

# Cloudera on premises / CDP Private Cloud (PvC)

## Installation & Setup

:: Openshift Setup Guide ::

Published: December 2025



In partnership with:



By: Kuldeep Sahu, Partner Solutions Engineer, Cloudera Inc.

# TABLE OF CONTENTS

## TABLE OF CONTENTS

- [Hardware Requirements and setup steps for installing Openshift cluster](#)
- [FreeIPA installation](#)
- [DNS Zone & Record Creation Steps](#)
- [Bastion Node setup and dependencies installation e.g. ansible, oc, virtctl, kubectl etc.](#)
- [Prerequisites](#)
- [Network and additional configurations e.g. DNS settings, static IP assignments etc.](#)
- [Openshift installation](#)
- [OCP Virt setup](#)
- [Cluster/Namespace creation, VM creation and configuration e.g. IP assignment etc. for Cloudera installation \(Base and DS\)](#)

## INSTALL/CONFIGURE OPENSHIFT

- [Install OpenShift using the Assisted Installer](#)

- [OpenShift Credentials](#)

- [Vefify MTU setting on all nodes:](#)

- [MACHINECONFIG:](#)

- [KubeletConfig](#)

## INSTALL/CONFIGURE OPERATORS

- [Install Local Storage Operator](#)

- [Install OpenShift Data Foundation \(Internal mode\)](#)

- [Create a Replica 2 Block Pool and StorageClass for ODF](#)

- [Configuring the Image Registry Operator to use CephFS storage with Red Hat OpenShift Data Foundation](#)

- [Install OpenShift Virtualization Operator](#)

- [Install NMState Operator](#)

- [Create Secondary Network](#)

- [Create Project for CDP PVC Base VMs](#)

- [Create SSH Keys in Project](#)

- [Auto Register RHEL VMs](#)

- [Create VMs on OpenShift Virtualization](#)

- [cloud-init section of the VirtualMachine YAML](#)

- [cldr-mngr VirtualMachine YAML](#)

- [cldr-utility VirtualMachine YAML](#)

- [pvcbase-masterXX VirtualMachine YAML](#)

- [pvcbase-workerXX VirtualMachine YAML](#)

- [OPTIONAL Method to create VMs](#)

## CONFIGURE THE LDAP IDENTITY PROVIDER WITH OPENSHIFT:

- [Create OpenShift cluster using the Assisted Installer](#)

- [DISCOVERY ISO IMAGE](#)

- [BADFISH](#)

- [Create OpenShift cluster with the Agent Based Installer \(ABI\)](#)

[install-config.yaml](#)  
[agent-config.yaml](#)  
[Create VMs for OpenShift Cluster \(OpenShift Virtualization\)](#)  
[pvcocp-master01.cdp.rdu2.scalelab.redhat.com-vms.yaml](#)  
[pvcocp-master02.cdp.rdu2.scalelab.redhat.com-vms.yaml](#)  
[pvcocp-master03.cdp.rdu2.scalelab.redhat.com-vms.yaml](#)  
[pvcocp-worker01.cdp.rdu2.scalelab.redhat.com-vms.yaml](#)  
[pvcocp-worker02.cdp.rdu2.scalelab.redhat.com-vms.yaml](#)  
[pvcocp-worker03.cdp.rdu2.scalelab.redhat.com-vms.yaml](#)  
[pvcocp-worker04.cdp.rdu2.scalelab.redhat.com-vms.yaml](#)  
[pvcocp-worker05.cdp.rdu2.scalelab.redhat.com-vms.yaml](#)  
[pvcocp-worker06.cdp.rdu2.scalelab.redhat.com-vms.yaml](#)  
[pvcocp-worker07.cdp.rdu2.scalelab.redhat.com-vms.yaml](#)  
[pvcocp-worker08.cdp.rdu2.scalelab.redhat.com-vms.yaml](#)  
[pvcocp-worker09.cdp.rdu2.scalelab.redhat.com-vms.yaml](#)  
[pvcocp-worker10.cdp.rdu2.scalelab.redhat.com-vms.yaml](#)  
[pvcocp-infra01.cdp.rdu2.scalelab.redhat.com-vms.yaml](#)  
[pvcocp-infra02.cdp.rdu2.scalelab.redhat.com-vms.yaml](#)  
[pvcocp-infra03.cdp.rdu2.scalelab.redhat.com-vms.yaml](#)  
[NTP](#)

---

#### [ALTERNATIVE CONFIGURATION STEPS](#)

[OPTIONAL Method to create VMs](#)

[ADD NODES TO THE CLUSTER](#)

[BADFISH](#)

[NodeNetworkConfigurationPolicy](#)

[OPTIONAL METHOD TO CREATE VM YAML](#)

# INSTALL/CONFIGURE OPENSIFT

## Install OpenShift using the Assisted Installer

Installation parameters:

Parameter	Setting/Value
Cluster name	ocp
Base domain	redhat.local
OpenShift version	4.19.12
CPU architecture	x86_64
Integrate with external partner platforms	No platform integration
Number of control plane nodes	3 (highly available cluster)
Hosts' network configuration	Static IP, bridges, and bonds
Configure via	YAML view
Networking stack type	IPv4
DNS	192.168.1.210
Machine network	192.168.2.0/24
Default gateway	192.168.2.1
api.ocp.redhat.local (API IP)	192.168.2.183
*.apps.ocp.redhat.local (Ingress IP)	192.168.2.184

**STEP 1.** Create a new OpenShift cluster using the OpenShift Assisted Installer from the Red Hat Hybrid Cloud Console:

<https://console.redhat.com/openshift/assisted-installer/clusters/~new>

Enter the cluster details using the installation parameters provided, then click **Next**.

## Install OpenShift with the Assisted Installer

[Assisted Installer documentation](#) [What's new in Assisted Installer?](#)

### 1 Cluster details

- 2 Static network configurations >
- 3 Operators
- 4 Host discovery
- 5 Storage
- 6 Networking
- 7 Review and create

### Cluster details

I'm installing on a disconnected/air-gapped/secured environment [Developer Preview](#)

**Cluster name \***

ocp



**Base domain \***

redhat.local

Enter the name of your domain [domainname] or [domainname.com]. This cannot be changed after cluster installed. All DNS records must include the cluster name and be subdomains of the base you enter. The full cluster address will be:  
ocp.redhat.local

**OpenShift version \***

OpenShift 4.19.12

[Learn more about OpenShift releases](#)

**CPU architecture**

x86\_64

Edit pull secret [?](#)

### Integrate with external partner platforms

No platform integration

**Number of control plane nodes [?](#)**

3 (highly available cluster)

Include custom manifests [?](#)

Additional manifests will be applied at the install time for advanced configuration of the cluster.

**Hosts' network configuration**

DHCP only  Static IP, bridges, and bonds

**Encryption of installation disks**

Control plane nodes

Workers

Arbiter

**Next**

**Cancel**

## NETWORK CONFIGURATION NOTE:

The three host servers used here have a dual-port 10Gb network adapter. Two YAML NMState configuration examples are provided to utilize both network adapter ports using a network bond:

- **Balance-rr aggregate mode:** both network adapters/ports are used simultaneously for network traffic. This does require a static EtherChannel to be configured/enabled on the Ethernet switch (not LACP-negotiated).
- **Active-Backup fail-over mode:** only one network adapter/port will be used at a time for network traffic. This does not require any special network switch configuration.

In my environment, I will be using the **Balance-rr** network bonding mode, and I have configured static EtherChannels for the two ports for each server on my Ethernet switch. If you do not have an Ethernet switch that is capable, you may use the **Active-Backup** network bonding mode, as it does not require any special network switch configuration.

You may also optionally configure OpenShift to use only a single network adapter/port without using a network bond. In this case, it is also possible to assign the second adapter/port for selected traffic types, such as only for storage access, only for use by VMs, or only for VM live migration traffic.

Both configurations provided also set the network mtu to 9000, which requires that jumbo Ethernet frames be enabled on the Ethernet switch for the server network ports in use.

### STEP 2. Select the button to configure via the **YAML view**.

Install OpenShift with the Assisted Installer  
Assisted Installer documentation [\[?\]](#) What's new in Assisted Installer?

1 Cluster details      2 Static network configurations      Network-wide configurations

Static network configurations  
Network configuration can be done using either the form view or YAML view. Configurations done in this step are for discovering hosts.  
Configure via:  Form view  YAML view

### STEP 3. Enter the static network configuration for Host 1 using the following YAML code example along with the information provided in the table.

**NOTE:** You will need to provide the MAC addresses for your server's network adapters, as they will be different from those provided.

From the YAML view, click on “**Start from scratch**” and then copy/paste the YAML NMState configuration provided below, while editing the IP address to use for the host.

Enter the MAC address and Interface name for the first network adapter in the host.

Click on “**Add another MAC to interface name mapping**” and then enter the MAC address and Interface name for the second network adapter in the host.

When you have finished entering the static network configuration for Host 1, select the check box for “**Copy the YAML content**” and click on “**Add another host configuration**”.

**Cluster details**

**Static network configurations**

Network configuration can be done using either the form view or YAML view. Configurations done in this step are for discovering hosts.

Upload, drag and drop, or copy and paste a YAML file that contains NMState into the editor for network configurations. Each host also needs the MAC to interface name mapping. Learn more about NMState [\[?\]](#)

**Host 1**

**Static network configurations**

```

1 dns-resolver:
2   config:
3     server:
4       - 192.168.1.210
5   interfaces:
6     - ipv4:
7       address:
8         - ip: 192.168.2.170 # change this address per host
9         | prefix-length: 24
10      dhcp: false
11      enabled: true
12      ipv6:
13        enabled: false
14      link-aggregation:
15        mode: balance-rr
16        options:
17          miimon: "140"
18        port:
19          - eno1
20          - eno2
21      name: bond0
22      state: up

```

**MAC to interface name mapping** [\[?\]](#)

MAC address *	Interface name *
70:DF:2F:F6:A1:2C	eno1
MAC address *	Interface name *
70:DF:2F:F6:A1:2D	eno2

[Add another MAC to interface name mapping](#)

**Add another host configuration**  **Copy the YAML content**

**Next** **Back** **Cancel** **View cluster events**

Parameter	Setting/Value
<b>Host 1 (c240m4-01.redhat.local)</b>	
IP address (IPv4)	192.168.2.170
MAC Address: 70:DF:2F:F6:A1:2C	Interface name: eno1
MAC Address: 70:DF:2F:F6:A1:2D	Interface name: eno2
<b>Host 2 (c240m4-02.redhat.local)</b>	
IP address (IPv4)	192.168.2.171

MAC Address: 70:DF:2F:F6:A2:2C	Interface name: eno1
MAC Address: 70:DF:2F:F6:A2:2D	Interface name: eno2
<b>Host 3 (C240m4-03.redhat.local)</b>	
IP address (IPv4)	192.168.2.172
MAC Address: 70:DF:2F:F6:A3:2C	Interface name: eno1
MAC Address: 70:DF:2F:F6:A3:2D	Interface name: eno2

The following NMState configuration examples are provided for reference only.

These are also in the OpenShift Assisted Installer documentation: [10.4.1. NMState configuration](#) and [10.4.3. Additional NMState configuration examples](#).

#### YAML NMState configuration using Balance-rr network bonding mode 0.

```

dns-resolver:
  config:
    server:
      - 192.168.1.210
interfaces:
- ipv4:
    address:
      - ip: 192.168.2.170 # change this address per host
        prefix-length: 24
    dhcp: false
    enabled: true
  ipv6:
    enabled: false
link-aggregation:
  mode: balance-rr
  options:
    miimon: "140"
  port:
    - eno1
    - eno2
  name: bond0
  state: up
  type: bond
  mtu: 9000
routes:

```

```
config:
- destination: 0.0.0.0/0
  next-hop-address: 192.168.2.1
  next-hop-interface: bond0
  table-id: 254
```

#### YAML NMState configuration using Active-Backup network bonding mode 1

```
dns-resolver:
config:
  server:
    - 192.168.1.210
interfaces:
- ipv4:
  address:
    - ip: 192.168.2.170 # change this address per host
      prefix-length: 24
    dhcp: false
    enabled: true
  ipv6:
    enabled: false
  link-aggregation:
    mode: active-backup
    options:
      miimon: "140"
    port:
      - eno1
      - eno2
  name: bond0
  state: up
  type: bond
  mtu: 9000
routes:
config:
- destination: 0.0.0.0/0
  next-hop-address: 192.168.2.1
  next-hop-interface: bond0
  table-id: 254
```

**STEP 4.** Enter the static network configuration for Host 2 using the following.

Edit the IP address, and enter the MAC address and Interface name for the first network adapter in the host.

Click on “**Add another MAC to interface name mapping**” and then enter the MAC address and Interface name for the second network adapter in the host.

When you have finished entering the static network configuration for Host 2, select the check box for “**Copy the YAML content**” and click on “**Add another host configuration**”.

Cluster details

② Static network configurations

3 Operators

4 Host discovery

5 Storage

6 Networking

7 Review and create

Static network configurations

Network configuration can be done using either the form view or YAML view. Configurations done in this step are for discovering hosts.

Upload, drag and drop, or copy and paste a YAML file that contains NMState into the editor for network configurations. Each host also needs the MAC to interface name mapping. [Learn more about NMState](#)

Host 1

MAC to Interface name mapping 70:DF:2F:F6:A1:2C -> eno1 1 left

Host 2

dns-resolver:

config:

server:

- 192.168.1.210

interfaces:

ipv4:

address:

- ip: 192.168.2.171 # change this address per host

prefix-length: 24

dhcp: false

enabled: true

ipv6:

enabled: false

link-aggregation:

mode: balance-rr

options:

mimon: "140"

port:

- eno1

- eno2

name: bond0

state: up

MAC to interface name mapping

MAC address \* Interface name \*

70:DF:2F:F6:A2:2C eno1

MAC address \* Interface name \*

70:DF:2F:F6:A2:2D eno2

+ Add another MAC to interface name mapping

Add another host configuration  Copy the YAML content

## STEP 5. Enter the static network configuration for Host 3.

Enter the MAC address and Interface name for the first network adapter in the host.

Click on “**Add another MAC to interface name mapping**” and then enter the MAC address and Interface name for the second network adapter in the host.

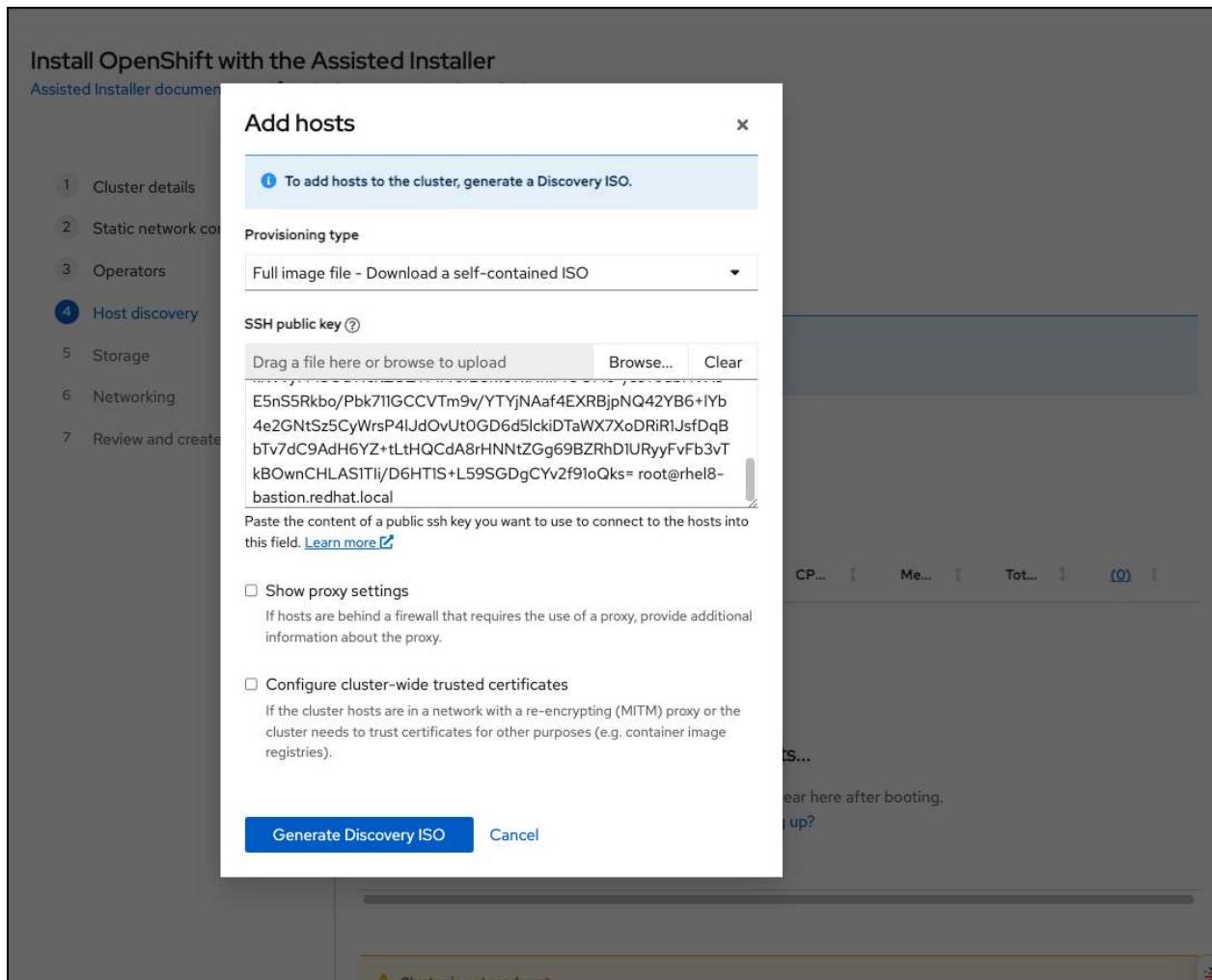
When you have finished entering the static network configuration for Host 3, click “**Next**”.

**STEP 6.** For Operators to install, we will install these separately. So, without selecting any here, click “**Next**”.

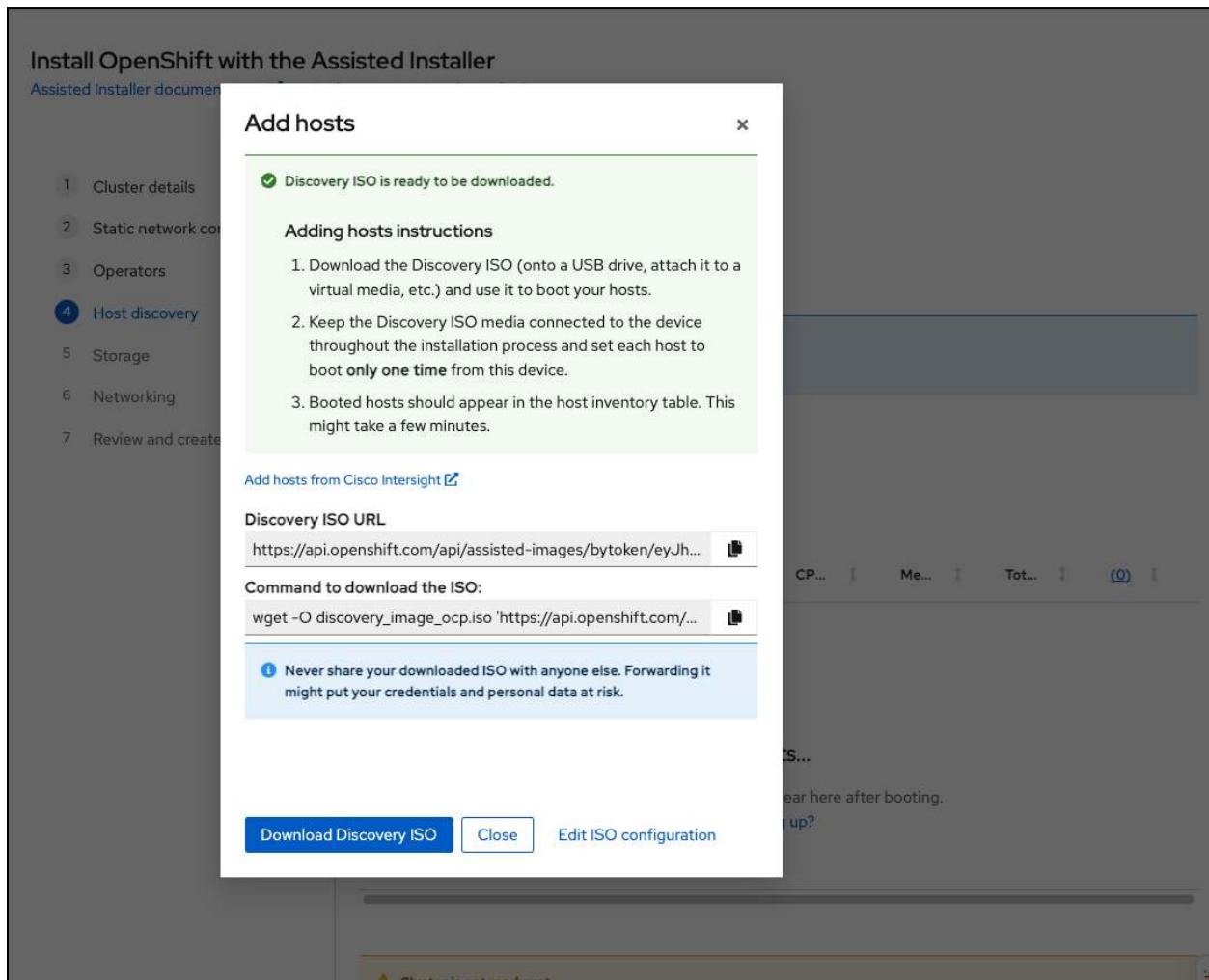
**STEP 7.** From the Host discovery screen, select “**Add hosts**” to generate a discovery ISO image.

The screenshot shows the 'Host discovery' step of a cluster setup wizard. On the left, a sidebar lists steps 1 through 7, with 'Host discovery' highlighted. The main area has a title 'Host discovery' and a large blue button labeled 'Add hosts'. Below it is a toggle switch for 'Run workloads on control plane nodes'. A light blue box contains 'Information & Troubleshooting' with links for 'Minimum hardware requirements' and 'Hosts not showing up?'. The 'Host Inventory' section shows a table with columns: Host name, Role, Status, Dis..., CP..., Me..., Tot..., and Actions. A progress bar at the bottom indicates '(0%)'. Below the table, a message says 'Waiting for hosts...' and notes that hosts might take a few minutes to appear after booting. A yellow warning box states 'Cluster is not ready yet.' with instructions to meet requirements like having exactly 3 dedicated control plane nodes. At the bottom are 'Next', 'Back', 'Cancel', and 'View cluster events' buttons.

**STEP 8.** From the “Add hosts” screen, select the **Full image file**, and paste your SSH public key (from `~/.ssh/id_rsa.pub` file), then click on “**Generate Discovery ISO**”.



**STEP 9.** Next, click on the “**Download Discovery ISO**” to download the discovery ISO image. Each server host will boot this image to start the server discovery process. When you have finished downloading the ISO image, click on “**Close**”.



Install OpenShift with the Assisted Installer  
[Assisted Installer documentation](#) [What's new in Assisted Installer?](#)

1 Cluster details  
 2 Static network configurations  
 3 Operators  
**4 Host discovery**  
 5 Storage  
 6 Networking  
 7 Review and create

**Host discovery**  
  
 Run workloads on control plane nodes [?](#)

**Information & Troubleshooting**  
[Minimum hardware requirements](#) [Hosts not showing up?](#)

**Host Inventory**  
 0 selected

Hostname	Role	Status	Discovered on	CPU ...	Memory	Total ...	(3)
c240m4-01.redhat.local	Auto-assign	Ready	9/26/2025, 6:08:05 PM	56	512.00 GiB	2.92 TB	⋮
c240m4-02.redhat.local	Auto-assign	Ready	9/26/2025, 6:07:08 PM	56	512.00 GiB	2.92 TB	⋮
c240m4-03.redhat.local	Auto-assign	Ready	9/26/2025, 6:07:46 PM	72	512.00 GiB	2.92 TB	⋮

```
sudo ip link set dev eno1 mtu 9000
sudo ip link set dev eno2 mtu 9000
sudo ip link set dev bond0 mtu 9000
sudo nmcli
```

```
sudo sgdisk -Z /dev/sdX
sudo wipefs -a /dev/sdX
```

**STEP 11.** From the Storage screen, expand each of the discovered hosts to ensure that the storage devices were discovered properly. Check that the disk that you want to install RHCOS onto is correctly set with a Role of “**installation disk**”. If not, use the drop-down selection to change the Role to the correct disk. In the screenshot, the installation disk is correctly set to a 120GB SSD. Click on “**Next**” to continue.

Install OpenShift with the Assisted Installer  
[Assisted Installer documentation](#) [What's new in Assisted Installer?](#)

1 Cluster details  
 2 Static network configurations  
 3 Operators  
 4 Host discovery  
**5 Storage**  
 6 Networking  
 7 Review and create

**Storage**

Hostname	Role	Status	Total storage	Number of ...	(3)
c240m4-01.redhat.local	Auto-assign	Ready	2.92 TB	11	⋮
c240m4-02.redhat.local	Auto-assign	Ready	2.92 TB	11	⋮
c240m4-03.redhat.local	Auto-assign	Ready	2.92 TB	11	⋮

11 Disks									
Name	Role	Limitat...	Format?	Drive t...	Size	Serial	Model	WWN	
sda	None	▼ 1	<input type="checkbox"/>	SSD	0.00 B	20111102-00000002	vKVM-Mapped_vHDD		
sdb	None	▼ 1	<input type="checkbox"/>	SSD	0.00 B	20111102-00000002	vKVM-Mapped_vFDD		
sdc	None	▼ 1	<input type="checkbox"/>	SSD	0.00 B	20111102-00000002	CIMC-Mapped_vHDD		
sdd (bootable)	Installation disk		<input checked="" type="checkbox"/>	SSD	120.03 GB	PHWA638304H4I20CGN	INTEL_SSDSC2BB12	0x55cd2e414d4af529	
sde	None	▼	<input type="checkbox"/>	SSD	400.09 GB	S18PNWAGB03349	MZ6ER400HAGL_003	0x5002538475bfdfa0	
sdf	None	▼	<input type="checkbox"/>	SSD	400.09 GB	S18PNEAFC01271	MZ6ER400HAGL_003	0x5002538454c716e0	
sdg	None	▼	<input type="checkbox"/>	SSD	1.60 TB	PHDV6325005YIP6EGN	INTEL_SSDSC2BB01	0x55cd2e414d453bd4	
sdh	None	▼	<input type="checkbox"/>	SSD	200.05 GB	S18NNEAF800120	MZ6ER200HAGM_003	0x500253845481bf60	
sdi	None	▼	<input type="checkbox"/>	SSD	200.05 GB	S18NNEAFA00354	MZ6ER200HAGM_003	0x5002538454a497d0	
sr0	None	▼ 2	<input type="checkbox"/>	ODD	1.07 GB	20111102-00000002	vKVM-Mapped_vDVD		
srl (bootable)	None	▼ 2	<input type="checkbox"/>	ODD	1.32 GB	20111102-00000002	CIMC-Mapped_vDVD		

All bootable disks, except for read-only disks, will be formatted during installation. Make sure to back up any critical data before proceeding.

[Next](#) [Back](#) [Cancel](#) [View cluster events](#)

**STEP 12.** From the Networking screen, enter the following parameters, and click on **Next** to continue.

Parameter	Setting/Value
Network Management	Cluster-Managed Networking
Machine Network	192.168.2.0/24 (192.168.2.0 - 192.168.2.255)
API IP	192.168.2.183
Ingress IP	192.168.2.184
Use advanced networking	Not selected
Use the same host discovery SSH key	Selected

**Install OpenShift with the Assisted Installer**

[Assisted Installer documentation](#) [What's new in Assisted Installer?](#)

1 Cluster details  
2 Static network configurations >  
3 Operators  
4 Host discovery  
5 Storage  
6 Networking  
7 Review and create

**Networking**

**Network Management**  
 Cluster-Managed Networking  User-Managed Networking

**Networking stack type**  
 IPv4  Dual-stack

**Machine network \***  
192.168.2.0/24 (192.168.2.0 - 192.168.2.255)

**API IP \***  
192.168.2.183

**Ingress IP \***  
192.168.2.184

Use advanced networking  
Configure advanced networking properties (e.g. CIDR ranges).

**Host SSH Public Key for troubleshooting after installation**  
 Use the same host discovery SSH key

**Host inventory**

Hostname	Role	Status	Action	IPv4 address	IPv6 address	MAC address	(3)
c240m4-01.redhat.local	Auto-assign	Ready	bond0	192.168.2.170/24	-	70:df:2f:f6:a1:2c	:
c240m4-02.redhat.local	Auto-assign	Ready	bond0	192.168.2.171/24	-	70:df:2f:f6:a2:2c	:
c240m4-03.redhat.local	Auto-assign	Ready	bond0	192.168.2.172/24	-	70:df:2f:f6:a3:2c	:

**Next** **Back** **Cancel** **View cluster events**

**STEP 13.** From the Review and create screen, verify the installation parameters and click on **"Install cluster"** to start the installation.

**Install OpenShift with the Assisted Installer**

[Assisted Installer documentation](#) [What's new in Assisted Installer?](#)

1 Cluster details  
2 Static network configurations >  
3 Operators  
4 Host discovery  
5 Storage  
6 Networking  
7 Review and create

**Review and create**

Preflight checks  Cluster preflight checks  Host preflight checks  Cluster support level: Full

**Cluster summary**

**Cluster details**

Cluster address	ocp.redhat.local
OpenShift version	4.19.12
CPU architecture	x86_64
Hosts' network configuration	Static IP

**Host inventory**

Hosts	3
Total cores	184
Total memory	1.50 TiB
Total storage	8.76 TB

**Networking**

Networking management type	Cluster-managed networking
Stack type	IPv4
Machine networks CIDR	192.168.2.0/24
API IP	192.168.2.183
Ingress IP	192.168.2.184

Advanced networking settings	
Cluster network CIDR	10.128.0.0/14
Cluster network host prefix	23
Service network CIDR	172.30.0.0/16
Networking type	Open Virtual Network (OVN)

**Install cluster**   **Back**   **Cancel**   **View cluster events**

**STEP 14.** Monitor the progress of the installation. Select “View cluster events” and the “Status” field to the right of each server host.

OpenShift

[Cluster List](#) > [Assisted Clusters](#) > ocp

## ocp

**Installation progress**

Started on  
9/25/2025, 5:50:00 PM

Preparing for installation 0%

Control Planes 0 control plane nodes installed

Initialization Pending

**Host inventory (3) ✓**

**Cluster summary**

[Abort installation](#)   [Download kubeconfig](#)   [View cluster events](#)

[Download Installation Logs](#)

[Back to all clusters](#)

Cluster List > Assisted Clusters > ocp

## ocp

**Installation progress**

**Started on**  
9/25/2025, 5:50:00 PM

Preparing for installation 0%

Control Planes	Initialization
Installing 3 control plane nodes	Pending

**Actions:** Abort installation, Download kubeconfig, View cluster events

Download Installation Logs

**Host inventory (3)**

Hostname	Role	Status	Discovered on	CPU Cor...	Memory	Total s...
c240m4-01.redhat.local	Control plane node, Worker	Preparing for installation	9/25/2025, 4:46:35 PM	56	512.00 GiB	2.92 TB
c240m4-02.redhat.local	Control plane node, Worker (bootstrap)	Preparing for installation	9/25/2025, 4:44:40 PM	56	512.00 GiB	2.92 TB
c240m4-03.redhat.local	Control plane node, Worker	Preparing for installation	9/25/2025, 4:49:23 PM	72	512.00 GiB	2.92 TB

**Cluster summary**

## ocp

**Installation progress**

**Started on**  
9/25/2025, 5:50:00 PM

Installing

Control Planes	Nodes
Installing 3 control plane nodes	3 nodes

**Actions:** Abort installation, Download Installation Logs, Download and save your Assisted Installer's service account

**Host inventory (3)**

Hostname	Role	Status	Discovered on	CPU Cor...	Memory	Total s...
c240m4-01.redhat.local	Control plane node, Worker	Installing 3/7	9/25/2025, 4:49:23 PM	72	512.00 GiB	2.92 TB
c240m4-02.redhat.local	Control plane node, Worker	Installing 3/7	9/25/2025, 4:49:23 PM	72	512.00 GiB	2.92 TB
c240m4-03.redhat.local	Control plane node, Worker	Installing 3/7	9/25/2025, 4:49:23 PM	72	512.00 GiB	2.92 TB

**Cluster Events**

Hosts Severity Filter by text

Time	Message
9/25/2025, 5:51:32 PM	Host c240m4-03.redhat.local: reached installation stage Starting installation: master
9/25/2025, 5:51:32 PM	Host c240m4-03.redhat.local: updated status from installing to installing-in-progress (Starting installation)
9/25/2025, 5:51:32 PM	c240m4-03.redhat.local: Performing quick format of disk sdd/dev/disk/by-id/wwn-0x55cd2e414d4af529
9/25/2025, 5:51:21 PM	Host c240m4-03.redhat.local: updated status from preparing-successful to installing (Installation is in progress)
9/25/2025, 5:51:21 PM	Host c240m4-02.redhat.local: updated status from preparing-successful to installing (Installation is in progress)
9/25/2025, 5:51:21 PM	Host c240m4-01.redhat.local: updated status from preparing-successful to installing (Installation is in progress)
9/25/2025, 5:51:17 PM	Updated status of the cluster to installing
9/25/2025, 5:51:13 PM	Host c240m4-02.redhat.local: updated status from preparing-for-installation to preparing-successful (Host finished successfully to prepare for installation)

1 - 10 of 61 | 1 of 7 | Close

**STEP 15.** When the installation is complete, download the kubeconfig file, and record the password for the kubeadm user.

