	10
2.4. CREATING A BOOTABLE ISO IMAGE ON A USB DRIVE	11
2.5. BOOTING FROM AN HTTP-HOSTED ISO IMAGE USING THE REDFISH API	11
2.6. CREATING A CUSTOM LIVE RHCOS ISO FOR REMOTE SERVER ACCESS	12

CHAPTER 1. PREPARING TO INSTALL ON A SINGLE NODE

1.1. PREREQUISITES

- You reviewed details about the [OpenShift Container Platform installation and update](#) processes.
- You have read the documentation on [selecting a cluster installation method and preparing it for users](#).

1.2. ABOUT OPENSIFT ON A SINGLE NODE

You can create a single-node cluster with standard installation methods. OpenShift Container Platform on a single node is a specialized installation that requires the creation of a special ignition configuration ISO. The primary use case is for edge computing workloads, including intermittent connectivity, portable clouds, and 5G radio access networks (RAN) close to a base station. The major tradeoff with an installation on a single node is the lack of high availability.



IMPORTANT

The use of OpenShiftSDN with single-node OpenShift is not supported. OVN-Kubernetes is the default network plugin for single-node OpenShift deployments.

1.3. REQUIREMENTS FOR INSTALLING OPENSIFT ON A SINGLE NODE

Installing OpenShift Container Platform on a single node alleviates some of the requirements for high availability and large scale clusters. However, you must address the following requirements:

- **Administration host:** You must have a computer to prepare the ISO, to create the USB boot drive, and to monitor the installation.
- **CPU Architecture:** Installing OpenShift Container Platform on a single node supports **x86_64** and **arm64** CPU architectures.
- **Supported platforms:** Installing OpenShift Container Platform on a single node is supported on bare metal and [Certified third-party hypervisors](#). In most cases, you must specify the **platform:none: {}** parameter in the **install-config.yaml** configuration file. The following list shows the only exceptions and the corresponding parameter to specify in the **install-config.yaml** configuration file:
 - Amazon Web Services (AWS), where you use **platform=aws**
 - Google Cloud Platform (GCP), where you use **platform=gcp**
 - Microsoft Azure, where you use **platform=azure**
- **Production-grade server:** Installing OpenShift Container Platform on a single node requires a server with sufficient resources to run OpenShift Container Platform services and a production workload.

Table 1.1. Minimum resource requirements

Profile	vCPU	Memory	Storage
Minimum	8 vCPUs	16GB of RAM	120GB

**NOTE**

One vCPU equals one physical core. However, if you enable simultaneous multithreading (SMT), or Hyper-Threading, use the following formula to calculate the number of vCPUs that represent one physical core:

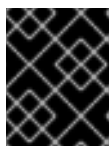
- $(\text{threads per core} \times \text{cores}) \times \text{sockets} = \text{vCPUs}$
- Adding Operators during the installation process might increase the minimum resource requirements.

The server must have a Baseboard Management Controller (BMC) when booting with virtual media.

- **Networking:** The server must have access to the internet or access to a local registry if it is not connected to a routable network. The server must have a DHCP reservation or a static IP address for the Kubernetes API, ingress route, and cluster node domain names. You must configure the DNS to resolve the IP address to each of the following fully qualified domain names (FQDN):

Table 1.2. Required DNS records

Usage	FQDN	Description
Kubernetes API	api.<cluster_name>.<base_domain>	Add a DNS A/AAAA or CNAME record. This record must be resolvable by both clients external to the cluster and within the cluster.
Internal API	api-int.<cluster_name>.<base_domain>	Add a DNS A/AAAA or CNAME record when creating the ISO manually. This record must be resolvable by nodes within the cluster.
Ingress route	*.apps.<cluster_name>.<base_domain>	Add a wildcard DNS A/AAAA or CNAME record that targets the node. This record must be resolvable by both clients external to the cluster and within the cluster.

**IMPORTANT**

Without persistent IP addresses, communications between the **apiserver** and **etcd** might fail.