The screenshot shows the Red Hat OpenShift Assisted Installer interface. At the top, it displays the start and install times: "Started on 9/25/2025, 5:50:00 PM" and "Installed on 9/25/2025, 7:00:11 PM". Below this, two green circular icons indicate the status of "Control Planes" and "Initialization", both marked as "Completed". A green banner at the bottom states "Installation completed successfully".

Below the banner are three buttons: "Launch OpenShift Console", "Download kubeconfig", and "View cluster events". There is also a link to "Download Installation Logs".

The "Web Console URL" is listed as <https://console-openshift-console.apps.ocp.redhat.local>, with a note "Not able to access the Web Console?".

For the "Username", "kubeadmin" is entered. The "Password" field contains ".....".

Two informational boxes are present: one about downloading the kubeconfig file and another about adding new hosts via a Discovery ISO.

The main content area shows "Host inventory (3)" with a table:

Hostname	Role	Status	Discovered on	CPU Cores	Memory	Total st...
c240m4-01.redhat.local	Control plane node, Worker	Installed	9/25/2025, 4:46:35 PM	56	512.00 GiB	2.92 TB
c240m4-02.redhat.local	Control plane node, Worker (bootstrap)	Installed	9/25/2025, 4:44:40 PM	56	512.00 GiB	2.92 TB
c240m4-03.redhat.local	Control plane node, Worker	Installed	9/25/2025, 4:49:23 PM	72	512.00 GiB	2.92 TB

Below the table, under "Cluster summary", it says "Cluster support level: Full" and "Your installed cluster will be fully supported".

The "Cluster details" section lists:

- Cluster address: ocp.redhat.local
- OpenShift version: 4.19.12
- CPU architecture: x86\_64
- Hosts' network configuration: Static IP

The "Host inventory" section provides summary statistics:

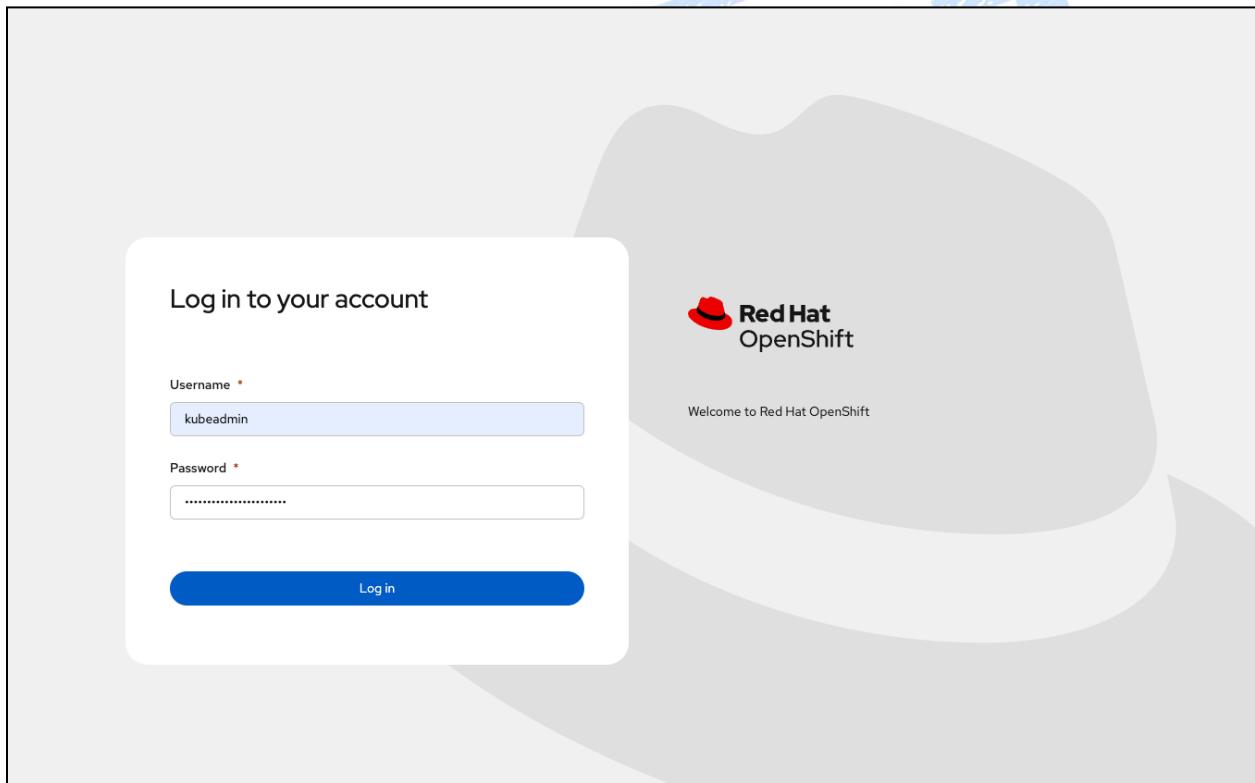
- Hosts: 3
- Total cores: 184
- Total memory: 1.50 TiB
- Total storage: 8.76 TB

Networking	
Networking management type	Cluster-managed networking
Stack type	IPv4
Machine networks CIDR	192.168.2.0/24
API IP	192.168.2.183
Ingress IP	192.168.2.184
Advanced networking settings	
Cluster network CIDR	10.128.0.0/14
Cluster network host prefix	23
Service network CIDR	172.30.0.0/16
Networking type	Open Virtual Network (OVN)

**STEP 16.** Login to the OpenShift console, click on the Web Console URL:

<https://console-openshift-console.apps.ocp.redhat.local/>

When prompted for the Username and Password, enter the kubeadmin user and the password recorded previously from the installation progress screen.



# OpenShift Credentials

Web Console URL: <https://console.openshift-console.apps.ocp.redhat.local/>

<https://console.redhat.com/openshift/assisted-installer/clusters/2e9aece0-66f8-42d5-92e0-6f802344fa1d>

Username: kubeadmin (select login with kube:admin)

Password: FvCFE-WIS52-EezGx-qDWY7

Username: admin (select login with ldap)

Password: redhat123

## Verify MTU setting on all nodes:

Run the following command to verify that the cluster installation set the MTU as expected:

```
[root@rhel8-bastion ~]# for i in $(oc get node -o jsonpath='{.items[*].metadata.name }'); do oc debug node/${i} -- chroot /host ip -d link |grep -1 maxmtu|grep -1 bond0; done
```

### Sample output

```
To use host binaries, run `chroot /host`  
Link/Loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00 promiscuity 0 allmulti 0 minmtu 0 maxmtu 0 numtxqueues 1 numrxqueues 1 gso_max_size  
65536 gso_max_segs 65535 tso_max_size 524280 tso_max_segs 65535 gro_max_size 65536 gso_ipv4_max_size 65536 gro_ipv4_max_size 65536  
2: eno1: <BROADCAST,MULTICAST,SLAVE,UP,LOWER_UP> mtu 9000 qdisc mq master bond0 state UP mode DEFAULT group default qlen 1000  
Link/ether 70:df:2f:f6:a1:2c brd ff:ff:ff:ff:ff:ff promiscuity 1 allmulti 0 minmtu 68 maxmtu 9000  
--  
--  
3: eno2: <BROADCAST,MULTICAST,SLAVE,UP,LOWER_UP> mtu 9000 qdisc mq master bond0 state UP mode DEFAULT group default qlen 1000  
Link/ether 70:df:2f:f6:a1:2c brd ff:ff:ff:ff:ff:ff permaddr 70:df:2f:f6:a1:2d promiscuity 1 allmulti 0 minmtu 68 maxmtu 9000  
--  
openvswitch numtxqueues 1 numrxqueues 1 gso_max_size 65536 gso_max_segs 65535 tso_max_size 65536 tso_max_segs 65535 gro_max_size 65536  
gso_ipv4_max_size 65536 gro_ipv4_max_size 65536  
10: bond0: <BROADCAST,MULTICAST,MASTER,UP,LOWER_UP> mtu 9000 qdisc noqueue master ovs-system state UP mode DEFAULT group default qlen 1000  
Link/ether 70:df:2f:f6:a1:2c brd ff:ff:ff:ff:ff:ff promiscuity 1 allmulti 0 minmtu 68 maxmtu 65536
```

```
Removing debug pod ...  
Starting pod/c240m4-02redhatLocal-debug-4ds4j ...  
To use host binaries, run `chroot /host`  
Link/Loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00 promiscuity 0 allmulti 0 minmtu 0 maxmtu 0 numtxqueues 1 numrxqueues 1 gso_max_size  
65536 gso_max_segs 65535 tso_max_size 524280 tso_max_segs 65535 gro_max_size 65536 gso_ipv4_max_size 65536 gro_ipv4_max_size 65536  
2: eno1: <BROADCAST,MULTICAST,SLAVE,UP,LOWER_UP> mtu 9000 qdisc mq master bond0 state UP mode DEFAULT group default qlen 1000  
Link/ether 70:df:2f:f6:a2:2c brd ff:ff:ff:ff:ff:ff promiscuity 1 allmulti 0 minmtu 68 maxmtu 9000  
--  
--  
3: eno2: <BROADCAST,MULTICAST,SLAVE,UP,LOWER_UP> mtu 9000 qdisc mq master bond0 state UP mode DEFAULT group default qlen 1000  
Link/ether 70:df:2f:f6:a2:2c brd ff:ff:ff:ff:ff:ff permaddr 70:df:2f:f6:a2:2d promiscuity 1 allmulti 0 minmtu 68 maxmtu 9000  
--  
openvswitch numtxqueues 1 numrxqueues 1 gso_max_size 65536 gso_max_segs 65535 tso_max_size 65536 tso_max_segs 65535 gro_max_size 65536  
gso_ipv4_max_size 65536 gro_ipv4_max_size 65536  
10: bond0: <BROADCAST,MULTICAST,MASTER,UP,LOWER_UP> mtu 9000 qdisc noqueue master ovs-system state UP mode DEFAULT group default qlen 1000  
Link/ether 70:df:2f:f6:a2:2c brd ff:ff:ff:ff:ff:ff promiscuity 1 allmulti 0 minmtu 68 maxmtu 65536
```

```
Removing debug pod ...  
Starting pod/c240m4-03redhatLocal-debug-hq9tl ...  
To use host binaries, run `chroot /host`  
Link/Loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00 promiscuity 0 allmulti 0 minmtu 0 maxmtu 0 numtxqueues 1 numrxqueues 1 gso_max_size  
65536 gso_max_segs 65535 tso_max_size 524280 tso_max_segs 65535 gro_max_size 65536 gso_ipv4_max_size 65536 gro_ipv4_max_size 65536  
2: eno1: <BROADCAST,MULTICAST,SLAVE,UP,LOWER_UP> mtu 9000 qdisc mq master bond0 state UP mode DEFAULT group default qlen 1000
```

```
Link/ether 70:df:2f:f6:a3:2c brd ff:ff:ff:ff:ff:ff promiscuity 1 allmulti 0 minmtu 68 maxmtu 9000
--
--  
3: eno2: <BROADCAST,MULTICAST,SLAVE,UP,LOWER_UP> mtu 9000 qdisc mq master bond0 state UP mode DEFAULT group default qlen 1000
Link/ether 70:df:2f:f6:a3:2c brd ff:ff:ff:ff:ff:ff permaddr 70:df:2f:f6:a3:2d promiscuity 1 allmulti 0 minmtu 68 maxmtu 9000
```

**DRAFT**

# MACHINECONFIG:

## [6.10.3. Automatically allocating resources for nodes](#)

### KubeletConfig

OpenShift Container Platform can automatically determine the optimal **system-reserved** CPU and memory resources for nodes associated with a specific machine config pool and update the nodes with those values when the nodes start. By default, the **system-reserved** CPU is **500m** and **system-reserved** memory is **1Gi**.

To automatically determine and allocate the **system-reserved** resources on nodes, create a **KubeletConfig** custom resource (CR) to set the **autoSizingReserved: true** parameter. A script on each node calculates the optimal values for the respective reserved resources based on the installed CPU and memory capacity on each node. The script takes into account that increased capacity requires a corresponding increase in the reserved resources.

Automatically determining the optimal **system-reserved** settings ensures that your cluster is running efficiently and prevents node failure due to resource starvation of system components, such as CRI-O and kubelet, without needing to manually calculate and update the values.

This feature is disabled by default.

### Procedure

STEP 1. Create a custom resource (CR) for your configuration change:  
**Sample configuration for a resource allocation CR**

```
[root@bastion ]# cat << EOF | oc apply -f -
apiVersion: machineconfiguration.openshift.io/v1
kind: KubeletConfig
metadata:
  name: dynamic-node-master
spec:
  machineConfigPoolSelector:
    matchLabels:
      pools.operator.machineconfiguration.openshift.io/master: ""
  autoSizingReserved: true
EOF
```

---

```
[root@bastion ]# cat << EOF | oc apply -f -
apiVersion: machineconfiguration.openshift.io/v1
kind: KubeletConfig
metadata:
  name: dynamic-node-worker
spec:
  machineConfigPoolSelector:
    matchLabels:
      pools.operator.machineconfiguration.openshift.io/worker: ""
  autoSizingReserved: true
EOF
```

The previous example enables automatic resource allocation on all worker nodes. OpenShift Container Platform drains the nodes, applies the kubelet config, and restarts the nodes.

**WARNING:** This will trigger a serial restart of nodes using this MachineConfigPool. Refer also to the documentation for [Modifying nodes](#) to see how to modify the reservations with those values

### Verification

STEP 1. View the /etc/node-sizing.env file on a node(s) you configured by entering the following command:

```
[root@bastion ]# for i in $(oc get node -o jsonpath='{.items[*].metadata.name }'); do oc debug node/${i} -- chroot /host cat /etc/node-sizing.env; done
Temporary namespace openshift-debug-spjgv is created for debugging node...
Starting pod/c240m4-01redhatlocal-debug-x56qq ...
To use host binaries, run `chroot /host`
SYSTEM_RESERVED_MEMORY=17Gi
SYSTEM_RESERVED_CPU=0.91
SYSTEM_RESERVED_ES=1Gi
Removing debug pod ...
Temporary namespace openshift-debug-spjgv was removed.
Temporary namespace openshift-debug-8wxrm is created for debugging node...
Starting pod/c240m4-02redhatlocal-debug-6k858 ...
To use host binaries, run `chroot /host`
SYSTEM_RESERVED_MEMORY=17Gi
SYSTEM_RESERVED_CPU=0.91
SYSTEM_RESERVED_ES=1Gi
Removing debug pod ...
Temporary namespace openshift-debug-8wxrm was removed.
Temporary namespace openshift-debug-m7tpk is created for debugging node...
Starting pod/c240m4-03redhatlocal-debug-s65tx ...
To use host binaries, run `chroot /host`
SYSTEM_RESERVED_MEMORY=17Gi
SYSTEM_RESERVED_CPU=0.91
SYSTEM_RESERVED_ES=1Gi
```

```
Removing debug pod ...
Temporary namespace openshift-debug-m7tpk was removed.
```

Or run it on a single node:

```
[root@bastion ]# oc debug node/c240m4-03.redhat.local -- chroot /host cat /etc/node-sizing.env
```

## INSTALL/CONFIGURE OPERATORS

### Install Local Storage Operator

#### 2.1. Install Local Storage Operator

**NOTE:** If re-installing, ensure that the SSDs are empty, and wiped before configuring ODF.

Access each node using: `ssh core@c240m4-0x`, or `oc debug node/c240m4-0x`, then run the following commands:

```
[root@bastion ]# ssh core@c240m4-01
[core@c240m4-01 ]# sudo sgdisk -Z /dev/sdX
[core@c240m4-01 ]# sudo wipefs -a /dev/sdX
```

Use the `sudo lsblk` command to identify the disks' device names, and use them in place of `/dev/sdX` above.

**STEP 1.** In OperatorHub, search on local storage, select the **Local Storage** Operator.

Project: All Projects ▾

## OperatorHub



Discover Operators from the Kubernetes community and Red Hat partners, curated by Red Hat. You can purchase commercial software through [Red Hat Marketplace](#). You can install Operators on your clusters to provide optional add-ons and shared services to your developers. After installation, the Operator capabilities will appear in the [Software Catalog](#), providing a self-service experience.

All Items

All Items

local storage ×

2 items

AI/Machine Learning  
Application Runtime  
Big Data  
Cloud Provider  
Database  
Developer Tools  
Development Tools  
Drivers and plugins  
Integration & Delivery  
Logging & Tracing  
Modernization & Migration  
Monitoring  
Networking  
Observability  
OpenShift Optional  
Openshift Optional  
Security  
Storage  
Streaming & Messaging  
Other



Red Hat

Local Storage  
provided by Red Hat

Configure and use local storage volumes.



Red Hat

LVM Storage  
provided by Red Hat

Logical volume manager storage provides dynamically provisioned local storage for container...

## STEP 2. Click on *Install*.

The screenshot shows the OperatorHub interface with the following details:

**Project: All Projects**

**OperatorHub**

Discover Operators from [Red Hat Marketplace](#). Operator capabilities will be updated automatically.

**Local Storage**  
4.19.0-202509091807 provided by Red Hat

**Install**

**Channel**: stable (selected)

Operator that configures local storage volumes for use in Kubernetes and OpenShift. OpenShift 4.2 and above are the only supported OpenShift versions.

**Version**: 4.19.0-202509...

**Capability level**:

- Basic Install
- Seamless Upgrades
- Full Lifecycle
- Deep Insights
- Auto Pilot

**Source**: Red Hat

**Provider**: Red Hat

**Infrastructure features**:

- Disconnected
- Designed for FIPS
- Proxy-aware

**Valid Subscriptions**:

- OpenShift Kubernetes Engine
- OpenShift Virtualization Engine

### STEP 3. Click *Install*.

OperatorHub > Operator Installation

#### Install Operator

Install your Operator by subscribing to one of the update channels to keep the Operator up to date. The strategy determines either manual or automatic updates.

Update channel \* 

Version \* 

Installation mode \* 

All namespaces on the cluster (default)  
This mode is not supported by this Operator

A specific namespace on the cluster  
Operator will be available in a single Namespace only.

Installed Namespace \* 

Operator recommended Namespace:  **openshift-local-storage**

Select a Namespace

**Namespace creation**  
Namespace **openshift-local-storage** does not exist and will be created.

Enable Operator recommended cluster monitoring on this Namespace

Update approval \* 

Automatic

Manual

**Install** **Cancel**

**Local Storage**  
provided by Red Hat

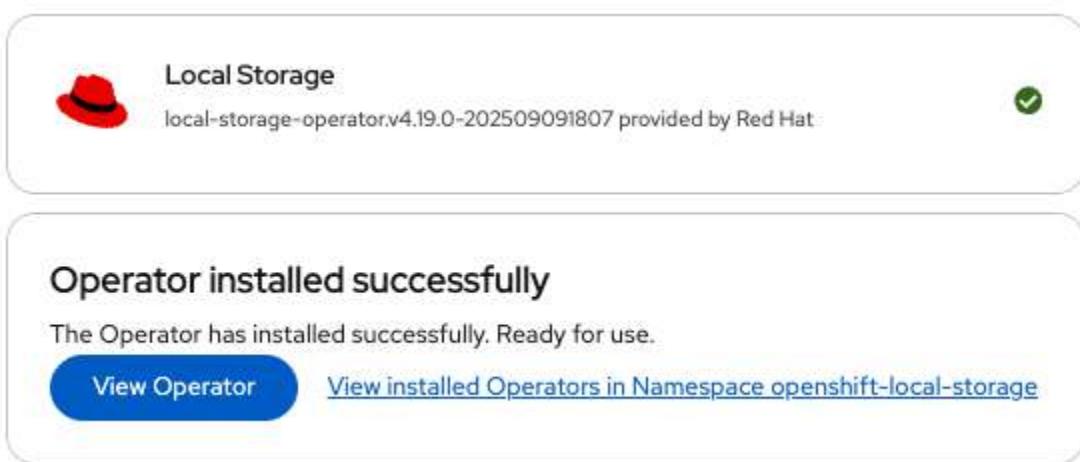
**Provided APIs**

**LV Local Volume**  
Manage local storage volumes for OpenShift

**LVS Local Volume Set**  
A Local Volume set allows you to filter a set of storage volumes, group them and create a dedicated storage class to consume storage from the set of volumes.

**LVD Local Volume Discovery**  
Discover list of potentially usable disks on the chosen set of nodes

STEP 4. Ensure that the operator installs successfully. Then continue with the next step by installing ODF. The installation process for ODF will configure the Local Storage operator.



# Install OpenShift Data Foundation (Internal mode)

**STEP 1.** In OperatorHub, search on odf, select “*OpenShift Data Foundation*”.

The screenshot shows the OperatorHub interface with a search bar containing 'odf'. Two items are listed: 'ODF Multicluster Orchestrator' and 'OpenShift Data Foundation'. Both items are provided by Red Hat and have a red hat icon. The 'OpenShift Data Foundation' item has a brief description: 'Orchestrator for OpenShift Data Foundation clusters running across multiple OpenShift...'. A sidebar on the left lists various operator categories like AI/Machine Learning, Application Runtime, Big Data, etc. A star icon is in the top right corner.

**STEP 2.** Click on “*Install*”.

The screenshot shows the detailed view of the 'OpenShift Data Foundation' operator. It includes the following sections: 'Channel' (stable-4.19), 'Version' (4.19.4-rhodf), 'Capability level' (Basic Install, Seamless Upgrades, Full Lifecycle, Deep Insights selected; Auto Pilot unselected), 'Source' (Red Hat), 'Provider' (Red Hat), 'Infrastructure features' (Disconnected, Container Storage Interface, Designed for FIPS, Proxy-aware), and 'Valid Subscriptions'. The main description states: 'The OpenShift Data Foundation operator is the primary operator for OpenShift Data Foundation. It serves to facilitate the other operators in OpenShift Data Foundation by performing administrative tasks outside their scope as well as watching and configuring their CustomResources.' A large blue 'Install' button is prominently displayed at the top left of the details panel.

### STEP 3. Click on “*Install*”.

OperatorHub > Operator Installation

#### Install Operator

Install your Operator by subscribing to one of the update channels to keep the Operator up to date. The strategy determines either manual or automatic updates.

**Update channel \***  stable-4.19

**Version \*** 4.19.5-rhodf

**Installation mode \***  
 All namespaces on the cluster (default)  
This mode is not supported by this Operator  
 A specific namespace on the cluster  
Operator will be available in a single Namespace only.

**Installed Namespace \***  
 Operator recommended Namespace:  openshift-storage  
 Select a Namespace

**Namespace creation**  
Namespace **openshift-storage** does not exist and will be created.

**Update approval \***   Automatic  
 Manual

**Console plugin \***   Enable  
 Disable

**Install** **Cancel**

### STEP 4. Ensure that the operator installs successfully. Then, click on “**Configure Operator**”.

 **OpenShift Data Foundation**  
odf-operator.v4.19.5-rhodf provided by Red Hat 

#### Configure Operator

The Operator has installed successfully. Complete the next configuration steps to prepare it for use.

**Configure Operator** [View installed Operators in Namespace \*\*openshift-storage\*\*](#)

**STEP 4.** Select “**Create a new StorageClass using local storage devices**”, “**Use Ceph RBD as the default StorageClass**”, and “**Set default StorageClass for virtualization**” then select “**Next**”.

**Create StorageSystem**

Create a StorageSystem to represent your Data Foundation system and all its required storage and computing resources.

Namespace: openshift-storage

1 Backing storage      Deployment type: Full deployment

2 Create local volume set

3 Capacity and nodes

4 Security and network

5 Review and create

**Backing storage type:**

Use an existing StorageClass  
Data Foundation will use an existing StorageClass available on your hosting platform.

Create a new StorageClass using local storage devices  
Data Foundation will use a StorageClass provided by the Local Storage Operator (LSO) on top of your attached drives. This option is available on any platform with devices attached to nodes.

Connect an external storage platform  
Data Foundation will create a dedicated StorageClass.

Enable network file system (NFS)  
Allow NFS to use low resources by default.

Use Ceph RBD as the default StorageClass  
Configure default RBD StorageClass to avoid adding manual annotations within a StorageClass and selecting a specific StorageClass when making storage requests or provisions in your PVCs.

Set default StorageClass for virtualization  
If enabled, RBD virtualization StorageClass will be marked as the default for KubeVirt VM disks (persistent volumes) upon installation.

Use external PostgreSQL  
Allow Noobaa to connect to an external postgres server

**Next**    **Back**    **Cancel**

**STEP 5.** Enter a name for the LocalVolumeSet and StorageClass, and select the appropriate settings for the storage across your cluster.

In this example, a 3-node OpenShift cluster is being used, with each node having 3 SSDs installed, for a total of 9 Disks.

Then click on “**Next**”.

**Create StorageSystem**

Create a StorageSystem to represent your Data Foundation system and all its required storage and computing resources.

Namespace: openshift-storage

1 Backing storage  
 2 Create local volume set  
 3 Capacity and nodes  
 4 Security and network  
 5 Review and create

**LocalVolumeSet name \***  
 odf-localvolumeset

A LocalVolumeSet will be created to allow you to filter a set of disks, group them and create a dedicated StorageClass to consume storage from them.

**StorageClass name**  
 odf-localvolumeset

**Selected capacity**  
 3 Node | 9 Disk  
 6.55 TiB  
 Out of 6.55 TiB

Filter disks by  
 Disks on all nodes (3 node)  
 Uses the available disks that match the selected filters on all nodes.

Disks on selected nodes  
 Uses the available disks that match the selected filters only on selected nodes.

**Disk type**  
 SSD / NVMe

**Info** Disk type is set to SSD/NVMe  
 Data Foundation supports only SSD/NVMe disk type for internal mode deployment.

» Advanced

Next Back Cancel

**STEP 6.** Select “Yes” to continue.

**Create LocalVolumeSet** ×

After the LocalVolumeSet is created you won't be able to edit it.

**Note:** If you wish to use the Arbiter stretch cluster, a minimum of 4 nodes (2 different zones, 2 nodes per zone) and 1 additional zone with 1 node is required. All nodes must be pre-labeled with zones in order to be validated on cluster creation.

Are you sure you want to continue?

**Yes** **Cancel**

**STEP 7.** On the “Capacity and nodes” screen you will see confirmation of the selected capacity, and nodes. Select the amount of resources to assign to ODF; either **Lean**, **Balanced** or **Performance** mode. For most cases, you would select the default “*Balanced*” mode setting. Select “**Next**”.

### Create StorageSystem

Create a StorageSystem to represent your Data Foundation system and all its required storage and computing resources.

Namespace: openshift-storage

1 Backing storage 2 Create local volume set <b>3 Capacity and nodes</b> 4 Security and network 5 Review and create	<b>Selected capacity</b>  <b>Available raw capacity</b> 6.54 TiB <small>The available capacity is based on all attached disks associated with the selected StorageClass odf-localvolumeset</small>  <b>Selected nodes</b> <small>Selected nodes are based on the StorageClass odf-localvolumeset and with a recommended requirement of 14 CPU and 34 GiB RAM per node.</small> <small>If not labeled, the selected nodes are labeled <code>cluster.ocs.openshift.io/openshift-storage=""</code> to make them target hosts for Data Foundation's components.</small> <table border="1" style="width: 100%; border-collapse: collapse; margin-top: 10px;"> <thead> <tr> <th>Name</th> <th>Role</th> <th>CPU</th> <th>Memory</th> <th>Zone</th> </tr> </thead> <tbody> <tr> <td><a href="#">N c240m4-01.redhat.local</a></td> <td>control-plane, master, worker</td> <td>56</td> <td>503.6 GiB</td> <td>-</td> </tr> <tr> <td><a href="#">N c240m4-02.redhat.local</a></td> <td>control-plane, master, worker</td> <td>56</td> <td>503.6 GiB</td> <td>-</td> </tr> <tr> <td><a href="#">N c240m4-03.redhat.local</a></td> <td>control-plane, master, worker</td> <td>72</td> <td>503.8 GiB</td> <td>-</td> </tr> </tbody> </table> <p>3 node selected (184 CPUs and 1.48 TiB on 0 zone)</p>	Name	Role	CPU	Memory	Zone	<a href="#">N c240m4-01.redhat.local</a>	control-plane, master, worker	56	503.6 GiB	-	<a href="#">N c240m4-02.redhat.local</a>	control-plane, master, worker	56	503.6 GiB	-	<a href="#">N c240m4-03.redhat.local</a>	control-plane, master, worker	72	503.8 GiB	-
Name	Role	CPU	Memory	Zone																	
<a href="#">N c240m4-01.redhat.local</a>	control-plane, master, worker	56	503.6 GiB	-																	
<a href="#">N c240m4-02.redhat.local</a>	control-plane, master, worker	56	503.6 GiB	-																	
<a href="#">N c240m4-03.redhat.local</a>	control-plane, master, worker	72	503.8 GiB	-																	

<b>Configure performance</b> ⓘ <small>Select a profile to customise the performance of the Data Foundation cluster to meet your requirements.</small>	<div style="border: 1px solid #ccc; padding: 5px; margin-bottom: 5px;"> <b>Balanced mode</b> </div> <div style="border: 1px solid #ccc; padding: 5px; margin-bottom: 5px;"> <b>Lean mode</b>            CPUs required: 29, Memory required: 81 GiB         </div> <div style="border: 1px solid #ccc; padding: 5px; margin-bottom: 5px;"> <b>Balanced mode</b> </div> <div style="border: 1px solid #ccc; padding: 5px; margin-bottom: 5px;"> <b>Performance mode</b>            CPUs required: 57, Memory required: 120 GiB         </div> <p><small>data use only</small></p> <p><small>When the nodes in the selected StorageClass are spread across fewer than 3 availability zones, the StorageCluster will be deployed with the host based failure domain.</small></p>
--	--

[Next](#)   [Back](#)   [Cancel](#)

**STEP 8.** Optionally enable encryption or to specify advanced networking settings, then select “**Next**”.

### Create StorageSystem

Create a StorageSystem to represent your Data Foundation system and all its required storage and computing resources.

Namespace: openshift-storage

1 Backing storage  
2 Create local volume set  
3 Capacity and nodes  
**4 Security and network**  
5 Review and create

**Encryption**

Enable data encryption for block and file storage  
Data encryption for block and file storage. MultiCloud Object Gateway supports encryption for objects by default.

In-transit encryption  
Encrypts all Ceph traffic including data, using Ceph msgrv2

**Network**

Default (Pod) ⓘ  
 Isolate network using Multus  
Multus allows a network separation between the data operations and the control plane operations.

**Next** **Back** **Cancel**

## STEP 9. Click on “Create StorageSystem”.

The screenshot shows the 'Create StorageSystem' wizard in progress, specifically step 5: 'Review and create'. The left sidebar lists steps 1 through 5. Step 5 is highlighted with a blue circle. The main panel displays configuration details:

- Backing storage**: Deployment type: Full deployment, Network file system: Disabled.
- Capacity and nodes**: Ceph RBD as the default StorageClass: Enabled, Default StorageClass for virtualization: Enabled, Backing storage type: odf-localvolumeset.
- Security and network**: Cluster capacity: 6.54 TiB, Selected nodes: 3 node, CPU and memory: 184 CPU and 1.48 TiB memory, Performance profile: Balanced, Automatic capacity scaling: Disabled, Zone: 0 zone, Taint nodes: Disabled.
- Encryption**: Enabled.
- In-transit encryption**: Enabled.
- Network**: Default (OVN).

At the bottom are three buttons: 'Create StorageSystem' (highlighted in blue), 'Back', and 'Cancel'.

## STEP 10. When complete, you will see a Ready status.

The screenshot shows the 'Data Foundation' interface under the 'Storage Systems' tab. A table lists existing StorageSystems:

Name	Status	Raw Capacity	Used capacity	IOPS	Throughput	Latency
SS_ocs-storagecluster	Phase: <span style="color: green;">Ready</span>	6.55 TiB	33.01 GiB	10.73 IOPS	28.4 KBps	1.09 ms

At the top right is a 'Create StorageSystem' button.

## Create a Replica 2 Block Pool and StorageClass for ODF

Perform the following from the OpenShift console to create the Replica 2 Block Pool:

**STEP 1.** Click **Storage > Data Foundation**.

**STEP 2.** In the **Storage systems** tab, select the storage system and then click the **Storage pools** tab.

**STEP 3.** Click **Create storage pool**.

**STEP 4.** Select **Volume type** as **Block**.

**STEP 5.** Enter **Pool name**, such as "ocs-storagecluster-cephblockpool-replica2".

**STEP 6.** Select Data protection policy as **2-way Replication**.

The screenshot shows the 'Create storage pool' dialog box. At the top, it says 'Create storage pool'. Below that, a note states: 'A storage pool is a logical entity which provides capacity to applications and workloads. With pools you can support policies for data resiliency and storage efficiency.' Under 'Volume type \*', there are two options: 'Filesystem' (radio button) and 'Block' (radio button, selected). Under 'Pool name \*', the input field contains 'ocs-storagecluster-cephblockpool-replica2' with a green checkmark icon to its right. Under 'Data protection policy \*', a dropdown menu is set to '2-way Replication'. Under 'Data compression', there is an unchecked checkbox with the text 'Optimize storage efficiency by enabling data compression within replicas.' At the bottom, there are 'Create' and 'Cancel' buttons.

**STEP 7.** Click **Create**.

After the Pool is created, perform the following from the OpenShift console to create the Replica 2 StorageClass:

**STEP 1.** Navigate to **Storage > StorageClasses**.

**STEP 2.** Click **Create Storage Class**.

**STEP 3.** Enter a **name**, such as "ocs-storagecluster-ceph-replica2-rbd"; then select the **openshift-storage.rbd,csi.ceph.com** provisioner, and select the Storage Pool entered in the previous step, such as "ocs-storagecluster-cephblockpool-replica2".

## StorageClass

[Edit YAML](#)

---

**Name \***  
ocs-storagecluster-ceph-replica2-rbd

**Description**

**Reclaim policy \***  
Delete

Determines what happens to persistent volumes when the associated persistent volume claim is deleted. Defaults to "Delete"

**Volume binding mode \***  
WaitForFirstConsumer

Determines when persistent volume claims will be provisioned and bound. Defaults to "WaitForFirstConsumer"

**Provisioner \***  
openshift-storage.rbd.csi.ceph.com

Determines what volume plugin is used for provisioning PersistentVolumes.

**Storage system \***  
SS ocs-storagecluster-storesystem

Select a StorageSystem for your workloads.

**Storage Pool \***  
ocs-storagecluster-cephblockpool-replica2

Storage pool into which volume data shall be stored

Enable Encryption  
An encryption key will be generated for each PersistentVolume created using this StorageClass.

**Additional parameters**  
Specific fields for the selected provisioner.

Parameter	Value
<input type="text"/> Parameter	<input type="text"/> Value

[+ Add Parameter](#)

Allow PersistentVolumeClaims to be expanded

[Create](#)
|
[Cancel](#)

**STEP 4.** Click **Create**.

## Configuring the Image Registry Operator to use CephFS storage with Red Hat OpenShift Data Foundation

### Procedure

**STEP 1.** Create a PVC to use the cephfs storage class. For example:

```
[root@bastion ]# cat <<EOF | oc apply -f -
apiVersion: v1
kind: PersistentVolumeClaim
metadata:
  name: registry-storage-pvc
  namespace: openshift-image-registry
spec:
  accessModes:
    - ReadWriteMany
  resources:
    requests:
      storage: 100Gi
  storageClassName: ocs-storagecluster-cephfs
EOF
```

**STEP 2.** Configure the image registry to use the CephFS file system storage by entering the following command:

```
[root@bastion ]# oc patch config.image/cluster -p
'{"spec":{"managementState":"Managed","replicas":2,"storage":{"managementState":"Unmanaged","pvc":{"claim":"registry-storage-pvc"}}}}' --type=merge
```

### Example

```
apiVersion: imageregistry.operator.openshift.io/v1
kind: Config
metadata:
  name: cluster
spec:
  logLevel: Normal
  managementState: Managed
  replicas: 2
  storage:
    managementState: Unmanaged
    pvc:
      claim: registry-storage-pvc
```

Also for reference: [Configuring Image Registry to use OpenShift Data Foundation](#)

## Install OpenShift Virtualization Operator

**STEP 1.** Search on ***Virtualization***, select ***OpenShift Virtualization***. Click on “***Install***”.

The screenshot shows the OperatorHub interface with a search bar containing 'Virtualization'. The search results are displayed in a grid format:

- IBM Fusion Access for SAN** (Certified): A cloud-native storage solution provided by Fusion Access for SAN Team. It is described as a unified operator for KubeVirt.
- KubeVirt HyperConverged Cluster Operator** (Community): A unified operator for KubeVirt.
- Migration Toolkit for Virtualization Operator** (Red Hat): Facilitates migration of VM workloads to OpenShift Virtualization.
- OpenShift Virtualization** (Red Hat): Creates and maintains an OpenShift Virtualization Deployment.

The sidebar on the left lists various operator categories: All Items, AI/Machine Learning, Application Runtime, Big Data, Cloud Provider, Database, Developer Tools, Development Tools, Drivers and plugins, Integration & Delivery, Logging & Tracing, Modernization & Migration, Monitoring, Networking, Observability, OpenShift Optional, Openshift Optional, Security, Storage, Streaming & Messaging, and Other.

**STEP 2.** Click on “*Install*”.

The screenshot shows the OperatorHub interface. On the left, there's a sidebar with a tree view of categories like All Items, AI/Machine Learning, Application Runtime, etc. The main panel displays the details for the "OpenShift Virtualization" operator:

- Icon:** A lightning bolt icon.
- Name:** OpenShift Virtualization
- Version:** 4.19.6 provided by Red Hat
- Install Button:** A blue button labeled "Install".
- Channel:** stable
- Version:** 4.19.6
- Requirements:** Your cluster must be installed on bare metal infrastructure with Red Hat Enterprise Linux CoreOS workers.
- Details:** OpenShift Virtualization extends Red Hat OpenShift Container Platform, allowing you to host and manage virtualized workloads on the same platform as container-based workloads. From the OpenShift Container Platform web console, you can import a VMware virtual machine from vSphere, create new or clone existing VMs, perform live migrations between nodes, and more. You can use OpenShift Virtualization to manage both Linux and Windows VMs.
- Technology:** The technology behind OpenShift Virtualization is developed in the [KubeVirt](#) open source community. The KubeVirt project extends [Kubernetes](#) by adding additional virtualization resource types through [Custom Resource Definitions](#) (CRDs). Administrators can use Custom Resource Definitions to manage [VirtualMachine](#) resources alongside all other resources that Kubernetes provides.
- Capability level:** Basic Install, Seamless Upgrades, Full Lifecycle, Deep Insights (selected), Auto Pilot.
- Source:** Red Hat
- Provider:** Red Hat
- Infrastructure features:** Disconnected, Proxy-aware, Designed for FIPS, Single Node Clusters, Container Storage Interface, Container Network Interface, Configurable TLS ciphers.
- Valid Subscriptions:** OpenShift Kubernetes

**STEP 2.** Select “*Install*”.

## Install Operator



Install your Operator by subscribing to one of the update channels to keep the Operator up to date. The strategy determines either manual or automatic updates.

### Update channel \*

stable

OpenShift Virtualization  
provided by Red Hat

### Version \*

4.19.6

### Provided APIs

**OpenShift Virtualization Deployment** Required

Represents the deployment of OpenShift Virtualization

**HostPathProvisioner Deployment**

Represents the deployment of HostPathProvisioner

### Installation mode \*

All namespaces on the cluster (default)

This mode is not supported by this Operator

A specific namespace on the cluster

Operator will be available in a single Namespace only.

### Installed Namespace \*

Operator recommended Namespace: openshift-cnv

Select a Namespace

Namespace creation

Namespace openshift-cnv does not exist and will be created.

### Update approval \*

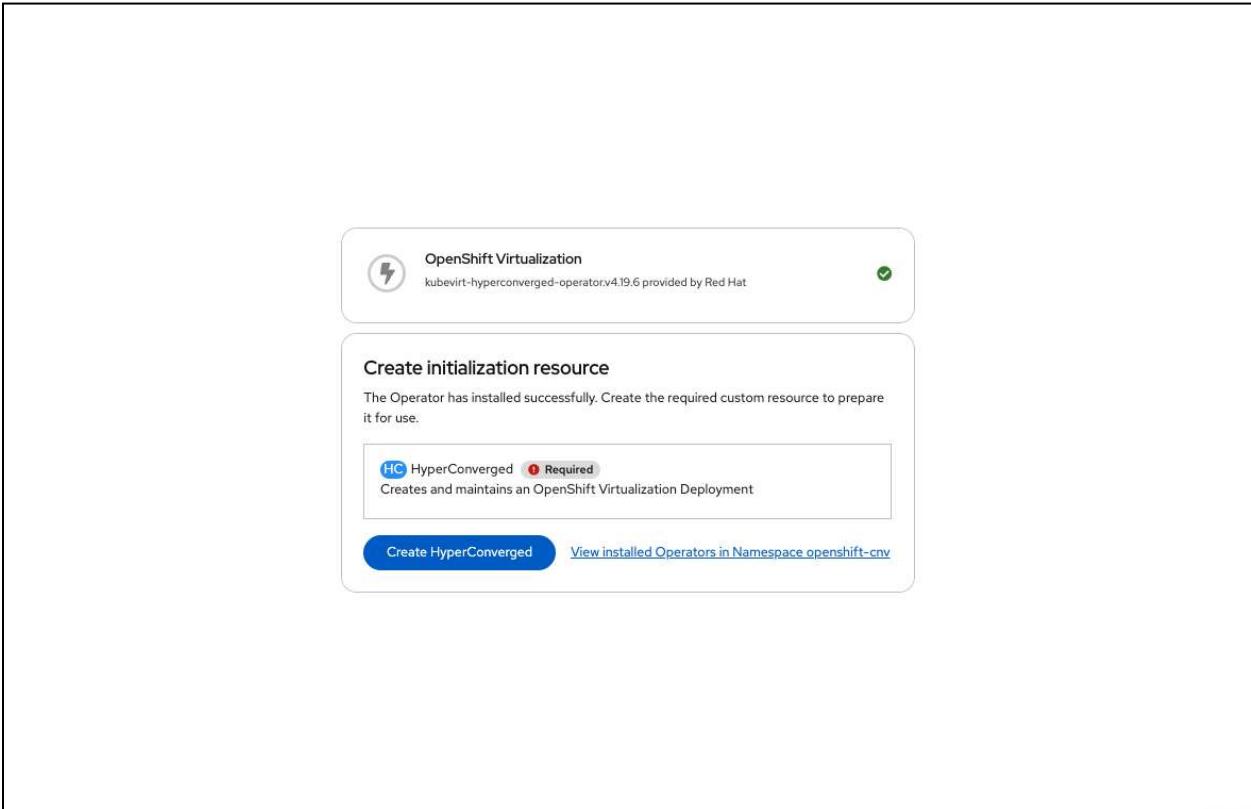
Automatic

Manual

**Install**

**Cancel**

**STEP 3.** Once the operator installation is completed, select “*Create HyperConverged*” to create a deployment of OpenShift Virtualization.



**STEP 4.** Create a deployment using the default settings; scroll down to the bottom of the page, and click on “*Create*”.

Project: openshift-cnv ▾

## Create HyperConverged

Create by completing the form. Default values may be provided by the Operator authors.

Configure via:  Form view  YAML view

**Note:** Some fields may not be represented in this form view. Please select "YAML view" for full control.

 OpenShift Virtualization Deployment  
provided by Red Hat  
Represents the deployment of OpenShift Virtualization

**Name \***  
kubevirt-hyperconverged

**Labels**  
app=frontend

**infra** >  
infra HyperConvergedConfig influences the pod configuration (currently only placement) for all the infra components needed on the virtualization enabled cluster but not necessarily directly on each node running VMs/VMIs.

**workloads** >  
workloads HyperConvergedConfig influences the pod configuration (currently only placement) of components which need to be running on a node where virtualization workloads should be able to run. Changes to Workloads HyperConvergedConfig can be applied only without existing workload.

**storageImport** >  
StorageImport contains configuration for importing containerized data

---

**enableCommonBootImageImport**  
 enableCommonBootImageImport  
Opt-in to automatic delivery/updates of the common data import cron templates.  
There are two sources for the data import cron templates: hard coded list of common templates, and custom (user defined) templates that can be added to the dataimportCronTemplates field. This field only controls the common templates. It is possible to use custom templates by adding them to the dataimportCronTemplates field.

**defaultRuntimeClass**  
DefaultRuntimeClass defines a cluster default for the RuntimeClass to be used for VMIs pods if not set there. Default RuntimeClass can be changed when kubevirt is running, existing VMIs are not impacted till the next restart/live-migration when they are eventually going to consume the new default RuntimeClass.

**virtualMachineOptions** >  
VirtualMachineOptions holds the cluster level information regarding the virtual machine.

**higherWorkloadDensity** >  
HigherWorkloadDensity holds configuration aimed to increase virtual machine density

**defaultCPUModel**  
DefaultCPUModel defines a cluster default for CPU model: default CPU model is set when VMI doesn't have any CPU model.  
When VMI has CPU model set, then VMI's CPU model is preferred.  
When default CPU model is not set and VMI's CPU model is not set too, host-model will be set.  
Default CPU model can be changed when kubevirt is running.

**tektonPipelinesNamespace**  
TektonPipelinesNamespace defines namespace in which example pipelines will be deployed.  
If unset, then the default value is the operator namespace.  
Deprecated: This field is ignored.

**obsoleteCPUs** >  
ObsoleteCPUs allows avoiding scheduling of VMs for obsolete CPU models

**liveMigrationConfig** >

Live migration limits and timeouts are applied so that migration processes do not overwhelm the cluster.

**tektonTasksNamespace**

TektonTasksNamespace defines namespace in which tekton tasks will be deployed. If unset, then the default value is the operator namespace.  
Deprecated: This field is ignored.

**certConfig** >

certConfig holds the rotation policy for internal, self-signed certificates

**tuningPolicy**

Select tuningPolicy ▾

TuningPolicy allows to configure the mode in which the RateLimits of kubevirt are set. If TuningPolicy is not present the default kubevirt values are used. It can be set to `annotation` for fine-tuning the kubevirt queryPerSeconds (qps) and burst values. Qps and burst values are taken from the annotation hco.kubevirt.io/tuningPolicy

**enableApplicationAwareQuota**

enableApplicationAwareQuota  
EnableApplicationAwareQuota if true, enables the Application Aware Quota feature

**mediatedDevicesConfiguration** >

MediatedDevicesConfiguration holds information about MDEV types to be defined on nodes, if available

**scratchSpaceStorageClass**

Override the storage class used for scratch space during transfer operations. The scratch space storage class is determined in the following order:  
value of scratchSpaceStorageClass, if that doesn't exist, use the default storage class, if there is no default storage class, use the storage class of the DataVolume, if no storage class specified, use no storage class for scratch space

**logVerbosityConfig** >

LogVerbosityConfig configures the verbosity level of Kubevirt's different components. The higher the value - the higher the log verbosity.

**commonTemplatesNamespace**

CommonTemplatesNamespace defines namespace in which common templates will be deployed. It overrides the default openshift namespace.

**tlsSecurityProfile** >

TLSecurityProfile specifies the settings for TLS connections to be propagated to all kubevirt-hyperconverged components. If unset, the hyperconverged cluster operator will consume the value set on the APIServer CR on OCP/OKD or Intermediate if on vanilla k8s.  
Note that only Old, Intermediate and Custom profiles are currently supported, and the maximum available MinTLSVersions is VersionTLS12.

**applicationAwareConfig** >

ApplicationAwareConfig set the AAQ configurations

**ksmConfiguration** >

KSMConfiguration holds the information regarding the enabling the KSM in the nodes (if available).

**instancetypeConfig** >

InstancetypeConfig holds the configuration of instance type related functionality within KubeVirt.

**evictionStrategy**

Select evictionStrategy ▾

EvictionStrategy defines at the cluster level if the VirtualMachineInstance should be migrated instead of shut-off in case of a node drain. If the VirtualMachineInstance specific field is set it overrides the cluster level one.

Allowed values:

- "None" no eviction strategy at cluster level.
- "LiveMigrate" migrate the VM on eviction; a not live migratable VM with no specific strategy will block the drain of the node until manually evicted.
- "LiveMigrateIfPossible" migrate the VM on eviction if live migration is possible, otherwise directly evict.
- "External" block the drain, track eviction and notify an external controller.

Defaults to LiveMigrate with multiple worker nodes, None on single worker clusters.

**featureGates**

featureGates is a map of feature gate flags. Setting a flag to `true` will enable the feature. Setting `false` or removing the feature gate, disables the feature.

**localStorageClassName**

Deprecated: LocalStorageClassName the name of the local storage class.

**workloadUpdateStrategy**

WorkloadUpdateStrategy defines at the cluster level how to handle automated workload updates

**deployVmConsoleProxy**

deployVmConsoleProxy

deploy VM console proxy resources in SSP operator

**uninstallStrategy**

BlockUninstallIfWorkloadsExist ▾

UninstallStrategy defines how to proceed on uninstall when workloads (VirtualMachines, DataVolumes) still exist. BlockUninstallIfWorkloadsExist will prevent the CR from being removed when workloads still exist. BlockUninstallIfWorkloadsExist is the safest choice to protect your workloads from accidental data loss, so it's strongly advised. RemoveWorkloads will cause all the workloads to be cascading deleted on uninstallation. WARNING: please notice that RemoveWorkloads will cause your workloads to be deleted as soon as this CR will be, even accidentally, deleted. Please correctly consider the implications of this option before setting it. BlockUninstallIfWorkloadsExist is the default behaviour.

**vddkInitImage**

VDDK Init Image eventually used to import VMs from external providers

Deprecated: please use the Migration Toolkit for Virtualization

**vmStateStorageClass**

VMStateStorageClass is the name of the storage class to use for the PVCs created to preserve VM state, like TPM.

**kubeSecondaryDNSNameServerIP**

KubeSecondaryDNSNameServerIP defines name server IP used by KubeSecondaryDNS

**dataImportCronTemplates**

DataImportCronTemplates holds list of data import cron templates (golden images)

**permittedHostDevices**

PermittedHostDevices holds information about devices allowed for passthrough

**commonBootImageNamespace**

CommonBootImageNamespace override the default namespace of the common boot images, in order to hide them.

If not set, HCO won't set any namespace, letting SSP to use the default. If set, use the namespace to create the DataImportCronTemplates and the common image streams, with this namespace. This field is not set by default.

**CommonInstancetypesDeployment**

CommonInstancetypesDeployment holds the configuration of common-instancetypes deployment within KubeVirt.

**resourceRequirements**

ResourceRequirements describes the resource requirements for the operand workloads.

**filesystemOverhead**

FilesystemOverhead describes the space reserved for overhead when using Filesystem volumes. A value is between 0 and 1, if not defined it is 0.055 (5.5 percent overhead)

**Create**    **Cancel**

## Install NMState Operator

**STEP 1.** In OperatorHub, search on *NMState*, select the *Kubernetes NMState Operator*.

**OperatorHub**

Discover Operators from the Kubernetes community and Red Hat partners, curated by Red Hat. You can purchase commercial software through [Red Hat Marketplace](#). You can install Operators on your clusters to provide optional add-ons and shared services to your developers. After installation, the Operator capabilities will appear in the [Software Catalog](#), providing a self-service experience.

All Items

All Items

x

1 items

**Kubernetes NMState Operator**  
provided by Red Hat, Inc.

Kubernetes NMState is a declarative means of configuring NetworkManager.

## STEP 2. Click on “*Install*”.

The screenshot shows the OperatorHub interface. On the left, there's a sidebar with categories like All Items, AI/Machine Learning, Application Runtime, etc. In the center, a search bar has 'nmstate' typed into it. A card for 'Kubernetes NMState Operator' is displayed, showing its version as 4.19.0-202509240557 provided by Red Hat, Inc. The card includes an 'Install' button, a Channel dropdown set to 'stable', a Version dropdown set to '4.19.0-2025092...', and sections for Capability level, Source, Provider, Infrastructure features, and Valid Subscriptions. The 'Basic Install' option is selected under Capability level.

You are logged in as a [View Profile](#)

Project: openshift-storage ▾

## OperatorHub

Discover Operators from the Kubernetes community and Red Hat partners to provide optional add-ons and shared services to your clusters to provide optional add-ons and shared services to your de

All Items

All Items

Search: nmstate

Kubernetes NMState Operator

4.19.0-202509240557 provided by Red Hat, Inc.

Install

Channel: stable

Version: 4.19.0-2025092...

Capability level:

- Basic Install
- Seamless Upgrades
- Full Lifecycle
- Deep Insights
- Auto Pilot

Source: Red Hat

Provider: Red Hat, Inc.

Infrastructure features:

- Disconnected
- Designed for FIPS

Valid Subscriptions:

- OpenShift Kubernetes Engine
- OpenShift Virtualization Engine
- OpenShift Container Platform
- OpenShift Platform Plus

## STEP 3. Select “*Install*”.

You are logged in as a temporary administrative user. Update the [cluster OAuth configuration](#) to allow others to log in.

OperatorHub > Operator Installation

## Install Operator

Install your Operator by subscribing to one of the update channels to keep the Operator up to date. The strategy determines either manual or automatic updates.

**Update channel \*** ⓘ

stable

**Version \***

4.19.0-202509240557

**Installation mode \***

All namespaces on the cluster (default)  
This mode is not supported by this Operator

A specific namespace on the cluster  
Operator will be available in a single Namespace only.

**Installed Namespace \***

Operator recommended Namespace: **PR openshift-nmstate**

Select a Namespace

**Namespace creation**  
Namespace `openshift-nmstate` does not exist and will be created.

**Provided APIs**

**NMS NMState**  
Represents an NMState deployment.

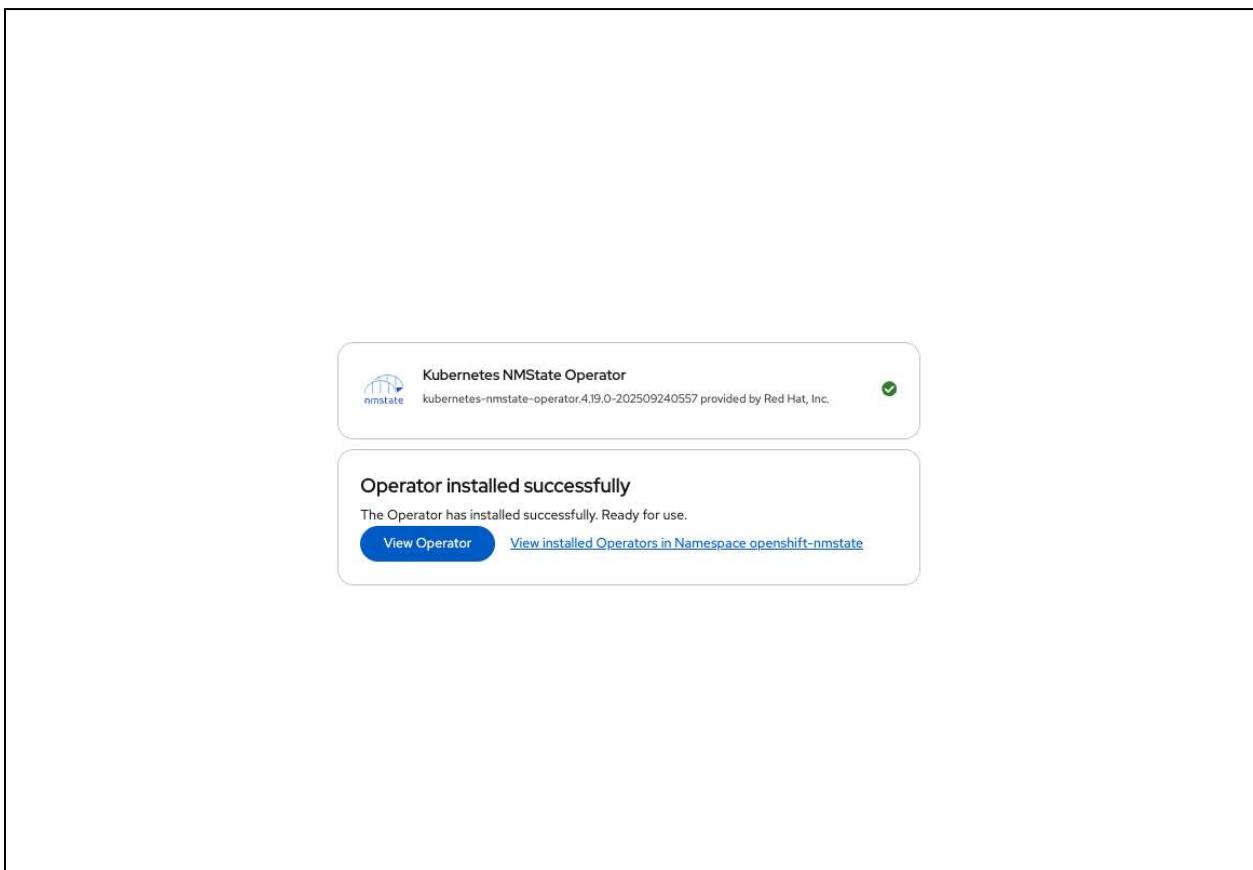
**Update approval \*** ⓘ

Automatic

Manual

**Install** **Cancel**

**STEP 4.** Once the operator installation is completed, select “*View Operator*”.



STEP 5. Select “*Create instance*”.

Project: openshift-nmstate ▾

Installed Operators > Operator details

 Kubernetes NMState Operator  
4.19.0-202509240557 provided by Red Hat, Inc.

Actions ▾

Details YAML Subscription Events NMState

**Provided APIs**

 NMS NMState  
Represents an NMState deployment.  
[Create instance](#)

**Description**  
A Kubernetes Operator to install Kubernetes NMState

**ClusterServiceVersion details**

Name	Status
kubernetes-nmstate-operator.4.19.0-202509240557	 Succeeded
Namespace	Status reason
 openshift-nmstate	install strategy completed with no errors
Labels	Edit
olm.managed=true operatorframework.io/arch.amd64=supported operatorframework.io/arch.arm64=supported operatorframework.io/arch.ppc64le=supported	
Operator Deployments	 nmstate-operator
Operator ServiceAccounts	

**STEP 6.** Select “Create”.



Project: openshift-nmstate ▾

### Create NMState

Create by completing the form. Default values may be provided by the Operator authors.

Configure via:  Form view  YAML view

**Note:** Some fields may not be represented in this form view. Please select "YAML view" for full control.

**Name \***  
nmstate

**Labels**  
app=frontend

**affinity**  
Affinity is an optional affinity selector that will be added to handler DaemonSet manifest.

**infraAffinity**  
InfraAffinity is an optional affinity selector that will be added to webhook, metrics & console-plugin Deployment manifests.

**infraTolerations**  
InfraTolerations is an optional list of tolerations to be added to webhook, metrics & console-plugin Deployment manifests. If InfraTolerations is specified, the webhook, metrics and the console plugin will be able to be scheduled on nodes with corresponding taints

**probeConfiguration**  
ProbeConfiguration is an optional configuration of NMstate probes testing various functionalities. If ProbeConfiguration is specified, the handler will use the config defined here instead of its default values.

**selfSignConfiguration**  
SelfSignConfiguration defines self signed certificate configuration

**tolerations**  
Tolerations is an optional list of tolerations to be added to handler DaemonSet manifest. If Tolerations is specified, the handler daemonset will be also scheduled on nodes with corresponding taints

**Create** **Cancel**

## Create Secondary Network

OpenShift Container Platform 4.18: Virtualization – Networking – [10.9 Connecting a virtual machine to an OVN-Kubernetes secondary network](#)

Create the following **NodeNetworkConfigurationPolicy** and **NetworkAttachmentDefinition**.

**STEP 1.** From the OpenShift console, navigate to **Networking > NodeNetworkConfigurationPolicy**, then click on “**Create NodeNetworkConfigurationPolicy**”. Click on “**Edit YAML**”, and then copy/paste the following YAML into the editor window. When complete, click “**Create**”.

```
apiVersion: nmstate.io/v1
kind: NodeNetworkConfigurationPolicy
metadata:
  name: localnet-network-mapping
spec:
  nodeSelector:
    node-role.kubernetes.io/worker: ''
  desiredState:
    ovn:
```

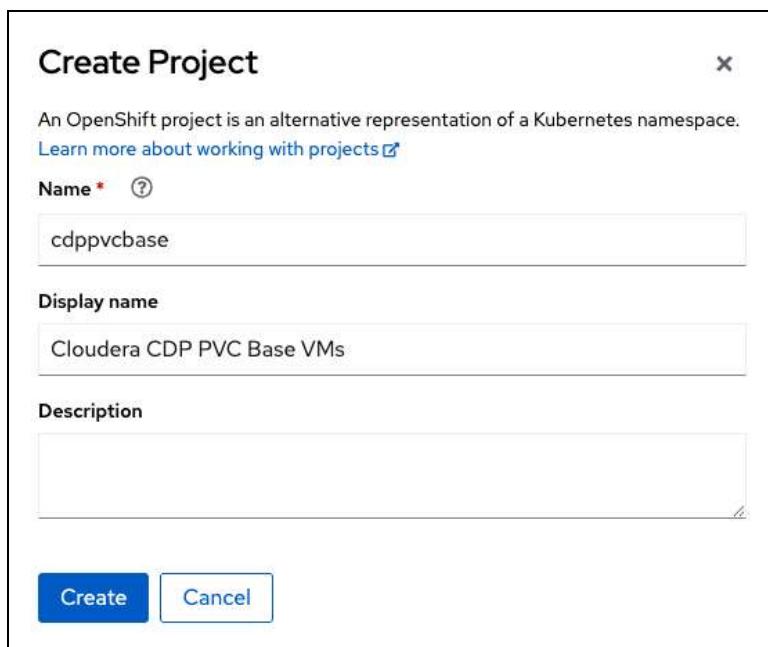
```
bridge-mappings:  
  - localnet: localnet-network  
    bridge: br-ex  
    state: present
```

**STEP 2.** From the OpenShift console, navigate to **Networking > NetworkAttachmentDefinition**, then click on “**Create NetworkAttachmentDefinition**”. Click on “**YAML view**”, and then copy/paste the following YAML into the editor window. When complete, click “**Create**”.

```
apiVersion: k8s.cni.cncf.io/v1  
kind: NetworkAttachmentDefinition  
metadata:  
  name: localnet-network  
  namespace: default  
spec:  
  config: |2  
  {  
    "cniVersion": "0.3.1",  
    "name": "localnet-network",  
    "type": "ovn-k8s-cni-overlay",  
    "topology": "localnet",  
    "netAttachDefName": "default/localnet-network"  
  }
```

## Create Project for CDP PVC Base VMs

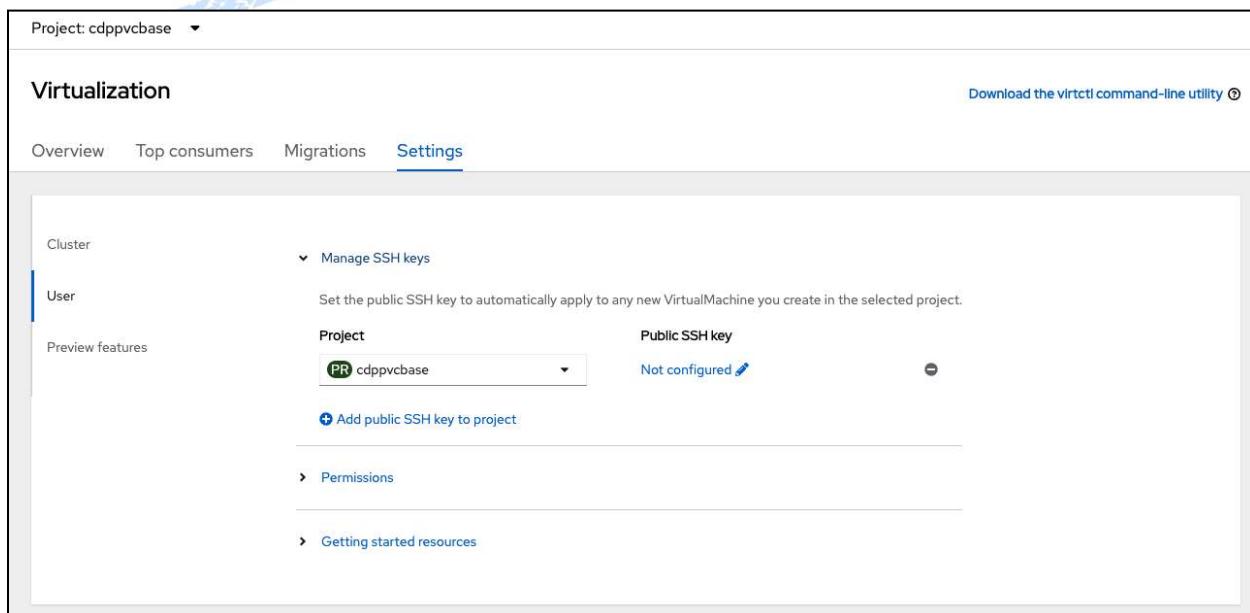
**STEP 1.** Navigate to **Home > Projects** from the OpenShift Console. Select the Create Project button to create the **cdppvcbase** project:



The dialog box has a title "Create Project" and a descriptive message: "An OpenShift project is an alternative representation of a Kubernetes namespace. Learn more about working with projects". It contains three input fields: "Name" (cdppvcbase), "Display name" (Cloudera CDP PVC Base VMs), and "Description". At the bottom are "Create" and "Cancel" buttons.