CHAPTER 2. INSTALLING OPENSIFT ON A SINGLE NODE

You can install single-node OpenShift by using either the web-based Assisted Installer or the **coreos-installer** tool to generate a discovery ISO image. The discovery ISO image writes the Red Hat Enterprise Linux CoreOS (RHCOS) system configuration to the target installation disk, so that you can run a single-cluster node to meet your needs.

Consider using single-node OpenShift when you want to run a cluster in a low-resource or an isolated environment for testing, troubleshooting, training, or small-scale project purposes.

2.1. INSTALLING SINGLE-NODE OPENSIFT USING THE ASSISTED INSTALLER

To install OpenShift Container Platform on a single node, use the web-based Assisted Installer wizard to guide you through the process and manage the installation.

2.1.1. Generating the discovery ISO with the Assisted Installer

Installing OpenShift Container Platform on a single node requires a discovery ISO, which the Assisted Installer can generate.

Procedure

1. On the administration host, open a browser and navigate to [Red Hat OpenShift Cluster Manager](#).
2. Click **Create New Cluster** to create a new cluster.
3. In the **Cluster name** field, enter a name for the cluster.
4. In the **Base domain** field, enter a base domain. For example:

```
example.com
```

All DNS records must be subdomains of this base domain and include the cluster name, for example:

```
<cluster_name>.example.com
```



NOTE

You cannot change the base domain or cluster name after cluster installation.

5. Select **Install single node OpenShift (SNO)** and complete the rest of the wizard steps. Download the discovery ISO.
6. Complete the remaining Assisted Installer wizard steps.



IMPORTANT

Ensure that you take note of the discovery ISO URL for installing with virtual media.

If you enable OpenShift Virtualization during this process, you must have a second local storage device of at least 50GiB for your virtual machines.

Additional resources

- [Persistent storage using logical volume manager storage](#)
- [What you can do with OpenShift Virtualization](#)

2.1.2. Installing single-node OpenShift with the Assisted Installer

Use the Assisted Installer to install the single-node cluster.

Prerequisites

- Ensure that the boot drive order in the server BIOS settings defaults to booting the server from the target installation disk.

Procedure

1. Attach the discovery ISO image to the target host.
2. Boot the server from the discovery ISO image. The discovery ISO image writes the system configuration to the target installation disk and automatically triggers a server restart.
3. On the administration host, return to the browser. Wait for the host to appear in the list of discovered hosts. If necessary, reload the [Assisted Clusters](#) page and select the cluster name.
4. Complete the install wizard steps. Add networking details, including a subnet from the available subnets. Add the SSH public key if necessary.
5. Monitor the installation's progress. Watch the cluster events. After the installation process finishes writing the operating system image to the server's hard disk, the server restarts.
6. Optional: Remove the discovery ISO image.
The server restarts several times automatically, deploying the control plane.

Additional resources

- [Creating a bootable ISO image on a USB drive](#)
- [Booting from an HTTP-hosted ISO image using the Redfish API](#)
- [Adding worker nodes to single-node OpenShift clusters](#)

2.2. INSTALLING SINGLE-NODE OPENSIFT MANUALLY

To install OpenShift Container Platform on a single node, first generate the installation ISO, and then boot the server from the ISO. You can monitor the installation using the **openshift-install** installation program.

2.2.1. Generating the installation ISO with coreos-installer

Installing OpenShift Container Platform on a single node requires an installation ISO, which you can generate with the following procedure.

Prerequisites

- Install **podman**.



NOTE

See "Requirements for installing OpenShift on a single node" for networking requirements, including DNS records.

Procedure

1. Set the OpenShift Container Platform version:

```
$ export OCP_VERSION=<ocp_version> 1
```

- 1 Replace **<ocp_version>** with the current version, for example, **latest-4.13**

2. Set the host architecture:

```
$ export ARCH=<architecture> 1
```

- 1 Replace **<architecture>** with the target host architecture, for example, **aarch64** or **x86_64**.