## Create SSH Keys in Project

**STEP 1.** Navigate to **Virtualization > Overview > Settings > User > Manage SSH Keys**. Select the project **cdppvcbase**, then click on **Not configured** to add the public SSH key to the project.



The screenshot shows the "Virtualization" settings page for the "cdppvcbase" project. Under the "User" section, the "Manage SSH keys" option is expanded. It shows a dropdown for "Project" set to "cdppvcbase" and a status for "Public SSH key" as "Not configured". There is a link to "Add public SSH key to project". Other sections visible include "Cluster", "Preview features", "Permissions", and "Getting started resources".

**STEP 2.** Select **Add new**, and paste the contents of the public key into the page, provide a **secret name**, and click on **Save**.

**Public SSH key**

SSH key is saved in the project as a secret

None  Use existing  Add new

Drag a file here or browse to upload

```
ssh-rsa AAAAB3NzaC1yc2EAAAQABAAQACQC2y5hGR6SdUtCPSIH  
Alh+kF1nnO+9/ihDetHZS0b/WljSAAsSMUvJwHPrEzN0po+ltA8x3j  
wATIDH44bZR7Kzf/Qkkc2RYmtTRksy5qjkvsJYnddPxn9k732DUa  
7jS29mlj7+zlERPOmTgVlqAPt0semJ5SyEtPHbVI3FlmdgJyRsk02I
```

Secret name \*

Automatically apply this key to any new VirtualMachine you create in this project.

## Auto Register RHEL VMs

**STEP 1.** From the OpenShift console, navigate to **Virtualization > Overview > Settings > Cluster > Guest Management > Automatic subscription of new RHEL VirtualMachines**. Select **Monitor and manage subscriptions**. Enter the **Activation key**, **Organization ID** and select **Enable auto updates for RHEL VirtualMachines**.

The screenshot shows the 'Virtualization' settings page in the Red Hat OpenShift web console. At the top, there are tabs for 'Overview', 'Top consumers', 'Migrations', and 'Settings'. The 'Settings' tab is selected. On the left, there's a sidebar with 'Cluster' and 'User' sections. In the main area, there's a summary box for the 'Installed version' (4.18.3), 'Update status' (Up to date), and 'Channel' (stable). Below this, under 'Preview features', there are sections for 'General settings' (selected), 'Guest management', and 'Automatic subscription of new RHEL VirtualMachines'. There are fields for 'Activation key' (cdppvcbase), 'Organization ID' (11009103), and checkboxes for 'Enable auto updates for RHEL VirtualMachines' (checked) and 'Use custom registration server url'. At the bottom, there's a link to 'Enable guest system log access'.

For more information, refer to [Blog: Subscribing RHEL VMs in OpenShift Virtualization](#)

To create an activation key for your Red Hat Account and Org, you can follow the link provided at:

<https://console.redhat.com/settings/connector/activation-keys>

This registration only happens during VM creation via the CloudInit script.

## NOTE: In case the VM does not register with Red Hat Insights, you can use the following command to manually register the VM: rhc connect

Example:

```
# rhc connect --activation-key cdppvcbase --organization 11009103
```

This can also be performed by using the ansible command, in case it isn't applied during VM creation.

```
ansible all -m shell -a "sudo subscription-manager register --org=11009103 --activationkey=cdppvcbase"
```

## Create VMs on OpenShift Virtualization

The following table is provided as a reference for the VMs that will be created for CDP Base:

Hostname	CPU	RAM	Disk	IP Address
cldr-mngr.<domain_name>	32	64 GiB	600 GiB	10.1.49.100/23
cldr-utility.<domain_name>	32	64 GiB	600 GiB	10.1.49.101/23
pvcbase-master[01-03].<domain_name>	32	64 GiB	500 GiB	10.1.49.102-104/23
pvcbase-worker[01-96].<domain_name>	32	64 GiB	500 GiB	10.1.49.105-199/23

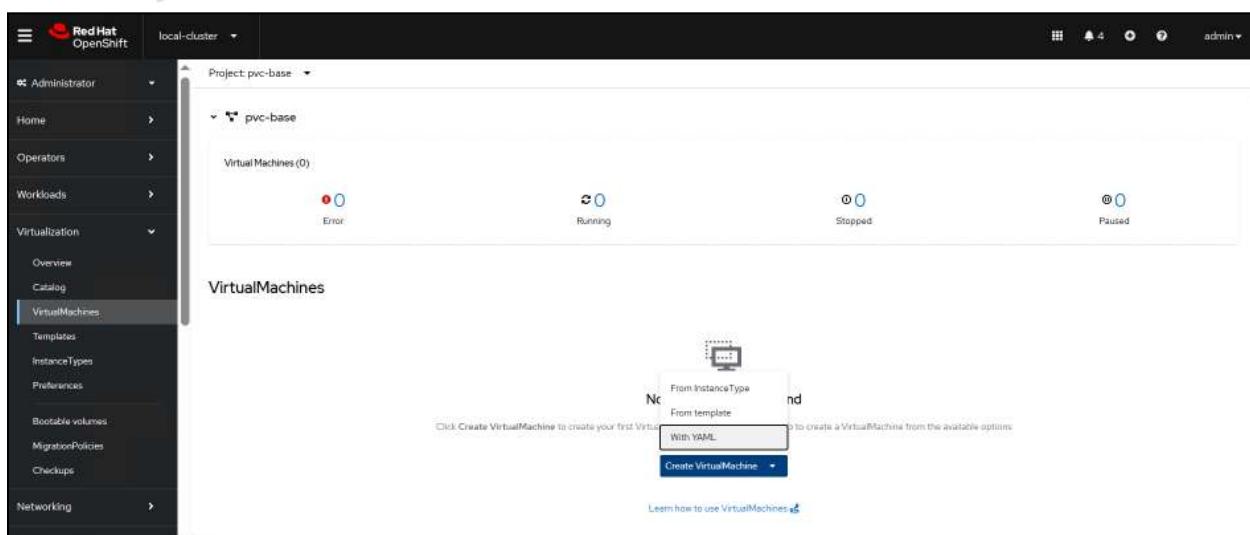
<domain\_name> for the RH scale lab will be: cdp.rdu2.scalelab.redhat.com

The IPA server was created in a server (or a VM) running outside from OpenShift Virtualization:

ipaserver.<domain_name>	16	32 GiB	250 GiB	10.1.49.1/23
-------------------------	----	--------	---------	--------------

**STEP 1.** We will quickly create the VirtualMachines using the YAML definition provided below.

From the OpenShift console, navigate to *Virtualization > VirtualMachines*. Make sure that the “cdppvcbase” Project is shown in the upper left part of the screen. Click on the “Create VirtualMachine” button and select “With YAML”.

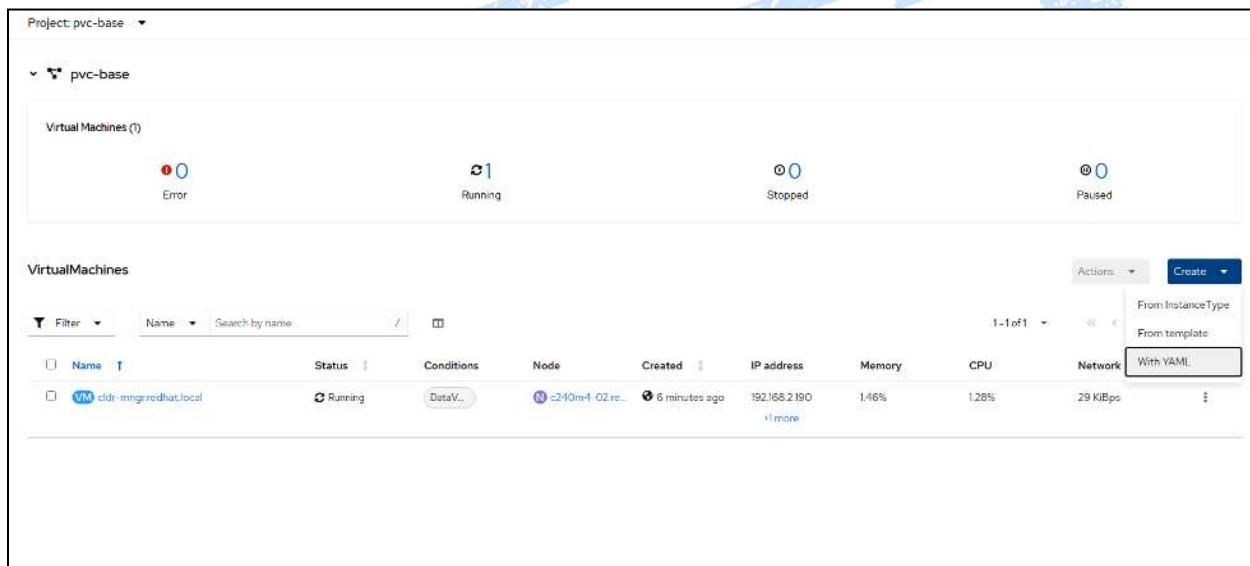


**STEP 2.** From the Create VirtualMachine screen, copy/paste the YAML code provided below into the editor window replacing the code that is initially shown.

**STEP 3.** To create the VM for `cldr-mngr`, no changes are needed to the provided YAML code. Click on “Create”.

**STEP 4.** Next, we will create the other VMs; using the VirtualMachine YAML code for the remaining VMs that is provided at the bottom of this section; after step 8.

**STEP 5.** Navigate back to *Virtualization > VirtualMachines*, and click on the “Create” button on the right side of the screen and then select “With YAML”.



The screenshot shows the Cloudera Manager interface with the following details:

- Project:** pvc-base
- Virtual Machines (0)**: Shows four status categories: Error (0), Running (1), Stopped (0), and Paused (0).
- VirtualMachines**: A table listing one VM:

Name	Status	Conditions	Node	Created	IP address	Memory	CPU	Network
VM cldr-mngr@hat.local	Running	DataV...	c240m4-02.re...	6 minutes ago	192.168.2.190	1.46%	1.28%	29 Kbps
- Create** dropdown menu:
  - From InstanceType
  - From template
  - With YAML** (highlighted)

**STEP 6.** Again, copy/paste the provided YAML code into the editor window, replacing the code that is initially shown.

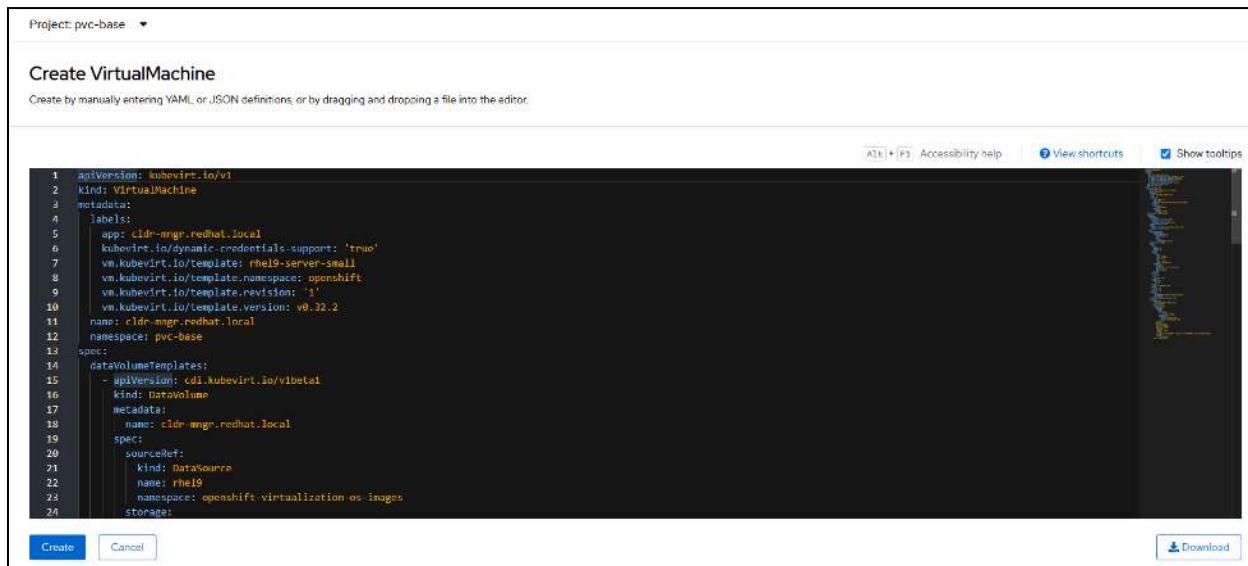
Project: pvc-base ▾

### Create VirtualMachine

Create by manually entering YAML or JSON definitions, or by dragging and dropping a file into the editor.

```
1 apiVersion: kubevirt.io/v1
2 kind: VirtualMachine
3 metadata:
4   labels:
5     app: cldr-mngr.redhat.local
6     kubevirt.io/dynamic-credentials-support: 'true'
7     vn.kubevirt.io/template: rhel9-server-small
8     vn.kubevirt.io/template.namespace: openshift
9     vn.kubevirt.io/template.revision: '1'
10    vn.kubevirt.io/template.version: v0.32.2
11    name: cldr-mngr.redhat.local
12    namespace: pvc-base
13  spec:
14    dataVolumeTemplates:
15      - apiVersion: cd1.kubevirt.io/v1beta1
16        kind: DataVolume
17        metadata:
18          name: cldr-mngr.redhat.local
19        spec:
20          sourceRef:
21            kind: DataSource
22            name: rhel9
23            namespace: openshift-virtualization-os-images
24          storage:
```

**Create** **Cancel** **Download**



**## NOTE:** Use the VirtualMachine YAML code for each of the remaining VMs that is provided at the bottom of this section; after step 8.

**STEP 7.** When any additional/optional changes have been made to the configuration parameters, click on “Create”.

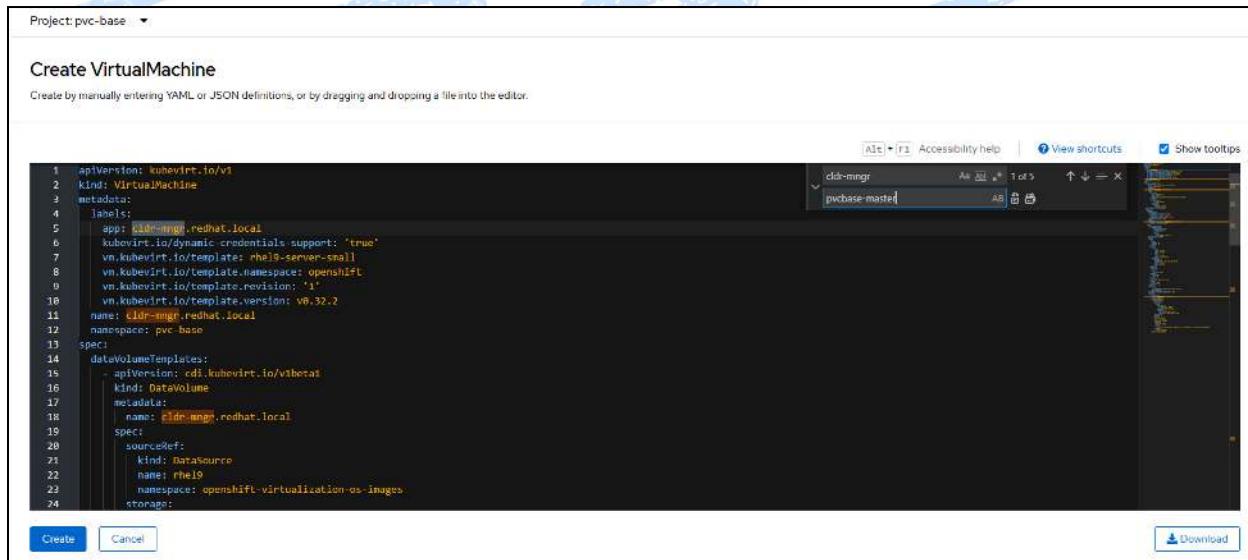
Project: pvc-base ▾

### Create VirtualMachine

Create by manually entering YAML or JSON definitions, or by dragging and dropping a file into the editor.

```
1 apiVersion: kubevirt.io/v1
2 kind: VirtualMachine
3 metadata:
4   labels:
5     app: cldr-mngr.redhat.local
6     kubevirt.io/dynamic-credentials-support: 'true'
7     vn.kubevirt.io/template: rhel9-server-small
8     vn.kubevirt.io/template.namespace: openshift
9     vn.kubevirt.io/template.revision: '1'
10    vn.kubevirt.io/template.version: v0.32.2
11    name: cldr-mngr.redhat.local
12    namespace: pvc-base
13  spec:
14    dataVolumeTemplates:
15      - apiVersion: cd1.kubevirt.io/v1beta1
16        kind: DataVolume
17        metadata:
18          name: cldr-mngr.redhat.local
19        spec:
20          sourceRef:
21            kind: DataSource
22            name: rhel9
23            namespace: openshift-virtualization-os-images
24          storage:
```

**Create** **Cancel** **Download**



**STEP 8.** Repeat the last three steps until all of the VMs have been created.

The following is the cloud-init section that is included in each of the VM YAML definitions provided.

## cloud-init section of the VirtualMachine YAML

```
- cloudInitNoCloud:
    networkData: |
        eternets:
            eth0:
                addresses:
                    - 10.1.49.100/23      # Set IP v4 address
                gateway4: 10.1.49.254   # Set default gateway
                nameservers:
                    search: [cdp.rdu2.scalelab.redhat.com] # Set DNS search domain
                    addresses: [10.1.49.1] # Set DNS server address
            version: 2
    userData: |-
        #cloud-config
        user: cloud-user          # Create user account
        password: r3dh4t!          # Set password for user account
        chpasswd: { expire: False } # Disable password expiration
        ssh_pwauth: True          # Enable password login for root over ssh
        hostname: cldr-mngr.cdp.rdu2.scalelab.redhat.com # set hostname for VM
        packages:
            - dnf-automatic
        runcmd:
            - subscription-manager register --org=11009103 --activationkey=cdpbbase
            - subscription-manager release --set=9.5           # Set RHEL minor version
            - systemctl enable --now dnf-automatic-install.timer
            - |
              #!/bin/bash
              # Add entries for sysctl
              echo "net.ipv6.conf.all.disable_ipv6 = 1" >> /etc/sysctl.conf
              echo "net.ipv6.conf.default.disable_ipv6 = 1" >> /etc/sysctl.conf
              echo "net.ipv6.conf.lo.disable_ipv6 = 0" >> /etc/sysctl.conf
              echo "vm.overcommit_memory=0" >> /etc/sysctl.conf
              echo "vm.swappiness=1" >> /etc/sysctl.conf
              # Enable sysctl
              sysctl -p
```

The VirtualMachine YAML code for all 7 of the VMs are provided below. Refer to the section [OPTIONAL Method to create VMs](#) for the process to create the VM configuration and its associated YAML code.

## cldr-mngr VirtualMachine YAML

```
apiVersion: kubevirt.io/v1
kind: VirtualMachine
metadata:
  annotations:
    kubemacpool.io/transaction-timestamp: '2025-10-27T20:30:50.413104701Z'
    kubevirt.io/latest-observed-api-version: v1
    kubevirt.io/storage-observed-api-version: v1
    vm.kubevirt.io/validations: |
      [
        {
          ...
```

```
        "name": "minimal-required-memory",
        "path": "jsonpath:::spec.domain.memory.guest",
        "rule": "integer",
        "message": "This VM requires more memory.",
        "min": 1610612736
    }
]
resourceVersion: '8837376'
name: cldr-mngr
uid: 38f278e3-7b56-4f65-8586-c4e16adbc40e
creationTimestamp: '2025-10-27T20:30:50Z'
generation: 1
managedFields:
  - apiVersion: kubevirt.io/v1
    fieldsType: FieldsV1
    fieldsV1:
      'f:metadata':
        'f:annotations':
          .: {}
        'f:kubemacpool.io/transaction-timestamp': {}
        'f:vm.kubevirt.io/validations': {}
      'f:labels':
        .: {}
      'f:app': {}
      'f:kubevirt.io/dynamic-credentials-support': {}
      'f:vm.kubevirt.io/template': {}
      'f:vm.kubevirt.io/template.namespace': {}
      'f:vm.kubevirt.io/template.revision': {}
      'f:vm.kubevirt.io/template.version': {}
    'f:spec':
      .: {}
      'f:dataVolumeTemplates': {}
      'f:runStrategy': {}
      'f:template':
        .: {}
      'f:metadata':
        .: {}
        'f:annotations':
          .: {}
          'f:vm.kubevirt.io/flavor': {}
          'f:vm.kubevirt.io/os': {}
          'f:vm.kubevirt.io/workload': {}
        'f:creationTimestamp': {}
      'f:labels':
        .: {}
        'f:affinity_group_1': {}
        'f:kubevirt.io/domain': {}
        'f:kubevirt.io/size': {}
    'f:spec':
      'f:accessCredentials': {}
      'f:volumes': {}

```

```

        'f:hostname': {}
        'f:architecture': {}
        .: {}
        'f:terminationGracePeriodSeconds': {}
        'f:domain':
            .: {}
            'f:cpu':
                .: {}
                'f:cores': {}
                'f:sockets': {}
                'f:threads': {}
            'f:devices':
                .: {}
                'f:disks': {}
                'f:interfaces': {}
                'f:networkInterfaceMultiqueue': {}
                'f:rng': {}
            'f:features':
                .: {}
                'f:acpi': {}
                'f:smm':
                    .: {}
                    'f:enabled': {}
        'f:firmware':
            .: {}
            'f:bootloader':
                .: {}
                'f:efi': {}
        'f:machine':
            .: {}
            'f:type': {}
        'f:memory':
            .: {}
            'f:guest': {}
        'f:resources': {}
    'f:networks': {}
    'f:affinity':
        .: {}
        'f:podAntiAffinity':
            .: {}
            'f:requiredDuringSchedulingIgnoredDuringExecution': {}

manager: OpenAPI-Generator
operation: Update
time: '2025-10-27T20:30:50Z'
- apiVersion: kubevirt.io/v1
  fieldsType: FieldsV1
  fieldsV1:
    'f:metadata':
      'f:annotations':
        'f:kubevirt.io/latest-observed-api-version': {}
        'f:kubevirt.io/storage-observed-api-version': {}

```

```

        'f:finalizers':
          .: {}
          'v:"kubevirt.io/virtualMachineControllerFinalize)": {}
manager: virt-controller
operation: Update
time: '2025-10-27T20:30:50Z'
- apiVersion: kubevirt.io/v1
  fieldsType: FieldsV1
  fieldsV1:
    'f:status':
      'f:printableStatus': {}
      'f:runStrategy': {}
      'f:conditions': {}
      .: {}
      'f:ready': {}
      'f:volumeSnapshotStatuses': {}
      'f:observedGeneration': {}
      'f:created': {}
      'f:desiredGeneration': {}
    manager: virt-controller
    operation: Update
    subresource: status
    time: '2025-10-27T20:31:39Z'
  namespace: cdppvcbase
  finalizers:
    - kubevirt.io/virtualMachineControllerFinalize
  labels:
    app: cldr-mngr
    kubevirt.io/dynamic-credentials-support: 'true'
    vm.kubevirt.io/template: rhel9-server-large
    vm.kubevirt.io/template.namespace: openshift
    vm.kubevirt.io/template.revision: '1'
    vm.kubevirt.io/template.version: v0.34.0
spec:
  dataVolumeTemplates:
    - apiVersion: cdi.kubevirt.io/v1beta1
      kind: DataVolume
      metadata:
        creationTimestamp: null
        name: cldr-mngr-dv
      spec:
        source:
          pvc:
            name: rhel9.5
            namespace: cdppvcbase
        storage:
          resources:
            requests:
              storage: 600Gi
            storageClassName: ocs-storagecluster-ceph-replica2-rbd
        runStrategy: RerunOnFailure

```

```
template:
  metadata:
    annotations:
      vm.kubevirt.io/flavor: large
      vm.kubevirt.io/os: rhel9
      vm.kubevirt.io/workload: server
    creationTimestamp: null
    labels:
      affinity_group_1: ''
      kubevirt.io/domain: cldr-mngr
      kubevirt.io/size: large
  spec:
    accessCredentials:
      - sshPublicKey:
          propagationMethod:
            noCloud: {}
        source:
          secret:
            secretName: cdp-ssh
    affinity:
      podAntiAffinity:
        requiredDuringSchedulingIgnoredDuringExecution:
          - labelSelector:
              matchExpressions:
                - key: affinity_group_1
                  operator: Exists
            topologyKey: kubernetes.io/hostname
    architecture: amd64
    domain:
      cpu:
        cores: 1
        sockets: 8
        threads: 1
      devices:
        disks:
          - disk:
              bus: virtio
              name: rootdisk
          - disk:
              bus: virtio
              name: cloudinitdisk
      interfaces:
        - bridge: {}
          macAddress: '02:00:e3:00:02:3f'
          model: virtio
          name: default
          state: up
        networkInterfaceMultiqueue: true
        rng: {}
      features:
        acpi: {}
#
```

```

    smm:
      enabled: true
    firmware:
      bootloader:
        efi: {}
    machine:
      type: pc-q35-rhel9.6.0
    memory:
      guest: 32Gi
    resources: {}
  hostname: cldr-mngr
  networks:
    - multus:
        networkName: default/localnet-network
        name: default
  terminationGracePeriodSeconds: 180
  volumes:
    - dataVolume:
        name: cldr-mngr-dv
        name: rootdisk
    - cloudInitNoCloud:
        networkData: |
          ethernets:
            eth0:
              addresses:
                - 10.1.49.100/23
            gateway4: 10.1.49.254
            nameservers:
              search: [cdp.rdu2.scalelab.redhat.com]
              addresses: [10.1.49.1]
            version: 2
        userData: |-
          #cloud-config
          user: cloud-user
          password: r3dh4t!
          chpasswd: { expire: False }
          ssh_pwauth: True
        hostname: cldr-mngr.cdp.rdu2.scalelab.redhat.com
        packages:
          - dnf-automatic
        runcmd:
          - subscription-manager register --org=11009103
--activationkey=cdpbase
          - subscription-manager release --set=9.5
          - systemctl enable --now dnf-automatic-install.timer
          - |
            #!/bin/bash
            # Add entries for sysctl
            echo "net.ipv6.conf.all.disable_ipv6 = 1" >> /etc/sysctl.conf
            echo "net.ipv6.conf.default.disable_ipv6 = 1" >> /etc/sysctl.conf
            echo "net.ipv6.conf.lo.disable_ipv6 = 0" >> /etc/sysctl.conf

```

```

echo "vm.overcommit_memory=0" >> /etc/sysctl.conf
echo "vm.swappiness=1" >> /etc/sysctl.conf
# Enable sysctl
sysctl -p
name: cloudinitdisk

```

## cldr-utility VirtualMachine YAML

```

apiVersion: kubevirt.io/v1
kind: VirtualMachine
metadata:
  annotations:
    kubemacpool.io/transaction-timestamp: '2025-10-27T20:21:32.838262833Z'
    kubevirt.io/latest-observed-api-version: v1
    kubevirt.io/storage-observed-api-version: v1
    vm.kubevirt.io/validations: |
      [
        {
          "name": "minimal-required-memory",
          "path": "jsonpath:::spec.domain.memory.guest",
          "rule": "integer",
          "message": "This VM requires more memory.",
          "min": 1610612736
        }
      ]
resourceVersion: '8817926'
name: cldr-utility
uid: 610bbc39-d416-4c0c-8d53-a06cb55c459f
creationTimestamp: '2025-10-27T20:21:32Z'
generation: 1
managedFields:
  - apiVersion: kubevirt.io/v1
    fieldsType: FieldsV1
    fieldsV1:
      'f:metadata':
        'f:annotations':
          .: {}
        'f:kubemacpool.io/transaction-timestamp': {}
        'f:vm.kubevirt.io/validations': {}
      'f:labels':
        .: {}
      'f:app': {}
      'f:kubevirt.io/dynamic-credentials-support': {}
      'f:vm.kubevirt.io/template': {}
      'f:vm.kubevirt.io/template.namespace': {}
      'f:vm.kubevirt.io/template.revision': {}
      'f:vm.kubevirt.io/template.version': {}
    f:spec':

```

```
.: {}
'f:dataVolumeTemplates': {}
'f:runStrategy': {}
'f:template':
  .: {}
'f:metadata':
  .: {}
'f:annotations':
  .: {}
  'f:vm.kubevirt.io/flavor': {}
  'f:vm.kubevirt.io/os': {}
  'f:vm.kubevirt.io/workload': {}
'f:creationTimestamp': {}
'f:labels':
  .: {}
  'f:affinity_group_1': {}
  'f:kubevirt.io/domain': {}
  'f:kubevirt.io/size': {}
'f:spec':
  'f:accessCredentials': {}
  'f:volumes': {}
  'f:hostname': {}
  'f:architecture': {}
  .: {}
'f:terminationGracePeriodSeconds': {}
'f:domain':
  .: {}
  'f:cpu':
    .: {}
    'f:cores': {}
    'f:sockets': {}
    'f:threads': {}
  'f:devices':
    .: {}
    'f:disks': {}
    'f:interfaces': {}
    'f:networkInterfaceMultiqueue': {}
    'f:rng': {}
  'f:features':
    .: {}
    'f:acpi': {}
    'f:smm':
      .: {}
      'f:enabled': {}
  'f:firmware':
    .: {}
    'f:bootloader':
      .: {}
      'f:efi': {}
  'f:machine':
    .: {}
```

```
        'f:type': {}
        'f:memory':
          .: {}
        'f:guest': {}
        'f:resources': {}
        'f:networks': {}
        'f:affinity':
          .: {}
        'f:podAntiAffinity':
          .: {}
        'f:requiredDuringSchedulingIgnoredDuringExecution': {}

manager: OpenAPI-Generator
operation: Update
time: '2025-10-27T20:21:32Z'
- apiVersion: kubevirt.io/v1
  fieldsType: FieldsV1
  fieldsV1:
    'f:metadata':
      'f:annotations':
        'f:kubevirt.io/latest-observed-api-version': {}
        'f:kubevirt.io/storage-observed-api-version': {}
    'f:finalizers':
      .: {}
      'v:"kubevirt.io/virtualMachineControllerFinalize)": {}
manager: virt-controller
operation: Update
time: '2025-10-27T20:21:32Z'
- apiVersion: kubevirt.io/v1
  fieldsType: FieldsV1
  fieldsV1:
    'f:status':
      'f:printableStatus': {}
      'f:runStrategy': {}
      'f:conditions':
        .: {}
      'f:ready': {}
      'f:volumeSnapshotStatuses': {}
      'f:observedGeneration': {}
      'f:created': {}
      'f:desiredGeneration': {}
    manager: virt-controller
    operation: Update
    subresource: status
    time: '2025-10-27T20:22:14Z'
  namespace: cdppvcbase
  finalizers:
    - kubevirt.io/virtualMachineControllerFinalize
  labels:
    app: cldr-utility
    kubevirt.io/dynamic-credentials-support: 'true'
    vm.kubevirt.io/template: rhel9-server-large
```

```

    vm.kubevirt.io/template.namespace: openshift
    vm.kubevirt.io/template.revision: '1'
    vm.kubevirt.io/template.version: v0.34.0
spec:
  dataVolumeTemplates:
    - apiVersion: cdi.kubevirt.io/v1beta1
      kind: DataVolume
      metadata:
        creationTimestamp: null
        name: cldr-utility-dv
      spec:
        source:
          pvc:
            name: rhel9.5
            namespace: cdppvcbase
        storage:
          resources:
            requests:
              storage: 600Gi
            storageClassName: ocs-storagecluster-ceph-replica2-rbd
  runStrategy: RerunOnFailure
  template:
    metadata:
      annotations:
        vm.kubevirt.io/flavor: large
        vm.kubevirt.io/os: rhel9
        vm.kubevirt.io/workload: server
      creationTimestamp: null
      labels:
        affinity_group_1: ''
        kubevirt.io/domain: cldr-utility
        kubevirt.io/size: large
    spec:
      accessCredentials:
        - sshPublicKey:
            propagationMethod:
              noCloud: {}
            source:
              secret:
                secretName: cdp-ssh
      affinity:
        podAntiAffinity:
          requiredDuringSchedulingIgnoredDuringExecution:
            - labelSelector:
                matchExpressions:
                  - key: affinity_group_1
                    operator: Exists
            topologyKey: kubernetes.io/hostname
      architecture: amd64
      domain:
        cpu:

```

```
cores: 1
sockets: 8
threads: 1
devices:
  disks:
    - disk:
        bus: virtio
        name: rootdisk
    - disk:
        bus: virtio
        name: cloudinitdisk
interfaces:
  - bridge: {}
    macAddress: '02:00:e3:00:02:29'
    model: virtio
    name: default
    state: up
    networkInterfaceMultiqueue: true
    rng: {}
features:
  acpi: {}
  smm:
    enabled: true
firmware:
  bootloader:
    efi: {}
machine:
  type: pc-q35-rhel9.6.0
memory:
  guest: 32Gi
  resources: {}
hostname: cldr-utility
networks:
  - multus:
      networkName: default/localnet-network
      name: default
terminationGracePeriodSeconds: 180
volumes:
  - dataVolume:
      name: cldr-utility-dv
      name: rootdisk
  - cloudInitNoCloud:
      networkData: |
        eternets:
          eth0:
            addresses:
              - 10.1.49.101/23
            gateway4: 10.1.49.254
            nameservers:
              search: [cdp.rdu2.scalelab.redhat.com]
              addresses: [10.1.49.1]
```

```

version: 2
userData: |-
  #cloud-config
  user: cloud-user
  password: r3dh4t!
  chpasswd: { expire: False }
  ssh_pwauth: True
  hostname: cldr-utility.cdp.rdu2.scalelab.redhat.com
  packages:
    - dnf-automatic
  runcmd:
    - subscription-manager register --org=11009103
--activationkey=cdpbase
    - subscription-manager release --set=9.5
    - systemctl enable --now dnf-automatic-install.timer
    - |
      #!/bin/bash
      # Add entries for sysctl
      echo "net.ipv6.conf.all.disable_ipv6 = 1" >> /etc/sysctl.conf
      echo "net.ipv6.conf.default.disable_ipv6 = 1" >> /etc/sysctl.conf
      echo "net.ipv6.conf.lo.disable_ipv6 = 0" >> /etc/sysctl.conf
      echo "vm.overcommit_memory=0" >> /etc/sysctl.conf
      echo "vm.swappiness=1" >> /etc/sysctl.conf
      # Enable sysctl
      sysctl -p
name: cloudinitdisk

```

## pvcbase-masterXX VirtualMachine YAML

```

apiVersion: kubevirt.io/v1
kind: VirtualMachine
metadata:
  name: pvcbase-master01
  generation: 1
  namespace: cdppvcbase
  finalizers:
    - kubevirt.io/virtualMachineControllerFinalize
  labels:
    app: pvcbase-master01
    kubevirt.io/dynamic-credentials-support: 'true'
    vm.kubevirt.io/template: rhel9-server-large
    vm.kubevirt.io/template.namespace: openshift
    vm.kubevirt.io/template.revision: '1'
    vm.kubevirt.io/template.version: v0.34.0
spec:
  dataVolumeTemplates:
    - apiVersion: cdi.kubevirt.io/v1beta1
      kind: DataVolume
      metadata:
        creationTimestamp: null

```

```
        name: pvcbase-master01-dv
      spec:
        source:
          pvc:
            name: rhel9.5
            namespace: cdppvcbase
        storage:
          resources:
            requests:
              storage: 500Gi
          storageClassName: ocs-storagecluster-ceph-replica2-rbd
        runStrategy: RerunOnFailure
      template:
        metadata:
          annotations:
            vm.kubevirt.io/flavor: large
            vm.kubevirt.io/os: rhel9
            vm.kubevirt.io/workload: server
        creationTimestamp: null
        labels:
          affinity_group_1: ''
          kubevirt.io/domain: pvcbase-master01
          kubevirt.io/size: large
      spec:
        accessCredentials:
          - sshPublicKey:
              propagationMethod:
                noCloud: {}
            source:
              secret:
                secretName: cdp-ssh
        affinity:
          podAntiAffinity:
            requiredDuringSchedulingIgnoredDuringExecution:
              - labelSelector:
                  matchExpressions:
                    - key: affinity_group_1
                      operator: Exists
            topologyKey: kubernetes.io/hostname
        architecture: amd64
        domain:
          cpu:
            cores: 1
            sockets: 8
            threads: 1
          devices:
            disks:
              - disk:
                  bus: virtio
                  name: rootdisk
              - disk:
```

```
        bus: virtio
        name: cloudfinitdisk
interfaces:
    - bridge: {}
      macAddress: '02:00:e3:00:02:a1'
      model: virtio
      name: default
      state: up
      networkInterfaceMultiqueue: true
      rng: {}
features:
    acpi: {}
    smm:
        enabled: true
firmware:
    bootloader:
        efi: {}
machine:
    type: pc-q35-rhel9.6.0
memory:
    guest: 32Gi
    resources: {}
hostname: pvcbase-master01
networks:
    - multus:
        networkName: default/localnet-network
        name: default
terminationGracePeriodSeconds: 180
volumes:
    - dataVolume:
        name: pvcbase-master01-dv
        name: rootdisk
    - cloudInitNoCloud:
        networkData: |
            eternets:
                eth0:
                    addresses:
                        - 10.1.49.102/23
                    gateway4: 10.1.49.254
                    nameservers:
                        search: [cdp.rdu2.scalelab.redhat.com]
                        addresses: [10.1.49.1]
        version: 2
userData: |-
    #cloud-config
    user: cloud-user
    password: r3dh4t!
    chpasswd: { expire: False }
    ssh_pwauth: True
    hostname: pvcbase-master01.cdp.rdu2.scalelab.redhat.com
    packages:
```

```

        - dnf-automatic
    runcmd:
        - subscription-manager register --org=11009103
--activationkey=cdpbase
        - subscription-manager release --set=9.5
        - systemctl enable --now dnf-automatic-install.timer
        - |
          #!/bin/bash
# Add entries for sysctl
echo "net.ipv6.conf.all.disable_ipv6 = 1" >> /etc/sysctl.conf
echo "net.ipv6.conf.default.disable_ipv6 = 1" >> /etc/sysctl.conf
echo "net.ipv6.conf.lo.disable_ipv6 = 0" >> /etc/sysctl.conf
echo "vm.overcommit_memory=0" >> /etc/sysctl.conf
echo "vm.swappiness=1" >> /etc/sysctl.conf
# Enable sysctl
sysctl -p
name: cloudinitdisk

```

## pvcbase-workerXX VirtualMachine YAML

```

apiVersion: kubevirt.io/v1
kind: VirtualMachine
metadata:
  name: pvcbase-worker01
  labels:
    app: pvcbase-worker01
    kubevirt.io/dynamic-credentials-support: 'true'
    vm.kubevirt.io/template: rhel9-server-large
    vm.kubevirt.io/template.namespace: openshift
    vm.kubevirt.io/template.revision: '1'
    vm.kubevirt.io/template.version: v0.34.0
spec:
  dataVolumeTemplates:
    - apiVersion: cdi.kubevirt.io/v1beta1
      kind: DataVolume
      metadata:
        creationTimestamp: null
        name: pvcbase-worker01-dv
      spec:
        source:
          pvc:
            name: rhel9.5
            namespace: cdppvcbase
      storage:
        resources:
          requests:
            storage: 500Gi
        storageClassName: ocs-storagecluster-ceph-replica2-rbd
  runStrategy: RerunOnFailure

```

```
template:
  metadata:
    annotations:
      vm.kubevirt.io/flavor: large
      vm.kubevirt.io/os: rhel9
      vm.kubevirt.io/workload: server
    creationTimestamp: null
    labels:
      affinity_group_1: ''
      kubevirt.io/domain: pvcbase-worker01
      kubevirt.io/size: large
  spec:
    accessCredentials:
      - sshPublicKey:
          propagationMethod:
            noCloud: {}
        source:
          secret:
            secretName: cdp-ssh
    affinity:
      podAntiAffinity:
        requiredDuringSchedulingIgnoredDuringExecution:
          - labelSelector:
              matchExpressions:
                - key: affinity_group_1
                  operator: Exists
            topologyKey: kubernetes.io/hostname
    architecture: amd64
    domain:
      cpu:
        cores: 1
        sockets: 8
        threads: 1
      devices:
        disks:
          - disk:
              bus: virtio
              name: rootdisk
          - disk:
              bus: virtio
              name: cloudinitdisk
      interfaces:
        - bridge: {}
          macAddress: '02:00:e3:00:02:42'
          model: virtio
          name: default
          state: up
      networkInterfaceMultiqueue: true
      rng: {}
    features:
      acpi: {}
```

```

    smm:
      enabled: true
    firmware:
      bootloader:
        efi: {}
    machine:
      type: pc-q35-rhel9.6.0
    memory:
      guest: 32Gi
    resources: {}
  hostname: pvcbase-worker01
  networks:
    - multus:
        networkName: default/localnet-network
        name: default
  terminationGracePeriodSeconds: 180
  volumes:
    - dataVolume:
        name: pvcbase-worker01-dv
        name: rootdisk
    - cloudInitNoCloud:
        networkData: |
          ethernets:
            eth0:
              addresses:
                - 10.1.49.105/23
            gateway4: 10.1.49.254
            nameservers:
              search: [cdp.rdu2.scalelab.redhat.com]
              addresses: [10.1.49.1]
            version: 2
        userData: |-
          #cloud-config
          user: cloud-user
          password: r3dh4t!
          chpasswd: { expire: False }
          ssh_pwauth: True
        hostname: pvcbase-worker01.cdp.rdu2.scalelab.redhat.com
      packages:
        - dnf-automatic
      runcmd:
        - subscription-manager register --org=11009103
--activationkey=cdpbase
      - subscription-manager release --set=9.5
      - systemctl enable --now dnf-automatic-install.timer
      - |
        #!/bin/bash
        # Add entries for sysctl
        echo "net.ipv6.conf.all.disable_ipv6 = 1" >> /etc/sysctl.conf
        echo "net.ipv6.conf.default.disable_ipv6 = 1" >> /etc/sysctl.conf
        echo "net.ipv6.conf.lo.disable_ipv6 = 0" >> /etc/sysctl.conf

```

```
echo "vm.overcommit_memory=0" >> /etc/sysctl.conf
echo "vm.swappiness=1" >> /etc/sysctl.conf
# Enable sysctl
sysctl -p
name: cloudinitdisk
```

## OPTIONAL Method to create VMs

If needed, please refer to the optional method to create VMs from the VM Template Catalog:  
[OPTIONAL Method to create VMs](#)

This method can be used to create the VirtualMachine YAML definitions.

**DRAFT**

## CONFIGURE THE LDAP IDENTITY PROVIDER WITH OPENSOURCE:

##### If needed, you can access the IPA Admin console:  
<https://idm.redhat.local/ipa/ui/>

From the WebUI for IPAServer Administration, enter the same admin credentials used for CLI authentication: (i.e. *admin/redhat123*)

```
[root@idm ~]# vi /etc/openldap/ldap.conf
```

```
# File modified by ipa-client-install
. .
# When no CA certificates are specified the Shared System Certificates
# are in use. In order to have these available along with the ones specified
# by TLS_CACERTDIR one has to include them explicitly:
#TLS_CACERT /etc/pki/tls/cert.pem

# System-wide Crypto Policies provide up to date cipher suite which should
# be used unless one needs a finer grinded selection of ciphers. Hence, the
# PROFILE=SYSTEM value represents the default behavior which is in place
# when no explicit setting is used. (see openssl-ciphers(1) for more info)
#TLS_CIPHER_SUITE PROFILE=SYSTEM

# Turning this off breaks GSSAPI used with krb5 when rdns = false
SASL_NOCANON      on

URI ldaps://ipaserver.cdp.rdu2.scalelab.redhat.com
BASE dc=cdp,dc=rdu2,dc=scalelab,dc=redhat,dc=com
SASL_MECH GSSAPI
```

If any changes are made to the IPA/IdM configuration, use the following command to restart the IPA server:

```
[root@idm ~]# ipactl restart
```

**STEP 1.** Create a secret with the Identity Management (IdM) admin user password. This secret will be used for the LDAP identity provider:

```
[root@bastion ]# oc create secret generic ldap-secret -n openshift-config \
--from-literal=bindPassword=redhat123
```

**STEP 2.** TLS communication needs certificate authority validation, in this case, the CA is obtained from IdM.

The certificate is stored on the ipaserver in the file: `/etc/ipa/ca.crt`

```
## Even when IdM is configured not to create/deploy a Certificate Authority,
but instead use certificates provided by an external CA, the installer still
stores the CA certificate in the file: /etc/ipa/ca.crt
```

Use the following command to download the Certificate Authority's root certificate to a file named `ca.crt`.

```
[root@bastion ]# scp
root@ipaserver.cdp.rdu2.scalelab.redhat.com:/etc/ipa/ca.crt /root/ca.crt
or
```

```
[root@bastion ]# wget -c -nv  
http://ipaserver.cdp.rdu2.scalelab.redhat.com/ipa/config/ca.crt
```

**STEP 3.** Create a configuration map containing the IdM Certificate Authority's root certificate to make OpenShift trust the IdM certificates.  
The ConfigMap key must be named ca.crt.

```
[root@bastion ]# oc create configmap ldap-ca-configmap -n openshift-config \  
--from-file=ca.crt=/root/ca.crt
```

**STEP 4.** Create a file to modify the OpenShift OAuth configuration by adding the LDAP identity provider and the other elements required for proper configuration.

```
[root@bastion ]# cat > ldap-cr.yaml << EOF  
apiVersion: config.openshift.io/v1  
kind: OAuth  
metadata:  
  name: cluster  
spec:  
  identityProviders:  
    - name: ldap  
      mappingMethod: claim  
      type: LDAP  
      ldap:  
        attributes:  
          email: []  
        id:  
          - dn  
        name:  
          - cn  
      preferredUsername:  
        - uid  
      bindDN:  
'uid=admin,cn=users,cn=accounts,dc=cdp,dc=rdu2,dc=scalelab,dc=redhat,dc=com'  
      bindPassword:  
        name: ldap-secret  
      ca:  
        name: ldap-ca-configmap  
      insecure: false  
      url:  
'ldaps://ipaserver.cdp.rdu2.scalelab.redhat.com/cn=users,cn=accounts,dc=cdp,dc=rdu2,d  
c=scalelab,dc=redhat,dc=com?uid'  
EOF
```

**STEP 5.** Apply the custom resource and wait until the pods in the openshift-authentication namespace are restarted so that the identity provider is active:

```
[root@bastion ]# oc apply -f ldap-cr.yaml
```

**STEP 6.** View the status, as the pods in the openshift-authentication namespace redeploy. This can take ten minutes or more before the pods are ready.

```
[root@bastion ]# oc get clusteroperator/authentication
```

```
[root@bastion ]# oc get pods -l app=oauth-openshift -n openshift-authentication
```

If there are any issues with the IdM integration, check the pod status and logs of the deployment.apps/oauth-openshift deployment in the openshift-authentication namespace.

```
[root@bastion ]# oc logs deployment.apps/oauth-openshift -n openshift-authentication
```

**STEP 7.** Add the cluster-admin role to the IdM admin user that needs cluster administration privileges.

```
[root@bastion ]# oc adm policy add-cluster-role-to-user cluster-admin admin
Warning: User 'admin' not found
clusterrole.rbac.authorization.k8s.io/cluster-admin added: "admin"
```

**STEP 8.** Verify that you are able to log in with the IdM admin user.

```
[root@bastion ]# oc login -u admin -p redhat123
https://api.vlan601.rdu2.scalelab.redhat.com:6443
```

```
[root@bastion ]# oc whoami
```

If necessary, the kubeadmin account is still available to use to login.

```
[root@bastion ]# oc login -u kubeadmin -p FvCFE-WIS52-EezGx-qDWY7
https://api.vlan601.rdu2.scalelab.redhat.com:6443
```

## Create OpenShift cluster using the Assisted Installer

The Assisted Installer is used to create an OpenShift cluster for CDP Data Services on bare-metal server nodes.

The following table is a reference of the bare-metal servers that will be used for the OpenShift cluster for CDP Data Services:

Hostname	CPU	RAM	Disk	MAC Address	IP Address
f36-h17-000-r640 (ocp master)	40	384 GiB	446 GiB	bc:97:e1:78:dd:91	10.1.49.216/23
f36-h18-000-r640 (ocp master)	40	384 GiB	446 GiB	bc:97:e1:78:d7:91	10.1.49.217/23
f36-h19-000-r640 (ocp master)	40	384 GiB	446 GiB	bc:97:e1:69:d2:41	10.1.49.218/23
d29-h11-000-r750 (ocp worker)	56	512 GiB	446 GiB	00:62:0b:1c:07:61	10.1.49.219/23
d29-h13-000-r750 (ocp worker)	56	512 GiB	446 GiB	00:62:0b:1d:d3:31	10.1.49.220/23
d30-h05-000-r750 (ocp worker)	56	512 GiB	446 GiB	00:62:0b:1d:9a:a1	10.1.49.221/23
d30-h07-000-r750 (ocp worker)	56	512 GiB	446 GiB	00:62:0b:1d:d1:a1	10.1.49.222/23

<domain\_name> for the RH scale lab will be: cdp.rdu2.scalelab.redhat.com

Installation parameters:

Parameter	Setting/Value
Cluster name	pvcocp
Base domain	cdp.rdu2.scalelab.redhat.com
OpenShift version	4.17.12
CPU architecture	x86_64
Integrate with external partner platforms	No platform integration
Number of control plane nodes	3 (highly available cluster)
Hosts' network configuration	Static IP, bridges, and bonds
Configure via	YAML view
Networking stack type	IPv4

DNS	10.1.49.1
Machine network	10.1.48.0/23
Default gateway	10.1.49.254
api.pvcocp.cdp.rdu2.scalelab.redhat.com (API IP)	10.1.49.5
*.apps.pvcocp.cdp.rdu2.scalelab.redhat.com (Ingress IP)	10.1.49.6

**STEP 1.** Create a new OpenShift cluster using the OpenShift Assisted Installer from the Red Hat Hybrid Cloud Console:

<https://console.redhat.com/openshift/assisted-installer/clusters/~new>

Enter the cluster details using the installation parameters provided, then click **Next**.



The screenshot shows the "Cluster details" step of the Red Hat Hybrid Cloud Console's Assisted Installer. The left sidebar lists steps 1 through 7. Step 1, "Cluster details", is selected and expanded. The form fields include:

- Cluster name:** pvcocp
- Base domain:** cdp.rdu2.scalelab.redhat.com
- OpenShift version:** OpenShift 4.17.42
- CPU architecture:** x86\_64
- Number of control plane nodes:** 3
- Include custom manifests:** (unchecked)
- Hosts' network configuration:** (radio buttons for) DHCP only and Static IP, bridges, and bonds
- Encryption of installation disks:** (radio buttons for) Control plane nodes, Workers, and Arbiter

At the bottom are "Next" and "Cancel" buttons, and a "View cluster events" link.

You're in Hybrid Cloud Console production mode. To see new pre-production features, turn on Preview mode.

**Static network configurations**

Network configuration can be done using either the form view or YAML view. Configurations done in this step are for discovering hosts.

**Host 1**  
MAC to Interface name mapping: bc:07:ef:78:d7:91 -> eno2np1

**Host 2**  
MAC to Interface name mapping: bc:07:ef:78:d7:91 -> eno2np1

**Host 3**  
MAC to Interface name mapping: bc:07:d1:69:d2:41 -> eno2np1

**Host 4**  
MAC to Interface name mapping: 00:62:0b:fc:07:b1 -> ens2np1

**Host 5**  
MAC to Interface name mapping: 00:62:0b:fd:d3:31 -> ens2np1

**Host 6**  
MAC to Interface name mapping: 00:62:0b:fd:9a:a1 -> ens2np1

**Host 7**  
MAC to Interface name mapping: 00:62:0b:fd:d1:a1 -> ens2np1

Add another host configuration   Copy the YAML content

## R640

```

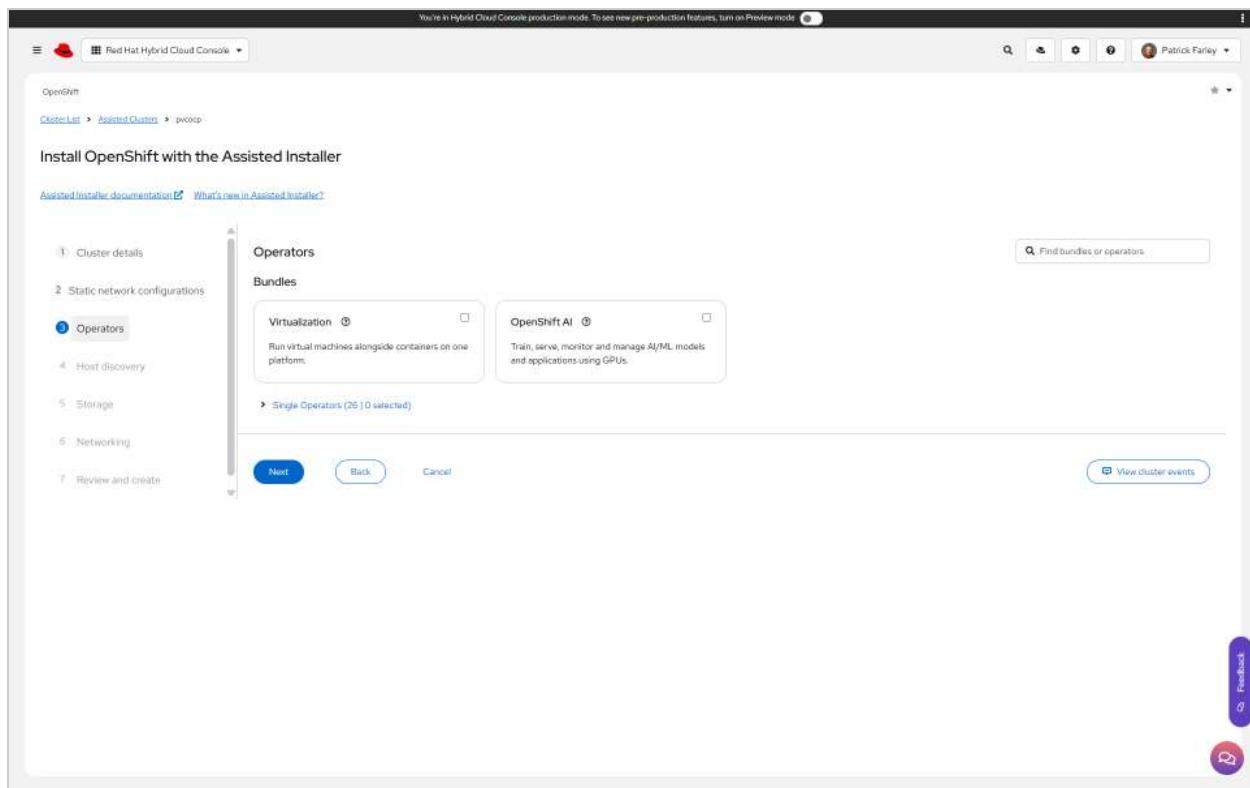
interfaces:
- ipv4:
  address:
    - ip: 10.1.49.216 # Edit this IP address for each server
      prefix-length: 23
    dhcp: false
    enabled: true
    name: eno2np1
    state: up
    type: ethernet
    mtu: 9000
- ipv4:
  dhcp: false
  enabled: false
  name: eno1np0
  state: down
  type: ethernet
routes:
  config:
    - destination: 0.0.0.0/0
      next-hop-address: 10.1.49.254
      next-hop-interface: eno2np1
      table-id: 254
dns-resolver:

```

```
config:  
  server:  
    - 10.1.49.1  
  search:  
    - cdp.rdu2.scalelab.redhat.com
```

R750

```
interfaces:  
- ipv4:  
  address:  
    - ip: 10.1.49.216 # Edit this IP address for each server  
      prefix-length: 23  
    dhcp: false  
    enabled: true  
  name: eno12409np1  
  state: up  
  type: ethernet  
  mtu: 9000  
- ipv4:  
  dhcp: false  
  enabled: false  
  name: eno12399np0  
  state: down  
  type: ethernet  
routes:  
  config:  
    - destination: 0.0.0.0/0  
      next-hop-address: 10.1.49.254  
      next-hop-interface: eno2np1  
      table-id: 254  
dns-resolver:  
  config:  
    server:  
      - 10.1.49.1  
    search:  
      - cdp.rdu2.scalelab.redhat.com
```



## DISCOVERY ISO IMAGE

```
dnf install -y httpd
```

```
wget -O discovery_image_pvcocp.iso  
'https://api.openshift.com/api/assisted-images/bytoken/eyJhbGciOiJIUzI1NiI  
sInR5cCI6IkpXVCJ9.eyJleHAiOjE3NjIzM2MDAsInN1YiI6IjY5YTEyMzk2LTAyMTQtNDkx  
Zi1hMWMyLWJjODUzZjlmYzY0NiJ9.3s-YXD5-BpeWK6Mfw3x9mR1wVaw7pPmJ0kWlisv6Cc/4  
.17/x86_64/full.iso'
```

```
cp discovery_image_pvcocp.iso /var/www/html
```

Verify that the discovery image is available at:  
[http://10.1.49.1/discovery\\_image\\_pvcocp.iso](http://10.1.49.1/discovery_image_pvcocp.iso)

## BADFISH

Configure the servers to boot from the Discovery image:

```
export USER=quads  
export PASS=rdu2@4618
```

```
dnf install -y podman
```

```
vi masterlist
```

```
10.6.61.148
```

```
10.6.61.152
```

```
10.6.61.153
```

```
for HOST in `cat ~/masterlist`;do podman run -it --rm quay.io/quads/badfish  
-H $HOST -u $USER -p $PASS --power-off;done
```

```
for HOST in `cat ~/masterlist`;do podman run -it --rm quay.io/quads/badfish  
-H $HOST -u $USER -p $PASS --unmount-virtual-media;done
```

```
for HOST in `cat ~/masterlist`;do podman run -it --rm quay.io/quads/badfish  
-H $HOST -u $USER -p $PASS --mount-virtual-media  
http://10.1.49.1/discovery_image_pvcocp.iso;done
```

```
for HOST in `cat ~/masterlist`;do podman run -it --rm quay.io/quads/badfish  
-H $HOST -u $USER -p $PASS --boot-to-virtual-media;done
```

```
for HOST in `cat ~/masterlist`;do podman run -it --rm quay.io/quads/badfish  
-H $HOST -u $USER -p $PASS --power-on;done
```

```
for HOST in `cat ~/masterlist`;do podman run -it --rm quay.io/quads/badfish  
-H $HOST -u $USER -p $PASS --power-state;done
```

**Get Current Boot mode setting**

```
for HOST in `cat ~/masterlist`;do podman run -it --rm quay.io/quads/badfish  
-H $HOST -u $USER -p $PASS --get-bios-attribute --attribute BootMode;done
```

**Set UEFI mode**

```
for HOST in `cat ~/masterlist`;do podman run -it --rm quay.io/quads/badfish  
-H $HOST -u $USER -p $PASS --set-bios-attribute --attribute BootMode --value  
Uefi;done
```

**Set BIOS mode**

```
for HOST in `cat ~/masterlist`;do podman run -it --rm quay.io/quads/badfish  
-H $HOST -u $USER -p $PASS --set-bios-attribute --attribute BootMode --value  
Bios;done
```

```
vi workerlist
```

```
10.1.37.21
```

```
10.1.64.196
10.1.65.226
10.1.65.96
10.1.65.139
10.1.64.17
10.1.65.3

for HOST in `cat ~/workerlist`;do podman run -it --rm quay.io/quads/badfish
-H $HOST -u $USER -p $PASS --power-off;done

for HOST in `cat ~/workerlist`;do podman run -it --rm quay.io/quads/badfish
-H $HOST -u $USER -p $PASS --unmount-virtual-media;done

for HOST in `cat ~/workerlist`;do podman run -it --rm quay.io/quads/badfish
-H $HOST -u $USER -p $PASS --mount-virtual-media
http://10.1.49.1/discovery_image_pvcocp.iso;done

for HOST in `cat ~/workerlist`;do podman run -it --rm quay.io/quads/badfish
-H $HOST -u $USER -p $PASS --boot-to-virtual-media;done

for HOST in `cat ~/workerlist`;do podman run -it --rm quay.io/quads/badfish
-H $HOST -u $USER -p $PASS --power-on;done

for HOST in `cat ~/workerlist`;do podman run -it --rm quay.io/quads/badfish
-H $HOST -u $USER -p $PASS --power-state;done

Get Current Boot mode setting
for HOST in `cat ~/workerlist`;do podman run -it --rm quay.io/quads/badfish
-H $HOST -u $USER -p $PASS --get-bios-attribute --attribute BootMode;done

Set UEFI mode
for HOST in `cat ~/workerlist`;do podman run -it --rm quay.io/quads/badfish
-H $HOST -u $USER -p $PASS --set-bios-attribute --attribute BootMode --value
Uefi;done

Set BIOS mode
for HOST in `cat ~/workerlist`;do podman run -it --rm quay.io/quads/badfish
-H $HOST -u $USER -p $PASS --set-bios-attribute --attribute BootMode --value
Bios;done
```

You're in Hybrid Cloud Console production mode. To see new pre-production features, turn on Preview mode.

### Host discovery

Add hosts

Run workloads on control plane nodes

**Information & Troubleshooting**

Minimum hardware requirements Hosts not showbooted?

### Host Inventory

Hostname	Role	Status	Discovered	CPU	Memory	Total
d29-h11-000-r750.cdp.rdu2.scalelab.redhat.com	Worker	Ready	04/11/2025, 22:51:43	112	512.00 GiB	45.77 TB
d29-h13-000-r750.cdp.rdu2.scalelab.redhat.com	Worker	Ready	04/11/2025, 22:52:16	112	512.00 GiB	45.77 TB
d30-h05-000-r750.cdp.rdu2.scalelab.redhat.com	Worker	Ready	04/11/2025, 22:52:15	112	512.00 GiB	45.77 TB
d30-h07-000-r750.cdp.rdu2.scalelab.redhat.com	Worker	Ready	04/11/2025, 22:51:19	112	512.00 GiB	45.77 TB
f36-h17-000-r640.cdp.rdu2.scalelab.redhat.com	Control plane node	Ready	04/11/2025, 22:32:26	80	384.00 GiB	958.58 GB
f36-h18-000-r640.cdp.rdu2.scalelab.redhat.com	Control plane node	Ready	04/11/2025, 22:32:49	80	384.00 GiB	958.58 GB
f36-h19-000-r640.cdp.rdu2.scalelab.redhat.com	Control plane node	Ready	04/11/2025, 22:33:03	80	384.00 GiB	958.58 GB

**Next** **Back** **Cancel** **View cluster events** **Feedback**

You're in Hybrid Cloud Console production mode. To see new pre-production features, turn on Preview mode.