3. Download the OpenShift Container Platform client (**oc**) and make it available for use by entering the following commands:

```
$ curl -k https://mirror.openshift.com/pub/openshift-
v4/clients/ocp/$OCP_VERSION/openshift-client-linux.tar.gz -o oc.tar.gz
```

```
$ tar zxf oc.tar.gz
```

```
$ chmod +x oc
```

4. Download the OpenShift Container Platform installer and make it available for use by entering the following commands:

```
$ curl -k https://mirror.openshift.com/pub/openshift-
v4/clients/ocp/$OCP_VERSION/openshift-install-linux.tar.gz -o openshift-install-linux.tar.gz
```

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

```
$ chmod +x openshift-install
```

5. Retrieve the RHCOS ISO URL by running the following command:

```
$ export ISO_URL=$(./openshift-install coreos print-stream-json | grep location | grep $ARCH | grep iso | cut -d" " -f4)
```

6. Download the RHCOS ISO:

```
$ curl -L $ISO_URL -o rhcos-live.iso
```

7. Prepare the **install-config.yaml** file:

```
apiVersion: v1
baseDomain: <domain> ❶
compute:
- name: worker
  replicas: 0 ❷
controlPlane:
  name: master
  replicas: 1 ❸
metadata:
  name: <name> ❹
networking: ❺
  clusterNetwork:
  - cidr: 10.128.0.0/14
    hostPrefix: 23
  machineNetwork:
  - cidr: 10.0.0.0/16 ❻
  networkType: OVNKubernetes
  serviceNetwork:
  - 172.30.0.0/16
platform:
  none: {}
bootstrapInPlace:
  installationDisk: /dev/disk/by-id/<disk_id> ❼
pullSecret: '<pull_secret>' ❽
sshKey: |
  <ssh_key> ❾
```

- ❶ Add the cluster domain name.
- ❷ Set the **compute** replicas to **0**. This makes the control plane node schedulable.
- ❸ Set the **controlPlane** replicas to **1**. In conjunction with the previous **compute** setting, this setting ensures the cluster runs on a single node.
- ❹ Set the **metadata** name to the cluster name.
- ❺ Set the **networking** details. OVN-Kubernetes is the only allowed network plugin type for single-node clusters.
- ❻ Set the **cidr** value to match the subnet of the single-node OpenShift cluster.
- ❼ Set the path to the installation disk drive, for example, **/dev/disk/by-id/wwn-0x64cd98f04fde100024684cf3034da5c2**.
- ❽ Copy the [pull secret from the Red Hat OpenShift Cluster Manager](#) and add the contents to this configuration setting.
- ❾

this configuration setting.

9. Add the public SSH key from the administration host so that you can log in to the cluster after installation.

8. Generate OpenShift Container Platform assets by running the following commands:

```
$ mkdir ocp
```

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

```
$ ./openshift-install --dir=ocp create single-node-ignition-config
```

9. Embed the ignition data into the RHCOS ISO by running the following commands:

```
$ alias coreos-installer='podman run --privileged --pull always --rm \
-v /dev:/dev -v /run/udev:/run/udev -v $PWD:/data \
-w /data quay.io/coreos/coreos-installer:release'
```

```
$ coreos-installer iso ignition embed -fi ocp/bootstrap-in-place-for-live-iso.ign rhcos-live.iso
```

Additional resources

- See [Requirements for installing OpenShift on a single node](#) for more information about installing OpenShift Container Platform on a single node.
- See [Enabling cluster capabilities](#) for more information about enabling cluster capabilities that were disabled prior to installation.
- See [Optional cluster capabilities in OpenShift Container Platform 4.13](#) for more information about the features provided by each capability.

2.2.2. Monitoring the cluster installation using openshift-install

Use **openshift-install** to monitor the progress of the single-node cluster installation.

Prerequisites

- Ensure that the boot drive order in the server BIOS settings defaults to booting the server from the target installation disk.

Procedure

1. Attach the discovery ISO image to the target host.
2. Boot the server from the discovery ISO image. The discovery ISO image writes the system configuration to the target installation disk and automatically triggers a server restart.
3. On the administration host, monitor the installation by running the following command:

```
$ ./openshift-install --dir=ocp wait-for install-complete
```

The server restarts several times while deploying the control plane.

Verification

- After the installation is complete, check the environment by running the following command:

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

```
$ oc get nodes
```

Example output

```
NAME                STATUS ROLES    AGE  VERSION
control-plane.example.com Ready  master,worker 10m  v1.26.0
```

Additional resources

- [Creating a bootable ISO image on a USB drive](#)
- [Booting from an HTTP-hosted ISO image using the Redfish API](#)
- [Adding worker nodes to single-node OpenShift clusters](#)

2.3. INSTALLING SINGLE-NODE OPENSIFT ON AWS

2.3.1. Additional requirements for installing on a single node on AWS

The AWS documentation for installer-provisioned installation is written with a high availability cluster consisting of three control plane nodes. When referring to the AWS documentation, consider the differences between the requirements for a single-node OpenShift cluster and a high availability cluster.

- The required machines for cluster installation in AWS documentation indicates a temporary bootstrap machine, three control plane machines, and at least two compute machines. You require only a temporary bootstrap machine and one AWS instance for the control plane node and no worker nodes.
- The minimum resource requirements for cluster installation in the AWS documentation indicates a control plane node with 4 vCPUs and 100GB of storage. For a single node cluster, you must have a minimum of 8 vCPU cores and 120GB of storage.
- The **controlPlane.replicas** setting in the **install-config.yaml** file should be set to **1**.
- The **compute.replicas** setting in the **install-config.yaml** file should be set to **0**. This makes the control plane node schedulable.

2.3.2. Installing single-node OpenShift on AWS

Installing a single node cluster on AWS requires installer-provisioned installation using the "Installing a cluster on AWS with customizations" procedure.

Additional resources

- [Installing a cluster on AWS with customizations](#)

2.4. CREATING A BOOTABLE ISO IMAGE ON A USB DRIVE

You can install software using a bootable USB drive that contains an ISO image. Booting the server with the USB drive prepares the server for the software installation.

Procedure

1. On the administration host, insert a USB drive into a USB port.
2. Create a bootable USB drive, for example:

```
# dd if=<path_to_iso> of=<path_to_usb> status=progress
```

where:

<path_to_iso>

is the relative path to the downloaded ISO file, for example, **rhcos-live.iso**.

<path_to_usb>

is the location of the connected USB drive, for example, **/dev/sdb**.

After the ISO is copied to the USB drive, you can use the USB drive to install software on the server.

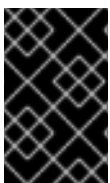
2.5. BOOTING FROM AN HTTP-HOSTED ISO IMAGE USING THE REDFISH API

You can provision hosts in your network using ISOs that you install using the Redfish Baseboard Management Controller (BMC) API.



NOTE

This example procedure demonstrates the steps on a Dell server.



IMPORTANT

Ensure that you have the latest firmware version of iDRAC that is compatible with your hardware. If you have any issues with the hardware or firmware, you must contact the provider.

Prerequisites

- Download the installation Red Hat Enterprise Linux CoreOS (RHCOS) ISO.
- Use a Dell PowerEdge server that is compatible with iDRAC9.

Procedure

1. Copy the ISO file to an HTTP server accessible in your network.
2. Boot the host from the hosted ISO file, for example:
 - a. Call the Redfish API to set the hosted ISO as the **VirtualMedia** boot media by running the following command:

```
$ curl -k -u <bmc_username>:<bmc_password> -d '{"Image": "<hosted_iso_file>",
"Inserted": true}' -H "Content-Type: application/json" -X POST
<host_bmc_address>/redfish/v1/Managers/iDRAC.Embedded.1/VirtualMedia/CD/Actions/Vi
rtualMedia.InsertMedia
```

Where:

<bmc_username>:<bmc_password>

Is the username and password for the target host BMC.

<hosted_iso_file>

Is the URL for the hosted installation ISO, for example:

<http://webserver.example.com/rhcos-live-minimal.iso>. The ISO must be accessible from the target host machine.

<host_bmc_address>

Is the BMC IP address of the target host machine.