### Storage

Hostname	Role	Status	Total storage	Number of disk
d29-h11-000-r750.cdp.rdu2.scalelab.redhat.com	Worker	Ready	45.77 TB	28
d29-h13-000-r750.cdp.rdu2.scalelab.redhat.com	Worker	Ready	45.77 TB	28
d30-h05-000-r750.cdp.rdu2.scalelab.redhat.com	Worker	Ready	45.77 TB	28
d30-h07-000-r750.cdp.rdu2.scalelab.redhat.com	Worker	Ready	45.77 TB	28
f36-h17-000-r640.cdp.rdu2.scalelab.redhat.com	Control plane node	Ready	958.58 GB	4
f36-h18-000-r640.cdp.rdu2.scalelab.redhat.com	Control plane node	Ready	958.58 GB	4
f36-h19-000-r640.cdp.rdu2.scalelab.redhat.com	Control plane node	Ready	958.58 GB	4

All bootable disks, except for read-only disks, will be formatted during installation. Make sure to back up any critical data before proceeding.

**Next** **Back** **Cancel** **View cluster events** **Feedback**

The screenshot shows the Red Hat Hybrid Cloud Console Assisted Installer interface. The top navigation bar indicates "You're in Hybrid Cloud Console production mode. To see new pre-production features, turn on Preview mode". The main title is "Install OpenShift with the Assisted Installer". The left sidebar lists steps: 1 Cluster details, 2 Static network configurations, 3 Operators, 4 Host discovery, 5 Storage, 6 Networking (selected), and 7 Review and create.

**Networking Configuration:**

- Network Management:** Cluster-Managed Networking (selected) vs User-Managed Networking.
- Networking stack type:** IPv4 (selected) vs Dual-stack.
- Machine network:** IP range 10.149.0/23 (10.149.0 – 10.149.255).
- API IP:** 10.149.5
- Ingress IP:** 10.149.6
- Use advanced networking:** Unchecked checkbox.
- Host SSH Public Key for troubleshooting after installation:** Checked checkbox.
- Use the same host discovery SSH key:** Checked checkbox.

**Host Inventory:**

Hostname	Role	Status	Active	IPv4 add.	IPv6	MAC address	...
d29-h1t-0000-r750.cdo.rdu2.scalelab.redhat.com	Worker	Ready Some validations failed	eno1@409np1	10.149.219/23	-	00:62:0b:fc:07:61	[...]
d29-h3-000-r750.cdo.rdu2.scalelab.redhat.com	Worker	Ready Some validations failed	eno1@409np1	10.149.220/23	-	00:62:0b:fd:d3:31	[...]
d30-h05-000-r750.cdo.rdu2.scalelab.redhat.com	Worker	Ready Some validations failed	eno1@409np1	10.149.221/23	-	00:62:0b:fd:9e:a1	[...]
d30-h07-000-r750.cdo.rdu2.scalelab.redhat.com	Worker	Ready Some validations failed	eno1@409np1	10.149.222/23	-	00:62:0b:fd:d1:a1	[...]
f36-h7-000-r540.cdo.rdu2.scalelab.redhat.com	Control plane node	Ready Some validations failed	eno2np1	10.149.216/23	-	bc:97:e1:78:dd:51	[...]
f36-h8-000-r640.cdo.rdu2.scalelab.redhat.com	Control plane node	Ready Some validations failed	eno2np1	10.149.217/23	-	bc:97:e1:78:d7:91	[...]
f36-h9-000-r640.cdo.rdu2.scalelab.redhat.com	Control plane node	Ready Some validations failed	eno2np1	10.149.218/23	-	bc:97:e1:69:d2:41	[...]

Buttons at the bottom: Next, Back, Cancel, View cluster events, and Feedback.

You're in Hybrid Cloud Console production mode. To see new pre-production features, turn on Preview mode.

**Cluster details**

- Cluster address: pvcocp.cdp.rdu2.scalelab.redhat.com
- OpenShift version: 4.17.42
- CPU architecture: x86\_64
- Hosts' network configuration: Static IP

**Host inventory**

Hosts	7
Total cores	688
Total memory	313 TiB
Total storage	185.95 TB

**Networking**

- Networking management type: Cluster-managed networking
- Stack type: IPv4
- Machine networks CIDR: 10.148.0/23
- API IP: 10.149.5
- Ingress IP: 10.149.6

**Advanced networking settings**

- Cluster network CIDR: 10.128.0.0/14
- Cluster network host prefix: 23
- Service network CIDR: 172.30.0.0/16
- Networking type: Open Virtual Network (OVN)

**Actions:** Install cluster, Back, Cancel, View cluster events, Feedback.

You're in Hybrid Cloud Console production mode. To see new pre-production features, turn on Preview mode.

**OpenShift**

[Cluster List](#) > [Added Cluster](#) > pvcocp

### pvcocp

**Installation progress**

Started on: 04/11/2025, 23:02:51  
Preparing for installation 0%

Control Planes	Workers	Initialization
0 control plane nodes installed	0 workers installed	Pending

**Actions:** Abort Installation, Download kubeconfig, View cluster events, Download Installation Logs, Feedback.

**Host inventory (7)**

Hostname	Role	Status	Discovered on	CPU Cores	Memory	Total storage
d29-h11-000-r750.cdp.rdu2.scalelab.redhat.com	Worker	Ready (Some validations failed)	04/11/2025, 22:51:43	112	512.00 GB	45.77 TB
d29-h13-000-r750.cdp.rdu2.scalelab.redhat.com	Worker	Ready (Some validations failed)	04/11/2025, 22:52:16	112	512.00 GB	45.77 TB
d30-h05-000-r750.cdp.rdu2.scalelab.redhat.com	Worker	Ready (Some validations failed)	04/11/2025, 22:52:16	112	512.00 GB	45.77 TB
d30-h07-000-r750.cdp.rdu2.scalelab.redhat.com	Worker	Ready (Some validations failed)	04/11/2025, 22:51:19	112	512.00 GB	45.77 TB
f36-h17-000-r640.cdp.rdu2.scalelab.redhat.com	Control plane node (bootstrap)	Ready (Some validations failed)	04/11/2025, 22:32:26	80	384.00 GB	958.58 GB
f36-h18-000-r640.cdp.rdu2.scalelab.redhat.com	Control plane node	Ready (Some validations failed)	04/11/2025, 22:32:49	80	384.00 GB	958.58 GB
f36-h19-000-r640.cdp.rdu2.scalelab.redhat.com	Control plane node	Ready (Some validations failed)	04/11/2025, 22:33:03	80	384.00 GB	958.58 GB

**Actions:** Cluster summary, Feedback.

You're in Hybrid Cloud Console production mode. To see new pre-production features, turn on Preview mode.

## pvcocp

**Installation progress**

Started on  
04/11/2025, 23:02:51  
Installing: 40%

Role	Status	Discoverd on	CPU C...	Memory	Total sto...
Control Planes	Installing 3 control plane nodes	04/11/2025, 22:51:43	102	512.00 GiB	45.77 TB
Workers	Installing 4 workers	04/11/2025, 22:52:16	102	512.00 GiB	45.77 TB
Initialization	Pending				

[Abort Installation](#) [Download kubeconfig](#) [View cluster events](#) [Download Installation Logs](#)

Download and save your kubeconfig file in a safe place. This file will be automatically deleted from Assisted Installer's service in 20 days.

[Host inventory \(7\)](#)

Hostname	Role	Status	Discoverd on	CPU C...	Memory	Total sto...
d29-h11-000-r750.cdp.rdu2.scalelab.redhat.com	Worker	Installing 4/4	04/11/2025, 22:51:43	102	512.00 GiB	45.77 TB
d29-h13-000-r750.cdp.rdu2.scalelab.redhat.com	Worker	Installing 4/4	04/11/2025, 22:52:16	102	512.00 GiB	45.77 TB
d30-h05-000-r750.cdp.rdu2.scalelab.redhat.com	Worker	Installing 4/4	04/11/2025, 22:52:15	102	512.00 GiB	45.77 TB
d30-h07-000-r750.cdp.rdu2.scalelab.redhat.com	Worker	Installing 4/4	04/11/2025, 22:51:19	102	512.00 GiB	45.77 TB
f36-h17-000-r640.cdp.rdu2.scalelab.redhat.com	Control plane node (bootstrap)	Installing 4/4	04/11/2025, 22:32:26	80	384.00 GiB	958.58 GB
f36-h18-000-r640.cdp.rdu2.scalelab.redhat.com	Control plane node	Installing 3/4	04/11/2025, 22:32:49	80	384.00 GiB	958.58 GB
f36-h19-000-r640.cdp.rdu2.scalelab.redhat.com	Control plane node	Installing 3/4	04/11/2025, 22:33:03	80	384.00 GiB	958.58 GB

[Cluster summary](#)

You're in Hybrid Cloud Console production mode. To see new pre-production features, turn on Preview mode.

## pvcocp

**Installation progress**

Started on  
05/11/2025, 10:31:54  
Installed on 05/11/2025, 11:09:05

Role	Status	Discoverd on	CPU C...	Memory	Total sto...
Control Planes	3 control plane nodes installed	05/11/2025, 10:31:54	102	512.00 GiB	45.77 TB
Workers	4 workers installed	05/11/2025, 11:09:05	102	512.00 GiB	45.77 TB
Initialization	Completed				

Installation completed successfully.

[Launch OpenShift Console](#) [Download kubeconfig](#) [View cluster events](#) [Download Installation Logs](#)

**Web Console URL:**  
<http://console.pvcocp.assisted-installer-20250411230251.rdu2.scalelab.redhat.com> [Edit](#)  
 Not able to access the Web Console?

**Username:** kubeadmin  
**Password:** \*\*\*\*

Download and save your kubeconfig file in a safe place. This file will be automatically deleted from Assisted Installer's service in 20 days.

Add new hosts by generating a new Discovery ISO under your cluster's "Add hosts" tab on [console.redhat.com/openshift](https://console.redhat.com/openshift).

Hostname	Role	Status	Discovered on	CPU Cores	Memory	Total stor...
d29-h11-000-r750.cdp.rdu2.scalelab.redhat.com	Worker	Installed	05/11/2025, 10:15:19	112	512.00 GB	45.29 TB
d29-h13-000-r750.cdp.rdu2.scalelab.redhat.com	Worker	Installed	05/11/2025, 10:15:56	112	512.00 GB	45.29 TB
d30-h05-000-r750.cdp.rdu2.scalelab.redhat.com	Worker	Installed	05/11/2025, 10:09:18	112	512.00 GB	45.29 TB
d30-h07-000-r750.cdp.rdu2.scalelab.redhat.com	Worker	Installed	05/11/2025, 10:09:34	112	512.00 GB	45.29 TB
f36-h17-000-r640.cdp.rdu2.scalelab.redhat.com	Control plane node (bootstrap)	Installed	05/11/2025, 10:19:50	80	384.00 GB	479.56 GB
f36-h18-000-r640.cdp.rdu2.scalelab.redhat.com	Control plane node	Installed	05/11/2025, 10:08:26	80	384.00 GB	479.56 GB
f36-h19-000-r640.cdp.rdu2.scalelab.redhat.com	Control plane node	Installed	05/11/2025, 10:26:14	80	384.00 GB	479.56 GB

Host inventory (7)

Cluster summary

Back to all clusters

<https://console.redhat.com/openshift/assisted-installer/clusters/415eb0de-ca80-47d8-a96c-151f60f58d20>

## Create OpenShift cluster with the Agent Based Installer (ABI)

The Agent Based Installer is used to create an OpenShift cluster for CDP Data Services. This method of OpenShift installation is used in place of the Assisted Installer.

- [Chapter 1. Preparing to install with the Agent-based Installer](#)
- [4.2. Installing OpenShift Container Platform with the Agent-based Installer](#)
- [4.2.1. Downloading the Agent-based Installer](#)
- [4.2.3. Creating the preferred configuration inputs](#)

**STEP 1:** Navigate to the [Red Hat Hybrid Cloud Console](#) using your login credentials.

**STEP 2:** Click *Download Installer* to download and extract the install program.

For this installation, we download the OpenShift installer for the latest stable release of **4.17**. Using this version of the installer will install the same version of OpenShift.

Download and extract the OpenShift installer using the following commands:

```
[root@ipaserver ~]# wget  
https://mirror.openshift.com/pub/openshift-v4/x86_64/clients/ocp/stable-4.17/  
openshift-install-linux.tar.gz  
[root@ipaserver ~]# tar xvf openshift-install-linux.tar.gz  
[root@ipaserver ~]# chmod +x openshift-install  
[root@ipaserver ~]# mv openshift-install /usr/local/bin/openshift-install
```

**STEP 3:** Download or copy the pull secret by clicking on *Download pull secret* or *Copy pull secret*. This will be used when the install-config.yaml file is created.

**STEP 4:** Click Download command-line tools and place the openshift-install binary in a directory that is on your PATH.

```
[root@ipaserver ~]# wget  
https://mirror.openshift.com/pub/openshift-v4/x86_64/clients/ocp/stable/openshift-client-li  
nux.tar.gz  
[root@ipaserver ~]# tar xvf openshift-client-linux.tar.gz  
[root@ipaserver ~]# chmod +x oc  
[root@ipaserver ~]# mv oc /usr/local/sbin/oc
```

**STEP 5:** Install the nmstate dependency by running the following command:

```
[root@ipaserver ~]# dnf install /usr/bin/nmstatectl -y
```

## install-config.yaml

**STEP 6:** Create the install-config.yaml file

```
[root@ipaserver ~]# cat > install-config.yaml << EOF  
apiVersion: v1  
baseDomain: rdu2.scalelab.redhat.com  
compute:  
- architecture: amd64  
  name: worker  
  replicas: 13  
controlPlane:  
  architecture: amd64  
  name: master  
  replicas: 3  
metadata:  
  name: cdp
```

```
networking:
  clusterNetwork:
    - cidr: 10.128.0.0/14
      hostPrefix: 23
  machineNetwork:
    - cidr: 10.1.48.0/23
  networkType: OVNKubernetes
  serviceNetwork:
    - 172.30.0.0/16
platform:
  baremetal:
    hosts:
      - name: pvcocp-master01.cdp.rdu2.scalelab.redhat.com
        role: master
        bootMACAddress: 02:00:e4:00:03:00
      - name: pvcocp-master02.cdp.rdu2.scalelab.redhat.com
        role: master
        bootMACAddress: 02:00:e4:00:03:01
      - name: pvcocp-master03.cdp.rdu2.scalelab.redhat.com
        role: master
        bootMACAddress: 02:00:e4:00:03:02
      - name: pvcocp-worker01.cdp.rdu2.scalelab.redhat.com
        role: worker
        bootMACAddress: 02:00:e4:00:03:03
      - name: pvcocp-worker02.cdp.rdu2.scalelab.redhat.com
        role: worker
        bootMACAddress: 02:00:e4:00:03:04
      - name: pvcocp-worker03.cdp.rdu2.scalelab.redhat.com
        role: worker
        bootMACAddress: 02:00:e4:00:03:05
      - name: pvcocp-worker04.cdp.rdu2.scalelab.redhat.com
        role: worker
        bootMACAddress: 02:00:e4:00:03:06
      - name: pvcocp-worker05.cdp.rdu2.scalelab.redhat.com
        role: worker
        bootMACAddress: 02:00:e4:00:03:07
      - name: pvcocp-worker06.cdp.rdu2.scalelab.redhat.com
        role: worker
        bootMACAddress: 02:00:e4:00:03:08
      - name: pvcocp-worker07.cdp.rdu2.scalelab.redhat.com
        role: worker
        bootMACAddress: 02:00:e4:00:03:09
      - name: pvcocp-worker08.cdp.rdu2.scalelab.redhat.com
        role: worker
        bootMACAddress: 02:00:e4:00:03:0a
      - name: pvcocp-worker09.cdp.rdu2.scalelab.redhat.com
```



```

3bHJUajF0WEJV0HB2dEZYSGV3QzcN2k2LWJkeEZTWTBvT3J1S21ydFdpT3lpTmlDekR2YU11X1Bj
Skx4cXNFUEJRZUdPeWd4aDVORHNjRVB5QmtZYTd1OHYtdVFVX3VldzNYSxo5d28yZjhsS1c1cGVPa
mJzZi1jcWxmX19INUZ4UWI5M24z0Ww1X291Ykh1NHFPcW9IM2U2Z2FCWERhZmx1UE1kYzRENkU2MD
NHZkxXY1NQcFFMNVE1QTFaa3Nhc183NUl1ejVkyU5UM1padTVo0ER3YVY2dTQ2TGszNU90b1kybzR
oUWhXYkxPUmpalto5LVplVnN2STkxaXIweVk5YkdyZ3MyQ1ZUN252MFowU3pNdEhpR0o0aWpzYXpj
Y3dMbVUtM2poSUdBaXREaEN2eEJiY0plUGVWaENwZ3dGTkstMFdZT0V1cE1HN1RmSjE5WHR1V1FVb
DNzaT1QV1g5Q2RHdU93RHViV1NMQXJ5RXAtTV1Zdw==", "email": "pfarley@redhat.com"}}
sshKey: |
  ssh-rsa
AAAAB3NzaC1yc2EAAAQABAAABgQC5f46fvSfwLSXx0zCC52qFa21AS0kvJoolggoGL+CUWGdq6
Kf4ExWCJ2d1iDkAcPt6y7jGshujhfUR6DE1kuFpozT0gqbjwZ17ICtA51mkhb7sxHuYYNFAZNl1i6D
38IbdMSfh23I2ypBpcQA0WeCzfPnjsjanilG5EfTw3K75QQ0MxLvE4W0uKbA/EKg9F8Eq2ImKL
HzekB8nCFoi2mVgwEYGTSM5zAz56tuSbuppbkXfgf+4FFb/LFp3TJyCPQu396ZRbEF64EWvbMC
01aAYNM3NpRqSqsGWWJoTnRzXSRKb096t0xWmTYrIf0s/nuRA1Sb3kU0EN4qC5YQs9wobpnLQ5T
3Um8QP4pQxWhWIHEAW3SWM01IfEBjT8TPS80CG3EfLRVxJFdNZc/Ifd3CGnevW8ov1Sek//HfspwM
MMauXQfY570fxhn9BxVtnC6ymG1DBv9hoa82qtvigfkMU1u0xD9Ep7rl14vprcQ5PF18yp09DPsKj
4Etc= root@ipaserver.cdp.rdu2.scalelab.redhat.com
EOF

```

## agent-config.yaml

STEP 7: Create the agent-config.yaml file

Hostname	CPU	RAM	Disk	MAC Address	IP Address
pvcocp-master01.<domain_name>	8	16 GiB	120 GiB	02:00:e4:00:03:00	10.1.49.200/23
pvcocp-master02.<domain_name>	8	16 GiB	120 GiB	02:00:e4:00:03:01	10.1.49.201/23
pvcocp-master03.<domain_name>	8	16 GiB	120 GiB	02:00:e4:00:03:02	10.1.49.202/23
pvcocp-worker01.<domain_name>	32	64 GiB	120 GiB	02:00:e4:00:03:03	10.1.49.203/23
pvcocp-worker02.<domain_name>	32	64 GiB	120 GiB	02:00:e4:00:03:04	10.1.49.204/23
pvcocp-worker03.<domain_name>	32	64 GiB	120 GiB	02:00:e4:00:03:05	10.1.49.205/23
pvcocp-worker04.<domain_name>	32	64 GiB	120 GiB	02:00:e4:00:03:06	10.1.49.206/23
pvcocp-worker05.<domain_name>	32	64 GiB	120 GiB	02:00:e4:00:03:07	10.1.49.207/23
pvcocp-worker06.<domain_name>	32	64 GiB	120 GiB	02:00:e4:00:03:08	10.1.49.208/23
pvcocp-worker07.<domain_name>	32	64 GiB	120 GiB	02:00:e4:00:03:09	10.1.49.209/23
pvcocp-worker08.<domain_name>	32	64 GiB	120 GiB	02:00:e4:00:03:0a	10.1.49.210/23
pvcocp-worker09.<domain_name>	32	64 GiB	120 GiB	02:00:e4:00:03:0b	10.1.49.211/23

pvcocp-worker10.<domain_name>	32	64 GiB	120 GiB	02:00:e4:00:03:0c	10.1.49.212/23
pvcocp-infra01.<domain_name>	32	64 GiB	120 GiB	02:00:e4:00:03:0d	10.1.49.213/23
pvcocp-infra02.<domain_name>	32	64 GiB	120 GiB	02:00:e4:00:03:0e	10.1.49.214/23
pvcocp-infra03.<domain_name>	32	64 GiB	120 GiB	02:00:e4:00:03:0f	10.1.49.215/23

<domain\_name> for the RH scale lab will be: cdp.rdu2.scalelab.redhat.com

[root@ipaserver ~]# cat > agent-config.yaml << EOF

```
apiVersion: v1alpha1
kind: AgentConfig
metadata:
  name: cdp
rendezvousIP: 10.1.49.200
hosts:
  - hostname: master01.cdp.rdu2.scalelab.redhat.com
    interfaces:
      - name: ens192
        macAddress: 02:00:e4:00:03:00 # Change for each host
networkConfig:
  interfaces:
    - name: ens192
      type: ethernet
      mtu: 9000
      state: up
      mac-address: 02:00:e4:00:03:00 # Change for each host
      ipv4:
        enabled: true
        address:
          - ip: 10.1.49.200 # Change for each host
            prefix-length: 23
        dhcp: false
  dns-resolver:
    config:
      server:
        - 10.1.49.1
  routes:
    config:
      - destination: 0.0.0.0/0
        next-hop-address: 10.1.49.254
        next-hop-interface: ens192
```

```
        table-id: 254
- hostname: master02.cdp.rdu2.scalelab.redhat.com
  interfaces:
    - name: ens192
      macAddress: 02:00:e4:00:03:01 # Change for each host
  networkConfig:
    interfaces:
      - name: ens192
        type: ethernet
        mtu: 9000
        state: up
        mac-address: 02:00:e4:00:03:01 # Change for each host
      ipv4:
        enabled: true
        address:
          - ip: 10.1.49.201 # Change for each host
            prefix-length: 23
        dhcp: false
  dns-resolver:
    config:
      server:
        - 10.1.49.1
  routes:
    config:
      - destination: 0.0.0.0/0
        next-hop-address: 10.1.49.254
        next-hop-interface: ens192
        table-id: 254
- hostname: master03.cdp.rdu2.scalelab.redhat.com
  interfaces:
    - name: ens192
      macAddress: 02:00:e4:00:03:02 # Change for each host
  networkConfig:
    interfaces:
      - name: ens192
        type: ethernet
        mtu: 9000
        state: up
        mac-address: 02:00:e4:00:03:02 # Change for each host
      ipv4:
        enabled: true
        address:
          - ip: 10.1.49.202 # Change for each host
            prefix-length: 23
        dhcp: false
  dns-resolver:
```

```

config:
  server:
    - 10.1.49.1
routes:
  config:
    - destination: 0.0.0.0/0
      next-hop-address: 10.1.49.254
      next-hop-interface: ens192
      table-id: 254
  - hostname: worker01.cdp.rdu2.scalelab.redhat.com
interfaces:
  - name: ens192
    macAddress: 02:00:e4:00:03:03 # Change for each host
networkConfig:
  interfaces:
    - name: ens192
      type: ethernet
      mtu: 9000
      state: up
      mac-address: 02:00:e4:00:03:03 # Change for each host
      ipv4:
        enabled: true
        address:
          - ip: 10.1.49.203 # Change for each host
            prefix-length: 23
        dhcp: false
  dns-resolver:
    config:
      server:
        - 10.1.49.1
routes:
  config:
    - destination: 0.0.0.0/0
      next-hop-address: 10.1.49.254
      next-hop-interface: ens192
      table-id: 254
  - hostname: worker02.cdp.rdu2.scalelab.redhat.com
interfaces:
  - name: ens192
    macAddress: 02:00:e4:00:03:04 # Change for each host
networkConfig:
  interfaces:
    - name: ens192
      type: ethernet
      mtu: 9000
      state: up

```

```

mac-address: 02:00:e4:00:03:04 # Change for each host
ipv4:
  enabled: true
  address:
    - ip: 10.1.49.204 # Change for each host
      prefix-length: 23
  dhcp: false
dns-resolver:
  config:
    server:
      - 10.1.49.1
routes:
  config:
    - destination: 0.0.0.0/0
      next-hop-address: 10.1.49.254
      next-hop-interface: ens192
      table-id: 254
- hostname: worker03.cdp.rdu2.scalelab.redhat.com
interfaces:
  - name: ens192
    macAddress: 02:00:e4:00:03:05 # Change for each host
networkConfig:
  interfaces:
    - name: ens192
      type: ethernet
      mtu: 9000
      state: up
      mac-address: 02:00:e4:00:03:05 # Change for each host
      ipv4:
        enabled: true
        address:
          - ip: 10.1.49.205 # Change for each host
            prefix-length: 23
        dhcp: false
dns-resolver:
  config:
    server:
      - 10.1.49.1
routes:
  config:
    - destination: 0.0.0.0/0
      next-hop-address: 10.1.49.254
      next-hop-interface: ens192
      table-id: 254
- hostname: worker04.cdp.rdu2.scalelab.redhat.com
interfaces:

```

```
- name: ens192
  macAddress: 02:00:e4:00:03:06 # Change for each host
networkConfig:
  interfaces:
    - name: ens192
      type: ethernet
      mtu: 9000
      state: up
      mac-address: 02:00:e4:00:03:06 # Change for each host
      ipv4:
        enabled: true
        address:
          - ip: 10.1.49.206 # Change for each host
            prefix-length: 23
        dhcp: false
  dns-resolver:
    config:
      server:
        - 10.1.49.1
  routes:
    config:
      - destination: 0.0.0.0/0
        next-hop-address: 10.1.49.254
        next-hop-interface: ens192
        table-id: 254
- hostname: worker05.cdp.rdu2.scalelab.redhat.com
  interfaces:
    - name: ens192
      macAddress: 02:00:e4:00:03:07 # Change for each host
networkConfig:
  interfaces:
    - name: ens192
      type: ethernet
      mtu: 9000
      state: up
      mac-address: 02:00:e4:00:03:07 # Change for each host
      ipv4:
        enabled: true
        address:
          - ip: 10.1.49.207 # Change for each host
            prefix-length: 23
        dhcp: false
  dns-resolver:
    config:
      server:
        - 10.1.49.1
```

```
routes:
  config:
    - destination: 0.0.0.0/0
      next-hop-address: 10.1.49.254
      next-hop-interface: ens192
      table-id: 254
    - hostname: worker06.cdp.rdu2.scalelab.redhat.com
  interfaces:
    - name: ens192
      macAddress: 02:00:e4:00:03:08 # Change for each host
  networkConfig:
    interfaces:
      - name: ens192
        type: ethernet
        mtu: 9000
        state: up
        mac-address: 02:00:e4:00:03:08 # Change for each host
      ipv4:
        enabled: true
        address:
          - ip: 10.1.49.208 # Change for each host
            prefix-length: 23
        dhcp: false
    dns-resolver:
      config:
        server:
          - 10.1.49.1
  routes:
    config:
      - destination: 0.0.0.0/0
        next-hop-address: 10.1.49.254
        next-hop-interface: ens192
        table-id: 254
    - hostname: worker07.cdp.rdu2.scalelab.redhat.com
  interfaces:
    - name: ens192
      macAddress: 02:00:e4:00:03:09 # Change for each host
  networkConfig:
    interfaces:
      - name: ens192
        type: ethernet
        mtu: 9000
        state: up
        mac-address: 02:00:e4:00:03:09 # Change for each host
      ipv4:
        enabled: true
```

```

address:
  - ip: 10.1.49.209 # Change for each host
    prefix-length: 23
  dhcp: false
dns-resolver:
  config:
    server:
      - 10.1.49.1
routes:
  config:
    - destination: 0.0.0.0/0
      next-hop-address: 10.1.49.254
      next-hop-interface: ens192
      table-id: 254
- hostname: worker08.cdp.rdu2.scalelab.redhat.com
interfaces:
  - name: ens192
    macAddress: 02:00:e4:00:03:0a # Change for each host
networkConfig:
  interfaces:
    - name: ens192
      type: ethernet
      mtu: 9000
      state: up
      mac-address: 02:00:e4:00:03:0a # Change for each host
      ipv4:
        enabled: true
        address:
          - ip: 10.1.49.210 # Change for each host
            prefix-length: 23
        dhcp: false
  dns-resolver:
    config:
      server:
        - 10.1.49.1
  routes:
    config:
      - destination: 0.0.0.0/0
        next-hop-address: 10.1.49.254
        next-hop-interface: ens192
        table-id: 254
- hostname: worker09.cdp.rdu2.scalelab.redhat.com
interfaces:
  - name: ens192
    macAddress: 02:00:e4:00:03:0b # Change for each host
networkConfig:

```

```

interfaces:
  - name: ens192
    type: ethernet
    mtu: 9000
    state: up
    mac-address: 02:00:e4:00:03:0b # Change for each host
    ipv4:
      enabled: true
      address:
        - ip: 10.1.49.211 # Change for each host
          prefix-length: 23
      dhcp: false
    dns-resolver:
      config:
        server:
          - 10.1.49.1
    routes:
      config:
        - destination: 0.0.0.0/0
          next-hop-address: 10.1.49.254
          next-hop-interface: ens192
          table-id: 254
- hostname: worker10.cdp.rdu2.scalelab.redhat.com
  interfaces:
    - name: ens192
      macAddress: 02:00:e4:00:03:0c # Change for each host
  networkConfig:
    interfaces:
      - name: ens192
        type: ethernet
        mtu: 9000
        state: up
        mac-address: 02:00:e4:00:03:0c # Change for each host
      ipv4:
        enabled: true
        address:
          - ip: 10.1.49.212 # Change for each host
            prefix-length: 23
        dhcp: false
    dns-resolver:
      config:
        server:
          - 10.1.49.1
    routes:
      config:
        - destination: 0.0.0.0/0

```

```

        next-hop-address: 10.1.49.254
        next-hop-interface: ens192
        table-id: 254
- hostname: infra01.cdp.rdu2.scalelab.redhat.com
interfaces:
- name: ens192
  macAddress: 02:00:e4:00:03:0d # Change for each host
networkConfig:
interfaces:
- name: ens192
  type: ethernet
  mtu: 9000
  state: up
  mac-address: 02:00:e4:00:03:0d # Change for each host
ipv4:
  enabled: true
  address:
- ip: 10.1.49.213 # Change for each host
  prefix-length: 23
  dhcp: false
dns-resolver:
config:
server:
- 10.1.49.1
routes:
config:
- destination: 0.0.0.0/0
  next-hop-address: 10.1.49.254
  next-hop-interface: ens192
  table-id: 254
- hostname: infra02.cdp.rdu2.scalelab.redhat.com
interfaces:
- name: ens192
  macAddress: 02:00:e4:00:03:0e # Change for each host
networkConfig:
interfaces:
- name: ens192
  type: ethernet
  mtu: 9000
  state: up
  mac-address: 02:00:e4:00:03:0e # Change for each host
  ipv4:
    enabled: true
    address:
- ip: 10.1.49.214 # Change for each host
    prefix-length: 23

```

```

        dhcp: false
dns-resolver:
  config:
    server:
      - 10.1.49.1
routes:
  config:
    - destination: 0.0.0.0/0
      next-hop-address: 10.1.49.254
      next-hop-interface: ens192
      table-id: 254
- hostname: infra03.cdp.rdu2.scalelab.redhat.com
interfaces:
  - name: ens192
    macAddress: 02:00:e4:00:03:0f # Change for each host
networkConfig:
  interfaces:
    - name: ens192
      type: ethernet
      mtu: 9000
      state: up
      mac-address: 02:00:e4:00:03:0f # Change for each host
      ipv4:
        enabled: true
        address:
          - ip: 10.1.49.215 # Change for each host
            prefix-length: 23
        dhcp: false
  dns-resolver:
    config:
      server:
        - 10.1.49.1
  routes:
    config:
      - destination: 0.0.0.0/0
        next-hop-address: 10.1.49.254
        next-hop-interface: ens192
        table-id: 254

```

EOF

#### STEP 8: Create Install Dir

```
[root@ipaserver ~]# mkdir ocp-install-dir
```

## STEP 9: Create backup copies of the yaml manifests

```
[root@ipaserver ~]# cp agent-config.yaml ocp-install-dir/agent-config.yaml  
[root@ipaserver ~]# cp install-config.yaml  
ocp-install-dir/install-config.yaml
```

## STEP 10: Create ISO agent image

```
[root@ipaserver ~]# openshift-install --dir ocp-install-dir agent create  
image  
INFO Configuration has 3 master replicas and 13 worker replicas  
WARNING hosts from install-config.yaml are ignored  
INFO The rendezvous host IP (node0 IP) is 10.1.49.200  
INFO Extracting base ISO from release payload  
INFO Verifying cached file  
INFO Using cached Base ISO /root/.cache/agent/image_cache/coreos-x86_64.iso  
INFO Consuming Install Config from target directory  
INFO Consuming Agent Config from target directory  
INFO Generated ISO at ocp-install-dir/agent.x86_64.iso.
```

## STEP 11: Upload the Discovery ISO image to a PVC using the virtctl utility:

```
[root@ipaserver ~]# oc project cdppvcds  
  
[root@ipaserver ~]# virtctl image-upload dv discovery-image-cdp --size=2Gi  
--image-path=ocp-install-dir/agent.x86_64.iso --insecure  
--storage-class=ocs-storagecluster-ceph-rbd  
PVC cdp-ds/discovery-image-cdp not found  
DataVolume cdp-ds/discovery-image-cdp created  
Waiting for PVC discovery-image-cdp upload pod to be ready...  
Pod now ready  
Uploading data to https://cdi-uploadproxy-openshift-cnv.apps.vlan601.rdu2.scaleLab.redhat.com  
  
1.24 GiB / 1.24 GiB  
[-----]  
-----] 100.00% 161.43 MiB p/s  
8.1s  
  
Uploading data completed successfully, waiting for processing to complete, you can hit ctrl-c  
without interrupting the progress  
Processing completed successfully  
Uploading ocp-install-dir/agent.x86_64.iso completed successfully
```

## Create VMs for OpenShift Cluster (OpenShift Virtualization)

The following table is a reference of the VirtualMachines that will be created for the OpenShift cluster for CDP Data Services:

Hostname	CPU	RAM	Disk	MAC Address	IP Address
pvcocp-master01.<domain_name>	8	16 GiB	120 GiB	02:00:e4:00:03:00	10.1.49.200/23
pvcocp-master02.<domain_name>	8	16 GiB	120 GiB	02:00:e4:00:03:01	10.1.49.201/23
pvcocp-master03.<domain_name>	8	16 GiB	120 GiB	02:00:e4:00:03:02	10.1.49.202/23
pvcocp-worker01.<domain_name>	32	64 GiB	120 GiB	02:00:e4:00:03:03	10.1.49.203/23
pvcocp-worker02.<domain_name>	32	64 GiB	120 GiB	02:00:e4:00:03:04	10.1.49.204/23
pvcocp-worker03.<domain_name>	32	64 GiB	120 GiB	02:00:e4:00:03:05	10.1.49.205/23
pvcocp-worker04.<domain_name>	32	64 GiB	120 GiB	02:00:e4:00:03:06	10.1.49.206/23
pvcocp-worker05.<domain_name>	32	64 GiB	120 GiB	02:00:e4:00:03:07	10.1.49.207/23
pvcocp-worker06.<domain_name>	32	64 GiB	120 GiB	02:00:e4:00:03:08	10.1.49.208/23
pvcocp-worker07.<domain_name>	32	64 GiB	120 GiB	02:00:e4:00:03:09	10.1.49.209/23
pvcocp-worker08.<domain_name>	32	64 GiB	120 GiB	02:00:e4:00:03:0a	10.1.49.210/23
pvcocp-worker09.<domain_name>	32	64 GiB	120 GiB	02:00:e4:00:03:0b	10.1.49.211/23
pvcocp-worker10.<domain_name>	32	64 GiB	120 GiB	02:00:e4:00:03:0c	10.1.49.212/23
pvcocp-infra01.<domain_name>	32	64 GiB	120 GiB	02:00:e4:00:03:0d	10.1.49.213/23
pvcocp-infra02.<domain_name>	32	64 GiB	120 GiB	02:00:e4:00:03:0e	10.1.49.214/23
pvcocp-infra03.<domain_name>	32	64 GiB	120 GiB	02:00:e4:00:03:0f	10.1.49.215/23

<domain\_name> for the RH scale lab will be: cdp.rdu2.scalelab.redhat.com

**STEP 13:** Create the VM's for the OpenShift cluster using the following VM YAML files:

```
[root@bastion ~]# oc new-project cdp-ds
[root@bastion ~]# oc create -f <nodename>-vm.yaml
```

```
for i in {100..199}; do oc create -f <nodename>-vm.yaml; done
```

## pvcocp-master01.cdp.rdu2.scalelab.redhat.com-vms.yaml

```
apiVersion: kubevirt.io/v1
kind: VirtualMachine
metadata:
  labels:
    app: pvcocp-master01.cdp.rdu2.scalelab.redhat.com
    kubevirt.io/dynamic-credentials-support: 'true'
    vm.kubevirt.io/template: rhel9-server-small
    vm.kubevirt.io/template.namespace: openshift
    vm.kubevirt.io/template.revision: '1'
    vm.kubevirt.io/template.version: v0.32.2
  name: pvcocp-master01.cdp.rdu2.scalelab.redhat.com
  namespace: cdppvcds
spec:
  dataVolumeTemplates:
    - apiVersion: cdi.kubevirt.io/v1beta1
      kind: DataVolume
      metadata:
        annotations:
          cdi.kubevirt.io/storage.bind.immediate.requested: 'true'
        creationTimestamp: null
        name: pvcocp-master01.cdp.rdu2.scalelab.redhat.com
      spec:
        source:
          blank: {}
        storage:
          resources:
            requests:
              storage: 120Gi
    - metadata:
        name: pvcocp-master01.cdp.rdu2.scalelab.redhat.com-cdrom
      spec:
        source:
          pvc:
            name: discovery-image-cdp
            namespace: cdppvcds
        storage:
          resources:
            requests:
              storage: 2Gi
  runStrategy: RerunOnFailure
  template:
    metadata:
      annotations:
```

```
vm.kubevirt.io/flavor: small
vm.kubevirt.io/os: rhel9
vm.kubevirt.io/workload: server
creationTimestamp: null
labels:
  kubevirt.io/domain: pvcocp-master01.cdp.rdu2.scalelab.redhat.com
  kubevirt.io/size: small
spec:
  architecture: amd64
  domain:
    cpu:
      cores: 1
      sockets: 8
      threads: 1
    devices:
      disks:
        - bootOrder: 1
          disk:
            bus: virtio
            name: rootdisk
        - bootOrder: 2
          cdrom:
            bus: sata
            name: installation-cdrom
    interfaces:
      - bridge: {}
        model: virtio
        macAddress: '02:00:e4:00:03:00'
        name: eth0
    rng: {}
  features:
    acpi: {}
    smm:
      enabled: true
  firmware:
    bootloader:
      efi: {}
  machine:
    type: pc-q35-rhel9.4.0
  memory:
    guest: 16Gi
    resources: {}
  networks:
    - multus:
        networkName: default/localnet-network
        name: eth0
```

```

terminationGracePeriodSeconds: 180
volumes:
  - dataVolume:
      name: pvcocp-master01.cdp.rdu2.scalelab.redhat.com
      name: rootdisk
  - dataVolume:
      name: pvcocp-master01.cdp.rdu2.scalelab.redhat.com-cdrom
      name: installation-cdrom

```

## pvcocp-master02.cdp.rdu2.scalelab.redhat.com-vms.yaml

```

apiVersion: kubevirt.io/v1
kind: VirtualMachine
metadata:
  labels:
    app: pvcocp-master02.cdp.rdu2.scalelab.redhat.com
    kubevirt.io/dynamic-credentials-support: 'true'
    vm.kubevirt.io/template: rhel9-server-small
    vm.kubevirt.io/template.namespace: openshift
    vm.kubevirt.io/template.revision: '1'
    vm.kubevirt.io/template.version: v0.32.2
  name: pvcocp-master02.cdp.rdu2.scalelab.redhat.com
  namespace: cdppvcds
spec:
  dataVolumeTemplates:
    - apiVersion: cdi.kubevirt.io/v1beta1
      kind: DataVolume
      metadata:
        annotations:
          cdi.kubevirt.io/storage.bind.immediate.requested: 'true'
          creationTimestamp: null
          name: pvcocp-master02.cdp.rdu2.scalelab.redhat.com
      spec:
        source:
          blank: {}
        storage:
          resources:
            requests:
              storage: 120Gi
    - metadata:
        name: pvcocp-master02.cdp.rdu2.scalelab.redhat.com-cdrom
      spec:
        source:
          pvc:
            name: discovery-image-cdp
            namespace: cdppvcds

```

```
storage:
  resources:
    requests:
      storage: 2Gi
runStrategy: RerunOnFailure
template:
  metadata:
    annotations:
      vm.kubevirt.io/flavor: small
      vm.kubevirt.io/os: rhel9
      vm.kubevirt.io/workload: server
  creationTimestamp: null
  labels:
    kubevirt.io/domain: pvcocp-master02.cdp.rdu2.scalelab.redhat.com
    kubevirt.io/size: small
spec:
  architecture: amd64
  domain:
    cpu:
      cores: 1
      sockets: 8
      threads: 1
    devices:
      disks:
        - bootOrder: 1
          disk:
            bus: virtio
            name: roottdisk
        - bootOrder: 2
          cdrom:
            bus: sata
            name: installation-cdrom
    interfaces:
      - bridge: {}
        model: virtio
        macAddress: '02:00:e4:00:03:01'
        name: eth0
    rng: {}
  features:
    acpi: {}
    smm:
      enabled: true
  firmware:
    bootloader:
      efi: {}
  machine:
```

```

        type: pc-q35-rhel9.4.0
    memory:
        guest: 16Gi
    resources: {}
networks:
- multus:
    networkName: default/localnet-network
    name: eth0
terminationGracePeriodSeconds: 180
volumes:
- dataVolume:
    name: pvcocp-master02.cdp.rdu2.scalelab.redhat.com
    name: rootdisk
- dataVolume:
    name: pvcocp-master02.cdp.rdu2.scalelab.redhat.com-cdrom
    name: installation-cdrom

```

## pvcocp-master03.cdp.rdu2.scalelab.redhat.com-vms.yaml

```

apiVersion: kubevirt.io/v1
kind: VirtualMachine
metadata:
labels:
    app: pvcocp-master03.cdp.rdu2.scalelab.redhat.com
    kubevirt.io/dynamic-credentials-support: 'true'
    vm.kubevirt.io/template: rhel9-server-small
    vm.kubevirt.io/template.namespace: openshift
    vm.kubevirt.io/template.revision: '1'
    vm.kubevirt.io/template.version: v0.32.2
name: pvcocp-master03.cdp.rdu2.scalelab.redhat.com
namespace: cdppvcds
spec:
dataVolumeTemplates:
- apiVersion: cdi.kubevirt.io/v1beta1
  kind: DataVolume
  metadata:
    annotations:
      cdi.kubevirt.io/storage.bind.immediate.requested: 'true'
    creationTimestamp: null
    name: pvcocp-master03.cdp.rdu2.scalelab.redhat.com
spec:
source:
blank: {}
storage:
resources:
requests:

```

```
        storage: 120Gi
- metadata:
    name: pvcocp-master03.cdp.rdu2.scalelab.redhat.com-cdrom
spec:
source:
pvc:
    name: discovery-image-cdp
    namespace: cdppvcds
storage:
resources:
requests:
storage: 2Gi
runStrategy: RerunOnFailure
template:
metadata:
annotations:
    vm.kubevirt.io/flavor: small
    vm.kubevirt.io/os: rhel9
    vm.kubevirt.io/workload: server
creationTimestamp: null
labels:
    kubevirt.io/domain: pvcocp-master03.cdp.rdu2.scalelab.redhat.com
    kubevirt.io/size: small
spec:
architecture: amd64
domain:
cpu:
cores: 1
sockets: 8
threads: 1
devices:
disks:
- bootOrder: 1
disk:
bus: virtio
name: rootdisk
- bootOrder: 2
cdrom:
bus: sata
name: installation-cdrom
interfaces:
- bridge: {}
model: virtio
macAddress: '02:00:e4:00:03:02'
name: eth0
rng: {}
```

```

features:
  acpi: {}
  smm:
    enabled: true
firmware:
  bootloader:
    efi: {}
machine:
  type: pc-q35-rhel9.4.0
memory:
  guest: 16Gi
resources: {}
networks:
  - multus:
      networkName: default/localnet-network
      name: eth0
terminationGracePeriodSeconds: 180
volumes:
  - dataVolume:
      name: pvcocp-worker01.cdp.rdu2.scalelab.redhat.com
      name: rootdisk
  - dataVolume:
      name: pvcocp-worker01.cdp.rdu2.scalelab.redhat.com-cdrom
      name: installation-cdrom

```

## pvcocp-worker01.cdp.rdu2.scalelab.redhat.com-vms.yaml

```

apiVersion: kubevirt.io/v1
kind: VirtualMachine
metadata:
  labels:
    app: pvcocp-worker01.cdp.rdu2.scalelab.redhat.com
    kubevirt.io/dynamic-credentials-support: 'true'
    vm.kubevirt.io/template: rhel9-server-small
    vm.kubevirt.io/template.namespace: openshift
    vm.kubevirt.io/template.revision: '1'
    vm.kubevirt.io/template.version: v0.32.2
  name: pvcocp-worker01.cdp.rdu2.scalelab.redhat.com
  namespace: cdppvcds
spec:
  dataVolumeTemplates:
    - apiVersion: cdi.kubevirt.io/v1beta1
      kind: DataVolume
      metadata:
        annotations:
          cdi.kubevirt.io/storage.bind.immediate.requested: 'true'

```

```
creationTimestamp: null
name: pvcocp-worker01.cdp.rdu2.scalelab.redhat.com
spec:
  source:
    blank: {}
  storage:
    resources:
      requests:
        storage: 120Gi
  - metadata:
      name: pvcocp-worker01.cdp.rdu2.scalelab.redhat.com-cdrom
    spec:
      source:
        pvc:
          name: discovery-image-cdp
          namespace: cdppvcds
      storage:
        resources:
          requests:
            storage: 2Gi
runStrategy: RerunOnFailure
template:
  metadata:
    annotations:
      vm.kubevirt.io/flavor: small
      vm.kubevirt.io/os: rhel9
      vm.kubevirt.io/workload: server
  creationTimestamp: null
  labels:
    kubevirt.io/domain: pvcocp-worker01.cdp.rdu2.scalelab.redhat.com
    kubevirt.io/size: small
spec:
  architecture: amd64
  domain:
    cpu:
      cores: 1
      sockets: 32
      threads: 1
    devices:
      disks:
        - bootOrder: 1
          disk:
            bus: virtio
            name: rootdisk
        - bootOrder: 2
          cdrom:
```

```

        bus: sata
        name: installation-cdrom
    interfaces:
        - bridge: {}
          model: virtio
          macAddress: '02:00:e4:00:03:03'
          name: eth0
        rng: {}
    features:
        acpi: {}
        smm:
            enabled: true
    firmware:
        bootloader:
            efi: {}
    machine:
        type: pc-q35-rhel9.4.0
    memory:
        guest: 64Gi
    resources: {}
networks:
    - multus:
        networkName: default/localnet-network
        name: eth0
terminationGracePeriodSeconds: 180
volumes:
    - dataVolume:
        name: pvcocp-worker01.cdp.rdu2.scalelab.redhat.com
        name: rootdisk
    - dataVolume:
        name: pvcocp-worker01.cdp.rdu2.scalelab.redhat.com-cdrom
        name: installation-cdrom

```

## pvcocp-worker02.cdp.rdu2.scalelab.redhat.com-vms.yaml

```

apiVersion: kubevirt.io/v1
kind: VirtualMachine
metadata:
  labels:
    app: pvcocp-worker02.cdp.rdu2.scalelab.redhat.com
    kubevirt.io/dynamic-credentials-support: 'true'
    vm.kubevirt.io/template: rhel9-server-small
    vm.kubevirt.io/template.namespace: openshift
    vm.kubevirt.io/template.revision: '1'
    vm.kubevirt.io/template.version: v0.32.2
  name: pvcocp-worker02.cdp.rdu2.scalelab.redhat.com

```

```
namespace: cdppvcds
spec:
  dataVolumeTemplates:
    - apiVersion: cdi.kubevirt.io/v1beta1
      kind: DataVolume
      metadata:
        annotations:
          cdi.kubevirt.io/storage.bind.immediate.requested: 'true'
        creationTimestamp: null
        name: pvcocp-worker02.cdp.rdu2.scalelab.redhat.com
      spec:
        source:
          blank: {}
        storage:
          resources:
            requests:
              storage: 120Gi
    - metadata:
        name: pvcocp-worker02.cdp.rdu2.scalelab.redhat.com-cdrom
      spec:
        source:
          pvc:
            name: discovery-image-cdp
            namespace: cdppvcds
        storage:
          resources:
            requests:
              storage: 2Gi
  runStrategy: RerunOnFailure
  template:
    metadata:
      annotations:
        vm.kubevirt.io/flavor: small
        vm.kubevirt.io/os: rhel9
        vm.kubevirt.io/workload: server
      creationTimestamp: null
      labels:
        kubevirt.io/domain: pvcocp-worker02.cdp.rdu2.scalelab.redhat.com
        kubevirt.io/size: small
    spec:
      architecture: amd64
      domain:
        cpu:
          cores: 1
          sockets: 32
          threads: 1
```

```

devices:
  disks:
    - bootOrder: 1
      disk:
        bus: virtio
        name: rootdisk
    - bootOrder: 2
      cdrom:
        bus: sata
        name: installation-cdrom
  interfaces:
    - bridge: {}
      model: virtio
      macAddress: '02:00:e4:00:03:04'
      name: eth0
  rng: {}
  features:
    acpi: {}
    smm:
      enabled: true
  firmware:
    bootloader:
      efi: {}
  machine:
    type: pc-q35-rhel9.4.0
  memory:
    guest: 64Gi
    resources: {}
  networks:
    - multus:
        networkName: default/localnet-network
        name: eth0
  terminationGracePeriodSeconds: 180
  volumes:
    - dataVolume:
        name: pvcocp-worker02.cdp.rdu2.scalelab.redhat.com
        name: rootdisk
    - dataVolume:
        name: pvcocp-worker02.cdp.rdu2.scalelab.redhat.com-cdrom
        name: installation-cdrom

```

## **pvcocp-worker03.cdp.rdu2.scalelab.redhat.com-vms.yaml**

```

apiVersion: kubevirt.io/v1
kind: VirtualMachine
metadata:

```

```
labels:
  app: pvcocp-worker03.cdp.rdu2.scalelab.redhat.com
  kubevirt.io/dynamic-credentials-support: 'true'
  vm.kubevirt.io/template: rhel9-server-small
  vm.kubevirt.io/template.namespace: openshift
  vm.kubevirt.io/template.revision: '1'
  vm.kubevirt.io/template.version: v0.32.2
  name: pvcocp-worker03.cdp.rdu2.scalelab.redhat.com
  namespace: cdppvcds
spec:
  dataVolumeTemplates:
    - apiVersion: cdi.kubevirt.io/v1beta1
      kind: DataVolume
      metadata:
        annotations:
          cdi.kubevirt.io/storage.bind.immediate.requested: 'true'
          creationTimestamp: null
          name: pvcocp-worker03.cdp.rdu2.scalelab.redhat.com
      spec:
        source:
          blank: {}
        storage:
          resources:
            requests:
              storage: 120Gi
    - metadata:
        name: pvcocp-worker03.cdp.rdu2.scalelab.redhat.com-cdrom
      spec:
        source:
          pvc:
            name: discovery-image-cdp
            namespace: cdppvcds
        storage:
          resources:
            requests:
              storage: 2Gi
  runStrategy: RerunOnFailure
  template:
    metadata:
      annotations:
        vm.kubevirt.io/flavor: small
        vm.kubevirt.io/os: rhel9
        vm.kubevirt.io/workload: server
      creationTimestamp: null
    labels:
      kubevirt.io/domain: pvcocp-worker03.cdp.rdu2.scalelab.redhat.com
```

```
kubevirt.io/size: small
spec:
  architecture: amd64
  domain:
    cpu:
      cores: 1
      sockets: 32
      threads: 1
    devices:
      disks:
        - bootOrder: 1
          disk:
            bus: virtio
            name: rootdisk
        - bootOrder: 2
          cdrom:
            bus: sata
            name: installation-cdrom
    interfaces:
      - bridge: {}
        model: virtio
        macAddress: '02:00:e4:00:03:05'
        name: eth0
    rng: {}
  features:
    acpi: {}
    smm:
      enabled: true
  firmware:
    bootloader:
      efi: {}
  machine:
    type: pc-q35-rhel9.4.0
  memory:
    guest: 64Gi
  resources: {}
networks:
  - multus:
    networkName: default/localnet-network
    name: eth0
terminationGracePeriodSeconds: 180
volumes:
  - dataVolume:
    name: pvcocp-worker03.cdp.rdu2.scalelab.redhat.com
    name: rootdisk
  - dataVolume:
```

```
    name: pvcocp-worker03.cdp.rdu2.scalelab.redhat.com-cdrom
    name: installation-cdrom
```

### pvcocp-worker04.cdp.rdu2.scalelab.redhat.com-vms.yaml

```
apiVersion: kubevirt.io/v1
kind: VirtualMachine
metadata:
  labels:
    app: pvcocp-worker04.cdp.rdu2.scalelab.redhat.com
    kubevirt.io/dynamic-credentials-support: 'true'
    vm.kubevirt.io/template: rhel9-server-small
    vm.kubevirt.io/template.namespace: openshift
    vm.kubevirt.io/template.revision: '1'
    vm.kubevirt.io/template.version: v0.32.2
  name: pvcocp-worker04.cdp.rdu2.scalelab.redhat.com
  namespace: cdppvcds
spec:
  dataVolumeTemplates:
    - apiVersion: cdi.kubevirt.io/v1beta1
      kind: DataVolume
      metadata:
        annotations:
          cdi.kubevirt.io/storage.bind.immediate.requested: 'true'
        creationTimestamp: null
        name: pvcocp-worker04.cdp.rdu2.scalelab.redhat.com
      spec:
        source:
          blank: {}
        storage:
          resources:
            requests:
              storage: 120Gi
    - metadata:
        name: pvcocp-worker04.cdp.rdu2.scalelab.redhat.com-cdrom
      spec:
        source:
          pvc:
            name: discovery-image-cdp
            namespace: cdppvcds
        storage:
          resources:
            requests:
              storage: 2Gi
  runStrategy: RerunOnFailure
  template:
```

```
metadata:
  annotations:
    vm.kubevirt.io/flavor: small
    vm.kubevirt.io/os: rhel9
    vm.kubevirt.io/workload: server
  creationTimestamp: null
  labels:
    kubevirt.io/domain: pvcocp-worker04.cdp.rdu2.scalelab.redhat.com
    kubevirt.io/size: small
spec:
  architecture: amd64
  domain:
    cpu:
      cores: 1
      sockets: 32
      threads: 1
    devices:
      disks:
        - bootOrder: 1
          disk:
            bus: virtio
            name: rootdisk
        - bootOrder: 2
          cdrom:
            bus: sata
            name: installation-cdrom
    interfaces:
      - bridge: {}
        model: virtio
        macAddress: '02:00:e4:00:03:06'
        name: eth0
    rng: {}
  features:
    acpi: {}
    smm:
      enabled: true
  firmware:
    bootloader:
      efi: {}
  machine:
    type: pc-q35-rhel9.4.0
  memory:
    guest: 64Gi
  resources: {}
networks:
  - multus:
```

```

        networkName: default/localnet-network
        name: eth0
terminationGracePeriodSeconds: 180
volumes:
- dataVolume:
    name: pvcocp-worker04.cdp.rdu2.scalelab.redhat.com
    name: rootdisk
- dataVolume:
    name: pvcocp-worker04.cdp.rdu2.scalelab.redhat.com-cdrom
    name: installation-cdrom

```

### pvcocp-worker05.cdp.rdu2.scalelab.redhat.com-vms.yaml

```

apiVersion: kubevirt.io/v1
kind: VirtualMachine
metadata:
labels:
  app: pvcocp-worker05.cdp.rdu2.scalelab.redhat.com
  kubevirt.io/dynamic-credentials-support: 'true'
  vm.kubevirt.io/template: rhel9-server-small
  vm.kubevirt.io/template.namespace: openshift
  vm.kubevirt.io/template.revision: '1'
  vm.kubevirt.io/template.version: v0.32.2
name: pvcocp-worker05.cdp.rdu2.scalelab.redhat.com
namespace: cdppvcds
spec:
dataVolumeTemplates:
- apiVersion: cdi.kubevirt.io/v1beta1
  kind: DataVolume
  metadata:
    annotations:
      cdi.kubevirt.io/storage.bind.immediate.requested: 'true'
      creationTimestamp: null
      name: pvcocp-worker05.cdp.rdu2.scalelab.redhat.com
  spec:
    source:
      blank: {}
    storage:
      resources:
        requests:
          storage: 120Gi
- metadata:
    name: pvcocp-worker05.cdp.rdu2.scalelab.redhat.com-cdrom
  spec:
    source:
      pvc:

```

```
        name: discovery-image-cdp
        namespace: cdppvcds
storage:
  resources:
    requests:
      storage: 2Gi
runStrategy: RerunOnFailure
template:
  metadata:
    annotations:
      vm.kubevirt.io/flavor: small
      vm.kubevirt.io/os: rhel9
      vm.kubevirt.io/workload: server
  creationTimestamp: null
  labels:
    kubevirt.io/domain: pvcocp-worker05.cdp.rdu2.scalelab.redhat.com
    kubevirt.io/size: small
spec:
  architecture: amd64
  domain:
    cpu:
      cores: 1
      sockets: 32
      threads: 1
    devices:
      disks:
        - bootOrder: 1
          disk:
            bus: virtio
            name: rootdisk
        - bootOrder: 2
          cdrom:
            bus: sata
            name: installation-cdrom
    interfaces:
      - bridge: {}
        model: virtio
        macAddress: '02:00:e4:00:03:07'
        name: eth0
    rng: {}
  features:
    acpi: {}
    smm:
      enabled: true
  firmware:
  bootloader:
```

```

        efi: {}
machine:
  type: pc-q35-rhel9.4.0
memory:
  guest: 64Gi
resources: {}
networks:
- multus:
    networkName: default/localnet-network
    name: eth0
terminationGracePeriodSeconds: 180
volumes:
- dataVolume:
    name: pvcocp-worker05.cdp.rdu2.scalelab.redhat.com
    name: rootdisk
- dataVolume:
    name: pvcocp-worker05.cdp.rdu2.scalelab.redhat.com-cdrom
    name: installation-cdrom

```

## pvcocp-worker06.cdp.rdu2.scalelab.redhat.com-vms.yaml

```

apiVersion: kubevirt.io/v1
kind: VirtualMachine
metadata:
  labels:
    app: pvcocp-worker06.cdp.rdu2.scalelab.redhat.com
    kubevirt.io/dynamic-credentials-support: 'true'
    vm.kubevirt.io/template: rhel9-server-small
    vm.kubevirt.io/template.namespace: openshift
    vm.kubevirt.io/template.revision: '1'
    vm.kubevirt.io/template.version: v0.32.2
  name: pvcocp-worker06.cdp.rdu2.scalelab.redhat.com
  namespace: cdppvcds
spec:
  dataVolumeTemplates:
  - apiVersion: cdi.kubevirt.io/v1beta1
    kind: DataVolume
    metadata:
      annotations:
        cdi.kubevirt.io/storage.bind.immediate.requested: 'true'
      creationTimestamp: null
      name: pvcocp-worker06.cdp.rdu2.scalelab.redhat.com
    spec:
      source:
        blank: {}
      storage:

```

```
resources:
  requests:
    storage: 120Gi
- metadata:
    name: pvcocp-worker06.cdp.rdu2.scalelab.redhat.com-cdrom
spec:
  source:
    pvc:
      name: discovery-image-cdp
      namespace: cdppvcds
  storage:
    resources:
      requests:
        storage: 2Gi
runStrategy: RerunOnFailure
template:
  metadata:
    annotations:
      vm.kubevirt.io/flavor: small
      vm.kubevirt.io/os: rhel9
      vm.kubevirt.io/workload: server
  creationTimestamp: null
  labels:
    kubevirt.io/domain: pvcocp-worker06.cdp.rdu2.scalelab.redhat.com
    kubevirt.io/size: small
spec:
  architecture: amd64
  domain:
    cpu:
      cores: 1
      sockets: 32
      threads: 1
    devices:
      disks:
        - bootOrder: 1
          disk:
            bus: virtio
            name: rootdisk
        - bootOrder: 2
          cdrom:
            bus: sata
            name: installation-cdrom
  interfaces:
    - bridge: {}
      model: virtio
      macAddress: '02:00:e4:00:03:08'
```

```

        name: eth0
    rng: {}
features:
    acpi: {}
    smm:
        enabled: true
firmware:
    bootloader:
        efi: {}
machine:
    type: pc-q35-rhel9.4.0
memory:
    guest: 64Gi
resources: {}
networks:
    - multus:
        networkName: default/localnet-network
        name: eth0
terminationGracePeriodSeconds: 180
volumes:
    - dataVolume:
        name: pvcocp-worker06.cdp.rdu2.scalelab.redhat.com
        name: rootdisk
    - dataVolume:
        name: pvcocp-worker06.cdp.rdu2.scalelab.redhat.com-cdrom
        name: installation-cdrom

```

## pvcocp-worker07.cdp.rdu2.scalelab.redhat.com-vms.yaml

```

apiVersion: kubevirt.io/v1
kind: VirtualMachine
metadata:
labels:
    app: pvcocp-worker07.cdp.rdu2.scalelab.redhat.com
    kubevirt.io/dynamic-credentials-support: 'true'
    vm.kubevirt.io/template: rhel9-server-small
    vm.kubevirt.io/template.namespace: openshift
    vm.kubevirt.io/template.revision: '1'
    vm.kubevirt.io/template.version: v0.32.2
name: pvcocp-worker07.cdp.rdu2.scalelab.redhat.com
namespace: cdppvcds
spec:
dataVolumeTemplates:
    - apiVersion: cdi.kubevirt.io/v1beta1
        kind: DataVolume
        metadata:

```

```
annotations:
  cdi.kubevirt.io/storage.bind.immediate.requested: 'true'
creationTimestamp: null
name: pvcocp-worker07.cdp.rdu2.scalelab.redhat.com
spec:
  source:
    blank: {}
  storage:
    resources:
      requests:
        storage: 120Gi
- metadata:
    name: pvcocp-worker07.cdp.rdu2.scalelab.redhat.com-cdrom
  spec:
    source:
      pvc:
        name: discovery-image-cdp
        namespace: cdppvcds
    storage:
      resources:
        requests:
          storage: 2Gi
runStrategy: RerunOnFailure
template:
  metadata:
    annotations:
      vm.kubevirt.io/flavor: small
      vm.kubevirt.io/os: rhel9
      vm.kubevirt.io/workload: server
    creationTimestamp: null
    labels:
      kubevirt.io/domain: pvcocp-worker07.cdp.rdu2.scalelab.redhat.com
      kubevirt.io/size: small
  spec:
    architecture: amd64
    domain:
      cpu:
        cores: 1
        sockets: 32
        threads: 1
      devices:
        disks:
          - bootOrder: 1
            disk:
              bus: virtio
              name: rootdisk
```

```

    - bootOrder: 2
      cdrom:
        bus: sata
        name: installation-cdrom
    interfaces:
      - bridge: {}
        model: virtio
        macAddress: '02:00:e4:00:03:09'
        name: eth0
      rng: {}
    features:
      acpi: {}
      smm:
        enabled: true
    firmware:
      bootloader:
        efi: {}
    machine:
      type: pc-q35-rhel9.4.0
    memory:
      guest: 64Gi
    resources: {}
  networks:
    - multus:
        networkName: default/localnet-network
        name: eth0
  terminationGracePeriodSeconds: 180
  volumes:
    - dataVolume:
        name: pvcocp-worker07.cdp.rdu2.scalelab.redhat.com
        name: roottdisk
    - dataVolume:
        name: pvcocp-worker07.cdp.rdu2.scalelab.redhat.com-cdrom
        name: installation-cdrom