- b. Set the host to boot from the **VirtualMedia** device by running the following command:

```
$ curl -k -u <bmc_username>:<bmc_password> -X PATCH -H 'Content-Type:
application/json' -d '{"Boot": {"BootSourceOverrideTarget": "Cd",
"BootSourceOverrideMode": "UEFI", "BootSourceOverrideEnabled": "Once"}}'
<host_bmc_address>/redfish/v1/Systems/System.Embedded.1
```

- c. Reboot the host:

```
$ curl -k -u <bmc_username>:<bmc_password> -d '{"ResetType": "ForceRestart"}' -H
'Content-type: application/json' -X POST
<host_bmc_address>/redfish/v1/Systems/System.Embedded.1/Actions/ComputerSystem.R
eset
```

- d. Optional: If the host is powered off, you can boot it using the **{"ResetType": "On"}** switch. Run the following command:

```
$ curl -k -u <bmc_username>:<bmc_password> -d '{"ResetType": "On"}' -H 'Content-
type: application/json' -X POST
<host_bmc_address>/redfish/v1/Systems/System.Embedded.1/Actions/ComputerSystem.R
eset
```

2.6. CREATING A CUSTOM LIVE RHCOS ISO FOR REMOTE SERVER ACCESS

In some cases, you cannot attach an external disk drive to a server, however, you need to access the server remotely to provision a node. It is recommended to enable SSH access to the server. You can create a live RHCOS ISO with SSHd enabled and with predefined credentials so that you can access the server after it boots.

Prerequisites

- You installed the **butane** utility.

Procedure

1. Download the **coreos-installer** binary from the **coreos-installer** image [mirror](#) page.
2. Download the latest live RHCOS ISO from mirror.openshift.com.
3. Create the **embedded.yaml** file that the **butane** utility uses to create the Ignition file:

```
variant: openshift
version: 4.13.0
metadata:
  name: sshd
  labels:
    machineconfiguration.openshift.io/role: worker
passwd:
users:
  - name: core 1
  ssh_authorized_keys:
    - '<ssh_key>'
```

1 The **core** user has sudo privileges.

4. Run the **butane** utility to create the Ignition file using the following command:

```
$ butane -pr embedded.yaml -o embedded.ign
```

5. After the Ignition file is created, you can include the configuration in a new live RHCOS ISO, which is named **rhcos-sshd-4.13.0-x86_64-live.x86_64.iso**, with the **coreos-installer** utility:

```
$ coreos-installer iso ignition embed -i embedded.ign rhcos-4.13.0-x86_64-live.x86_64.iso -o rhcos-sshd-4.13.0-x86_64-live.x86_64.iso
```

Verification

- Check that the custom live ISO can be used to boot the server by running the following command:

```
# coreos-installer iso ignition show rhcos-sshd-4.13.0-x86_64-live.x86_64.iso
```

Example output

```
{
  "ignition": {
    "version": "3.2.0"
  },
  "passwd": {
    "users": [
      {
        "name": "core",
        "sshAuthorizedKeys": [
          "ssh-rsa
AAAAAB3NzaC1yc2EAAAADAQABAAQACzNzG8AlzIDAhpyENpK2qKiTT8EbRWOrz7NXjRzo
pbPu215mocaJgjjwJjh1cYhgPhpAp6M/ttTk7I4OI7g4588ApX4bwJep6oWTU35LkY8ZxkGVPAJL
8kVITdKQviDv3XX12I4QfnDom4tm4gVbRH0gNT1wzhnLP+LKym2Ohr9D7p9NBnAdro6k++X
WgkDeijLRUTwdEyWunldW1f8G0Mg8Y1Xzr13BUo3+8aey7HLKJMDtobkz/C8ESYA/f7HJc5Fx
```

```
F0XbapWWovSSDJrr9OmIL9f4TfE+cQk3s+eoKiz2bgNPRgEEwihVbGsCN4grA+RzLCAOpec+
2dTJrQvFqsD alosadag@sonnelicht.local"
```

```
    ]
  }
]
}
}
```