```

### **pvcocp-worker08.cdp.rdu2.scalelab.redhat.com-vms.yaml**

```

apiVersion: kubevirt.io/v1
kind: VirtualMachine
metadata:
  labels:
    app: pvcocp-worker08.cdp.rdu2.scalelab.redhat.com
    kubevirt.io/dynamic-credentials-support: 'true'
    vm.kubevirt.io/template: rhel9-server-small
    vm.kubevirt.io/template.namespace: openshift
    vm.kubevirt.io/template.revision: '1'

```

```
vm.kubevirt.io/template.version: v0.32.2
name: pvcocp-worker08.cdp.rdu2.scalelab.redhat.com
namespace: cdppvcds
spec:
  dataVolumeTemplates:
    - apiVersion: cdi.kubevirt.io/v1beta1
      kind: DataVolume
      metadata:
        annotations:
          cdi.kubevirt.io/storage.bind.immediate.requested: 'true'
        creationTimestamp: null
        name: pvcocp-worker08.cdp.rdu2.scalelab.redhat.com
      spec:
        source:
          blank: {}
        storage:
          resources:
            requests:
              storage: 120Gi
    - metadata:
        name: pvcocp-worker08.cdp.rdu2.scalelab.redhat.com-cdrom
      spec:
        source:
          pvc:
            name: discovery-image-cdp
            namespace: cdppvcds
        storage:
          resources:
            requests:
              storage: 2Gi
  runStrategy: RerunOnFailure
  template:
    metadata:
      annotations:
        vm.kubevirt.io/flavor: small
        vm.kubevirt.io/os: rhel9
        vm.kubevirt.io/workload: server
    creationTimestamp: null
    labels:
      kubevirt.io/domain: pvcocp-worker08.cdp.rdu2.scalelab.redhat.com
      kubevirt.io/size: small
  spec:
    architecture: amd64
    domain:
      cpu:
        cores: 1
```

```

    sockets: 32
    threads: 1
  devices:
    disks:
      - bootOrder: 1
        disk:
          bus: virtio
          name: rootdisk
      - bootOrder: 2
        cdrom:
          bus: sata
          name: installation-cdrom
  interfaces:
    - bridge: {}
      model: virtio
      macAddress: '02:00:e4:00:03:0a'
      name: eth0
    rng: {}
  features:
    acpi: {}
    smm:
      enabled: true
  firmware:
    bootloader:
      efi: {}
  machine:
    type: pc-q35-rhel9.4.0
  memory:
    guest: 64Gi
  resources: {}
  networks:
    - multus:
        networkName: default/localnet-network
        name: eth0
  terminationGracePeriodSeconds: 180
  volumes:
    - dataVolume:
        name: pvcocp-worker08.cdp.rdu2.scalelab.redhat.com
        name: rootdisk
    - dataVolume:
        name: pvcocp-worker08.cdp.rdu2.scalelab.redhat.com-cdrom
        name: installation-cdrom

```

## pvcocp-worker09.cdp.rdu2.scalelab.redhat.com-vms.yaml

apiVersion: kubevirt.io/v1

```

kind: VirtualMachine
metadata:
  labels:
    app: pvcocp-worker09.cdp.rdu2.scalelab.redhat.com
    kubevirt.io/dynamic-credentials-support: 'true'
    vm.kubevirt.io/template: rhel9-server-small
    vm.kubevirt.io/template.namespace: openshift
    vm.kubevirt.io/template.revision: '1'
    vm.kubevirt.io/template.version: v0.32.2
  name: pvcocp-worker09.cdp.rdu2.scalelab.redhat.com
  namespace: cdppvcds
spec:
  dataVolumeTemplates:
    - apiVersion: cdi.kubevirt.io/v1beta1
      kind: DataVolume
      metadata:
        annotations:
          cdi.kubevirt.io/storage.bind.immediate.requested: 'true'
        creationTimestamp: null
        name: pvcocp-worker09.cdp.rdu2.scalelab.redhat.com
      spec:
        source:
          blank: {}
        storage:
          resources:
            requests:
              storage: 120Gi
    - metadata:
        name: pvcocp-worker09.cdp.rdu2.scalelab.redhat.com-cdrom
      spec:
        source:
          pvc:
            name: discovery-image-cdp
            namespace: cdppvcds
        storage:
          resources:
            requests:
              storage: 2Gi
  runStrategy: RerunOnFailure
  template:
    metadata:
      annotations:
        vm.kubevirt.io/flavor: small
        vm.kubevirt.io/os: rhel9
        vm.kubevirt.io/workload: server
    creationTimestamp: null

```

```
labels:
  kubevirt.io/domain: pvcocp-worker09.cdp.rdu2.scalelab.redhat.com
  kubevirt.io/size: small
spec:
  architecture: amd64
  domain:
    cpu:
      cores: 1
      sockets: 32
      threads: 1
    devices:
      disks:
        - bootOrder: 1
          disk:
            bus: virtio
            name: rootdisk
        - bootOrder: 2
          cdrom:
            bus: sata
            name: installation-cdrom
    interfaces:
      - bridge: {}
        model: virtio
        macAddress: '02:00:e4:00:03:0b'
        name: eth0
    rng: {}
  features:
    acpi: {}
    smm:
      enabled: true
  firmware:
    bootloader:
      efi: {}
  machine:
    type: pc-q35-rhel9.4.0
  memory:
    guest: 64Gi
  resources: {}
networks:
  - multus:
      networkName: default/localnet-network
      name: eth0
terminationGracePeriodSeconds: 180
volumes:
  - dataVolume:
      name: pvcocp-worker09.cdp.rdu2.scalelab.redhat.com
```

```
    name: rootdisk
  - dataVolume:
      name: pvcocp-worker09.cdp.rdu2.scalelab.redhat.com-cdrom
      name: installation-cdrom
```

## pvcocp-worker10.cdp.rdu2.scalelab.redhat.com-vms.yaml

```
apiVersion: kubevirt.io/v1
kind: VirtualMachine
metadata:
  labels:
    app: pvcocp-worker10.cdp.rdu2.scalelab.redhat.com
    kubevirt.io/dynamic-credentials-support: 'true'
    vm.kubevirt.io/template: rhel9-server-small
    vm.kubevirt.io/template.namespace: openshift
    vm.kubevirt.io/template.revision: '1'
    vm.kubevirt.io/template.version: v0.32.2
  name: pvcocp-worker10.cdp.rdu2.scalelab.redhat.com
  namespace: cdppvcds
spec:
  dataVolumeTemplates:
    - apiVersion: cdi.kubevirt.io/v1beta1
      kind: DataVolume
      metadata:
        annotations:
          cdi.kubevirt.io/storage.bind.immediate.requested: 'true'
        creationTimestamp: null
        name: pvcocp-worker10.cdp.rdu2.scalelab.redhat.com
      spec:
        source:
          blank: {}
        storage:
          resources:
            requests:
              storage: 120Gi
    - metadata:
        name: pvcocp-worker10.cdp.rdu2.scalelab.redhat.com-cdrom
      spec:
        source:
          pvc:
            name: discovery-image-cdp
            namespace: cdppvcds
        storage:
          resources:
            requests:
              storage: 2Gi
```

```
runStrategy: RerunOnFailure
template:
  metadata:
    annotations:
      vm.kubevirt.io/flavor: small
      vm.kubevirt.io/os: rhel9
      vm.kubevirt.io/workload: server
  creationTimestamp: null
  labels:
    kubevirt.io/domain: pvcocp-worker10.cdp.rdu2.scalelab.redhat.com
    kubevirt.io/size: small
spec:
  architecture: amd64
  domain:
    cpu:
      cores: 1
      sockets: 32
      threads: 1
    devices:
      disks:
        - bootOrder: 1
          disk:
            bus: virtio
            name: rootdisk
        - bootOrder: 2
          cdrom:
            bus: sata
            name: installation-cdrom
    interfaces:
      - bridge: {}
        model: virtio
        macAddress: '02:00:e4:00:03:0c'
        name: eth0
      rng: {}
    features:
      acpi: {}
      smm:
        enabled: true
  firmware:
    bootloader:
      efi: {}
  machine:
    type: pc-q35-rhel9.4.0
  memory:
    guest: 64Gi
  resources: {}
```

```

networks:
  - multus:
      networkName: default/localnet-network
      name: eth0
  terminationGracePeriodSeconds: 180
volumes:
  - dataVolume:
      name: pvcocp-worker10.cdp.rdu2.scalelab.redhat.com
      name: rootdisk
  - dataVolume:
      name: pvcocp-worker10.cdp.rdu2.scalelab.redhat.com-cdrom
      name: installation-cdrom

```

## pvcocp-infra01.cdp.rdu2.scalelab.redhat.com-vms.yaml

```

apiVersion: kubevirt.io/v1
kind: VirtualMachine
metadata:
  labels:
    app: pvcocp-infra01.cdp.rdu2.scalelab.redhat.com
    kubevirt.io/dynamic-credentials-support: 'true'
    vm.kubevirt.io/template: rhel9-server-small
    vm.kubevirt.io/template.namespace: openshift
    vm.kubevirt.io/template.revision: '1'
    vm.kubevirt.io/template.version: v0.32.2
  name: pvcocp-infra01.cdp.rdu2.scalelab.redhat.com
  namespace: cdppvcds
spec:
  dataVolumeTemplates:
    - apiVersion: cdi.kubevirt.io/v1beta1
      kind: DataVolume
      metadata:
        annotations:
          cdi.kubevirt.io/storage.bind.immediate.requested: 'true'
        creationTimestamp: null
        name: pvcocp-infra01.cdp.rdu2.scalelab.redhat.com
      spec:
        source:
          blank: {}
        storage:
          resources:
            requests:
              storage: 120Gi
    - metadata:
        name: pvcocp-infra01.cdp.rdu2.scalelab.redhat.com-cdrom

```

```
spec:
  source:
    pvc:
      name: discovery-image-cdp
      namespace: cdppvcds
  storage:
    resources:
      requests:
        storage: 2Gi
- metadata:
    name: pvcocp-infra01.cdp.rdu2.scalelab.redhat.com-disk1
spec:
  source:
    blank: {}
  storage:
    accessModes:
      - ReadWriteMany
  resources:
    requests:
      storage: 2000Gi
  storageClassName: ocs-storagecluster-ceph-rbd
  volumeMode: Block
runStrategy: RerunOnFailure
template:
  metadata:
    annotations:
      vm.kubevirt.io/flavor: small
      vm.kubevirt.io/os: rhel9
      vm.kubevirt.io/workload: server
    creationTimestamp: null
  labels:
    kubevirt.io/domain: pvcocp-infra01.cdp.rdu2.scalelab.redhat.com
    kubevirt.io/size: small
spec:
  architecture: amd64
  domain:
    cpu:
      cores: 1
      sockets: 32
      threads: 1
    devices:
      disks:
        - bootOrder: 1
          disk:
            bus: virtio
            name: rootdisk
```

```

    - bootOrder: 2
      cdrom:
        bus: sata
        name: installation-cdrom
    - bootOrder: 3
      disk:
        bus: virtio
        name: disk-1
  interfaces:
    - bridge: {}
      model: virtio
      macAddress: '02:00:e4:00:03:0d'
      name: eth0
    rng: {}
  features:
    acpi: {}
    smm:
      enabled: true
  firmware:
    bootloader:
      efi: {}
  machine:
    type: pc-q35-rhel9.4.0
  memory:
    guest: 64Gi
  resources: {}
networks:
  - multus:
      networkName: default/localnet-network
      name: eth0
terminationGracePeriodSeconds: 180
volumes:
  - dataVolume:
      name: pvcocp-infra01.cdp.rdu2.scalelab.redhat.com
      name: rootdisk
  - dataVolume:
      name: pvcocp-infra01.cdp.rdu2.scalelab.redhat.com-cdrom
      name: installation-cdrom
  - dataVolume:
      name: pvcocp-infra01.cdp.rdu2.scalelab.redhat.com-disk1
      name: disk-1

```

## pvcocp-infra02.cdp.rdu2.scalelab.redhat.com-vms.yaml

```

apiVersion: kubevirt.io/v1
kind: VirtualMachine

```

```
metadata:
  labels:
    app: pvcocp-infra02.cdp.rdu2.scalelab.redhat.com
    kubevirt.io/dynamic-credentials-support: 'true'
    vm.kubevirt.io/template: rhel9-server-small
    vm.kubevirt.io/template.namespace: openshift
    vm.kubevirt.io/template.revision: '1'
    vm.kubevirt.io/template.version: v0.32.2
  name: pvcocp-infra02.cdp.rdu2.scalelab.redhat.com
  namespace: cdppvcds
spec:
  dataVolumeTemplates:
    - apiVersion: cdi.kubevirt.io/v1beta1
      kind: DataVolume
      metadata:
        annotations:
          cdi.kubevirt.io/storage.bind.immediate.requested: 'true'
        creationTimestamp: null
        name: pvcocp-infra02.cdp.rdu2.scalelab.redhat.com
      spec:
        source:
          blank: {}
        storage:
          resources:
            requests:
              storage: 120Gi
    - metadata:
        name: pvcocp-infra02.cdp.rdu2.scalelab.redhat.com-cdrom
      spec:
        source:
          pvc:
            name: discovery-image-cdp
            namespace: cdppvcds
        storage:
          resources:
            requests:
              storage: 2Gi
    - metadata:
        name: pvcocp-infra02.cdp.rdu2.scalelab.redhat.com-disk1
      spec:
        source:
          blank: {}
        storage:
          accessModes:
            - ReadWriteMany
        resources:
```

```
    requests:
      storage: 2000Gi
    storageClassName: ocs-storagecluster-ceph-rbd
    volumeMode: Block
  runStrategy: RerunOnFailure
  template:
    metadata:
      annotations:
        vm.kubevirt.io/flavor: small
        vm.kubevirt.io/os: rhel9
        vm.kubevirt.io/workload: server
    creationTimestamp: null
    labels:
      kubevirt.io/domain: pvcocp-infra02.cdp.rdu2.scalelab.redhat.com
      kubevirt.io/size: small
  spec:
    architecture: amd64
    domain:
      cpu:
        cores: 1
        sockets: 32
        threads: 1
      devices:
        disks:
          - bootOrder: 1
            disk:
              bus: virtio
              name: rootdisk
          - bootOrder: 2
            cdrom:
              bus: sata
              name: installation-cdrom
          - bootOrder: 3
            disk:
              bus: virtio
              name: disk-1
      interfaces:
        - bridge: {}
          model: virtio
          macAddress: '02:00:e4:00:03:0e'
          name: eth0
      rng: {}
    features:
      acpi: {}
      smm:
        enabled: true
```

```

firmware:
  bootloader:
    efi: {}
machine:
  type: pc-q35-rhel9.4.0
memory:
  guest: 64Gi
  resources: {}
networks:
- multus:
    networkName: default/localnet-network
    name: eth0
terminationGracePeriodSeconds: 180
volumes:
- dataVolume:
    name: pvcocp-infra02.cdp.rdu2.scalelab.redhat.com
    name: rootdisk
- dataVolume:
    name: pvcocp-infra02.cdp.rdu2.scalelab.redhat.com-cdrom
    name: installation-cdrom
- dataVolume:
    name: pvcocp-infra02.cdp.rdu2.scalelab.redhat.com-disk1
    name: disk-1

```

## pvcocp-infra03.cdp.rdu2.scalelab.redhat.com-vms.yaml

```

apiVersion: kubevirt.io/v1
kind: VirtualMachine
metadata:
  labels:
    app: pvcocp-infra03.cdp.rdu2.scalelab.redhat.com
    kubevirt.io/dynamic-credentials-support: 'true'
    vm.kubevirt.io/template: rhel9-server-small
    vm.kubevirt.io/template.namespace: openshift
    vm.kubevirt.io/template.revision: '1'
    vm.kubevirt.io/template.version: v0.32.2
  name: pvcocp-infra03.cdp.rdu2.scalelab.redhat.com
  namespace: cdppvcds
spec:
  dataVolumeTemplates:
  - apiVersion: cdi.kubevirt.io/v1beta1
    kind: DataVolume
    metadata:
      annotations:
        cdi.kubevirt.io/storage.bind.immediate.requested: 'true'
    creationTimestamp: null

```

```
        name: pvcocp-infra03.cdp.rdu2.scalelab.redhat.com
      spec:
        source:
          blank: {}
        storage:
          resources:
            requests:
              storage: 120Gi
    - metadata:
        name: pvcocp-infra03.cdp.rdu2.scalelab.redhat.com-cdrom
      spec:
        source:
          pvc:
            name: discovery-image-cdp
            namespace: cdppvcds
        storage:
          resources:
            requests:
              storage: 2Gi
    - metadata:
        name: pvcocp-infra03.cdp.rdu2.scalelab.redhat.com-disk1
      spec:
        source:
          blank: {}
        storage:
          accessModes:
            - ReadWriteMany
        resources:
          requests:
            storage: 2000Gi
        storageClassName: ocs-storagecluster-ceph-rbd
        volumeMode: Block
      runStrategy: RerunOnFailure
      template:
        metadata:
          annotations:
            vm.kubevirt.io/flavor: small
            vm.kubevirt.io/os: rhel9
            vm.kubevirt.io/workload: server
        creationTimestamp: null
        labels:
          kubevirt.io/domain: pvcocp-infra03.cdp.rdu2.scalelab.redhat.com
          kubevirt.io/size: small
      spec:
        architecture: amd64
        domain:
```

```
cpu:
  cores: 1
  sockets: 32
  threads: 1
devices:
  disks:
    - bootOrder: 1
      disk:
        bus: virtio
        name: rootdisk
    - bootOrder: 2
      cdrom:
        bus: sata
        name: installation-cdrom
    - bootOrder: 3
      disk:
        bus: virtio
        name: disk-1
interfaces:
  - bridge: {}
    model: virtio
    macAddress: '02:00:e4:00:03:0f'
    name: eth0
  rng: {}
features:
  acpi: {}
  smm:
    enabled: true
firmware:
  bootloader:
    efi: {}
machine:
  type: pc-q35-rhel9.4.0
memory:
  guest: 64Gi
resources: {}
networks:
  - multus:
      networkName: default/localnet-network
      name: eth0
terminationGracePeriodSeconds: 180
volumes:
  - dataVolume:
      name: pvcocp-infra03.cdp.rdu2.scalelab.redhat.com
      name: rootdisk
  - dataVolume:
```

```
        name: pvcocp-infra03.cdp.rdu2.scalelab.redhat.com-cdrom
        name: installation-cdrom
      - dataVolume:
          name: pvcocp-infra03.cdp.rdu2.scalelab.redhat.com-disk1
          name: disk-1
---
```

#### STEP 14: Track the installation progress.

```
[root@bastion ~]# openshift-install --dir ocp-install-dir agent wait-for
install-complete --log-level=debug
```

```
[root@ipaserver ocp-agent-install]# openshift-install --dir ocp-install-dir agent
wait-for install-complete --log-level=debug
DEBUG OpenShift Installer 4.17.42
DEBUG Built from commit f4a8c8c1eead4b425c434d2300c1a9a387789ed9
DEBUG asset directory: ocp-install-dir
DEBUG Loading Agent Config...
DEBUG Using Agent Config Loaded from state file
DEBUG Loading Agent Manifests...
DEBUG Loading Agent PullSecret...
DEBUG Loading Agent Workflow...
DEBUG Using Agent Workflow Loaded from state file
DEBUG Loading Agent Installer ClusterInfo...
DEBUG Loading Agent Workflow...
DEBUG Loading AddNodes Config...
DEBUG Using AddNodes Config Loaded from state file
DEBUG Using Agent Installer ClusterInfo Loaded from state file
DEBUG Loading Install Config...
DEBUG Using Install Config Loaded from state file
DEBUG Using Agent PullSecret Loaded from state file
DEBUG Loading InfraEnv Config...
DEBUG Loading Agent Workflow...
DEBUG Loading Agent Installer ClusterInfo...
DEBUG Loading Install Config...
DEBUG Loading Agent Config...
DEBUG Using InfraEnv Config Loaded from state file
DEBUG Loading NMState Config...
DEBUG Loading Agent Workflow...
DEBUG Loading Agent Installer ClusterInfo...
DEBUG Loading Agent Hosts...
DEBUG Loading Agent Workflow...
DEBUG Loading AddNodes Config...
DEBUG Loading Install Config...
DEBUG Loading Agent Config...
DEBUG Using Agent Hosts Loaded from state file
DEBUG Loading Install Config...
DEBUG Using NMState Config Loaded from state file
DEBUG Loading AgentClusterInstall Config...
DEBUG Loading Agent Workflow...
DEBUG Loading Install Config...
DEBUG Loading Agent Hosts...
DEBUG Using AgentClusterInstall Config Loaded from state file
DEBUG Loading ClusterDeployment Config...
```

```

DEBUG Loading Agent Workflow...
DEBUG Loading Install Config...
DEBUG Using ClusterDeployment Config Loaded from state file
DEBUG Loading ClusterImageSet Config...
DEBUG Loading Agent Workflow...
DEBUG Loading Agent Installer ClusterInfo...
DEBUG Loading Release Image Pull Spec...
DEBUG Using Release Image Pull Spec Loaded from state file
DEBUG Loading Install Config...
DEBUG Using ClusterImageSet Config Loaded from state file
DEBUG Using Agent Manifests Loaded from state file
DEBUG Loading Install Config...
DEBUG Loading SSH Key...
DEBUG Loading Base Domain...
DEBUG Loading Platform...
DEBUG Loading Cluster Name...
DEBUG Loading Base Domain...
DEBUG Loading Platform...
DEBUG Loading Pull Secret...
DEBUG Loading Platform...
DEBUG Loading Agent Hosts...
DEBUG RendezvousIP from the AgentConfig 10.1.49.200
DEBUG Loading Agent Installer API Auth Config...
DEBUG Loading Agent Installer InfraEnv ID...
DEBUG Using Agent Installer InfraEnv ID Loaded from state file
DEBUG Loading Agent Workflow...
DEBUG Using Agent Workflow Loaded from state file
DEBUG Loading AddNodes Config...
DEBUG Using AddNodes Config Loaded from state file
DEBUG Using Agent Installer API Auth Config Loaded from state file
DEBUG Agent Rest API Initialized
INFO Cluster is not ready for install. Check validations
WARNING Cluster validation: The cluster has hosts that are not ready to install.
DEBUG Cluster validation: The cluster has the exact amount of dedicated control plane nodes.
DEBUG Cluster validation: API virtual IPs are defined.
DEBUG Cluster validation: api vips 10.1.49.3 belongs to the Machine CIDR and is not in use.
DEBUG Cluster validation: The Cluster Network CIDR is defined.
DEBUG Cluster validation: The base domain is defined.
DEBUG Cluster validation: Ingress virtual IPs are defined.
DEBUG Cluster validation: ingress vips 10.1.49.4 belongs to the Machine CIDR and is not in use.
DEBUG Cluster validation: The Machine Network CIDR is defined.
DEBUG Cluster validation: The Cluster Machine CIDR is equivalent to the calculated CIDR.
DEBUG Cluster validation: The Cluster Network prefix is valid.
DEBUG Cluster validation: The cluster has a valid network type
DEBUG Cluster validation: Same address families for all networks.
DEBUG Cluster validation: No CIDRS are overlapping.
DEBUG Cluster validation: No ntp problems found
DEBUG Cluster validation: The Service Network CIDR is defined.
DEBUG Cluster validation: cnv is disabled
DEBUG Cluster validation: lso is disabled
DEBUG Cluster validation: lvm is disabled
DEBUG Cluster validation: mce is disabled
DEBUG Cluster validation: odf is disabled
DEBUG Cluster validation: Platform requirements satisfied
DEBUG Cluster validation: The pull secret is set.
DEBUG Host worker08.cdp.rdu2.scalelab.redhat.com validation: Valid inventory exists for the host
DEBUG Host worker08.cdp.rdu2.scalelab.redhat.com validation: Sufficient CPU cores
DEBUG Host worker08.cdp.rdu2.scalelab.redhat.com validation: Sufficient minimum RAM
DEBUG Host worker08.cdp.rdu2.scalelab.redhat.com validation: Sufficient disk capacity
DEBUG Host worker08.cdp.rdu2.scalelab.redhat.com validation: Sufficient CPU cores for role worker
DEBUG Host worker08.cdp.rdu2.scalelab.redhat.com validation: Sufficient RAM for role worker
DEBUG Host worker08.cdp.rdu2.scalelab.redhat.com validation: Hostname worker08.cdp.rdu2.scalelab.redhat.com is unique in cluster
DEBUG Host worker08.cdp.rdu2.scalelab.redhat.com validation: Hostname worker08.cdp.rdu2.scalelab.redhat.com is allowed
DEBUG Host worker08.cdp.rdu2.scalelab.redhat.com validation: Speed of installation disk has not yet been measured
DEBUG Host worker08.cdp.rdu2.scalelab.redhat.com validation: Host is compatible with cluster platform baremetal
DEBUG Host worker08.cdp.rdu2.scalelab.redhat.com validation: Host agent is compatible with the service
DEBUG Host worker08.cdp.rdu2.scalelab.redhat.com validation: No request to skip formatting of the installation disk
DEBUG Host worker08.cdp.rdu2.scalelab.redhat.com validation: All disks that have skipped formatting are present in the host inventory
DEBUG Host worker08.cdp.rdu2.scalelab.redhat.com validation: Host is connected
DEBUG Host worker08.cdp.rdu2.scalelab.redhat.com validation: Media device is connected

```

```

DEBUG Host worker08.cdp.rdu2.scalelab.redhat.com validation: Machine Network CIDR is defined
DEBUG Host worker08.cdp.rdu2.scalelab.redhat.com validation: Host belongs to all machine network CIDRs
DEBUG Host worker08.cdp.rdu2.scalelab.redhat.com validation: Host has connectivity to the majority of hosts in the cluster
DEBUG Host worker08.cdp.rdu2.scalelab.redhat.com validation: Platform OpenShift Virtualization is allowed
WARNING Host worker08.cdp.rdu2.scalelab.redhat.com validation: Host couldn't synchronize with any NTP server
DEBUG Host worker08.cdp.rdu2.scalelab.redhat.com validation: Host clock is synchronized with service
DEBUG Host worker08.cdp.rdu2.scalelab.redhat.com validation: All required container images were either pulled successfully or no attempt was made to pull them
DEBUG Host worker08.cdp.rdu2.scalelab.redhat.com validation: Network latency requirement has been satisfied.
DEBUG Host worker08.cdp.rdu2.scalelab.redhat.com validation: Packet loss requirement has been satisfied.
DEBUG Host worker08.cdp.rdu2.scalelab.redhat.com validation: Host has been configured with at least one default route.
DEBUG Host worker08.cdp.rdu2.scalelab.redhat.com validation: Domain name resolution for the api.cdp.rdu2.scalelab.redhat.com domain was successful or not required
DEBUG Host worker08.cdp.rdu2.scalelab.redhat.com validation: Domain name resolution for the api-int.cdp.rdu2.scalelab.redhat.com domain was successful or not required
DEBUG Host worker08.cdp.rdu2.scalelab.redhat.com validation: Domain name resolution for the *.apps.cdp.rdu2.scalelab.redhat.com domain was successful or not required
DEBUG Host worker08.cdp.rdu2.scalelab.redhat.com validation: DNS wildcard check was successful
DEBUG Host worker08.cdp.rdu2.scalelab.redhat.com validation: Host subnets are not overlapping
DEBUG Host worker08.cdp.rdu2.scalelab.redhat.com validation: No IP collisions were detected by host 93bed28b-df8f-558c-81f9-6422039241dd
DEBUG Host worker08.cdp.rdu2.scalelab.redhat.com validation: cnv is disabled
DEBUG Host worker08.cdp.rdu2.scalelab.redhat.com validation: lso is disabled
DEBUG Host worker08.cdp.rdu2.scalelab.redhat.com validation: lvm is disabled
DEBUG Host worker08.cdp.rdu2.scalelab.redhat.com validation: mce is disabled
DEBUG Host worker08.cdp.rdu2.scalelab.redhat.com validation: odf is disabled
DEBUG Host master02.cdp.rdu2.scalelab.redhat.com validation: Valid inventory exists for the host
DEBUG Host master02.cdp.rdu2.scalelab.redhat.com validation: Sufficient CPU cores
DEBUG Host master02.cdp.rdu2.scalelab.redhat.com validation: Sufficient minimum RAM
DEBUG Host master02.cdp.rdu2.scalelab.redhat.com validation: Sufficient disk capacity
DEBUG Host master02.cdp.rdu2.scalelab.redhat.com validation: Sufficient CPU cores for role master
DEBUG Host master02.cdp.rdu2.scalelab.redhat.com validation: Sufficient RAM for role master
DEBUG Host master02.cdp.rdu2.scalelab.redhat.com validation: Hostname master02.cdp.rdu2.scalelab.redhat.com is unique in cluster
DEBUG Host master02.cdp.rdu2.scalelab.redhat.com validation: Hostname master02.cdp.rdu2.scalelab.redhat.com is allowed
DEBUG Host master02.cdp.rdu2.scalelab.redhat.com validation: Speed of installation disk has not yet been measured
DEBUG Host master02.cdp.rdu2.scalelab.redhat.com validation: Host is compatible with cluster platform baremetal
DEBUG Host master02.cdp.rdu2.scalelab.redhat.com validation: Host agent is compatible with the service
DEBUG Host master02.cdp.rdu2.scalelab.redhat.com validation: No request to skip formatting of the installation disk
DEBUG Host master02.cdp.rdu2.scalelab.redhat.com validation: All disks that have skipped formatting are present in the host inventory
DEBUG Host master02.cdp.rdu2.scalelab.redhat.com validation: Host is connected
DEBUG Host master02.cdp.rdu2.scalelab.redhat.com validation: Media device is connected
DEBUG Host master02.cdp.rdu2.scalelab.redhat.com validation: Machine Network CIDR is defined
DEBUG Host master02.cdp.rdu2.scalelab.redhat.com validation: Host belongs to all machine network CIDRs
DEBUG Host master02.cdp.rdu2.scalelab.redhat.com validation: Host has connectivity to the majority of hosts in the cluster
DEBUG Host master02.cdp.rdu2.scalelab.redhat.com validation: Platform OpenShift Virtualization is allowed
WARNING Host master02.cdp.rdu2.scalelab.redhat.com validation: Host couldn't synchronize with any NTP server
DEBUG Host master02.cdp.rdu2.scalelab.redhat.com validation: Host clock is synchronized with service
DEBUG Host master02.cdp.rdu2.scalelab.redhat.com validation: All required container images were either pulled successfully or no attempt was made to pull them
DEBUG Host master02.cdp.rdu2.scalelab.redhat.com validation: Network latency requirement has been satisfied.
DEBUG Host master02.cdp.rdu2.scalelab.redhat.com validation: Packet loss requirement has been satisfied.
DEBUG Host master02.cdp.rdu2.scalelab.redhat.com validation: Host has been configured with at least one default route.
DEBUG Host master02.cdp.rdu2.scalelab.redhat.com validation: Domain name resolution for the api.cdp.rdu2.scalelab.redhat.com domain was successful or not required
DEBUG Host master02.cdp.rdu2.scalelab.redhat.com validation: Domain name resolution for the api-int.cdp.rdu2.scalelab.redhat.com domain was successful or not required
DEBUG Host master02.cdp.rdu2.scalelab.redhat.com validation: Domain name resolution for the *.apps.cdp.rdu2.scalelab.redhat.com domain was successful or not required
DEBUG Host master02.cdp.rdu2.scalelab.redhat.com validation: DNS wildcard check was successful
DEBUG Host master02.cdp.rdu2.scalelab.redhat.com validation: Host subnets are not overlapping
DEBUG Host master02.cdp.rdu2.scalelab.redhat.com validation: No IP collisions were detected by host 64c50e5e-5bfd-5a20-b59b-a03a9a1458a1
DEBUG Host master02.cdp.rdu2.scalelab.redhat.com validation: cnv is disabled
DEBUG Host master02.cdp.rdu2.scalelab.redhat.com validation: lso is disabled
DEBUG Host master02.cdp.rdu2.scalelab.redhat.com validation: lvm is disabled
DEBUG Host master02.cdp.rdu2.scalelab.redhat.com validation: mce is disabled
DEBUG Host master02.cdp.rdu2.scalelab.redhat.com validation: odf is disabled
DEBUG Host worker02.cdp.rdu2.scalelab.redhat.com validation: Valid inventory exists for the host

```

```
DEBUG Host worker02.cdp.rdu2.scalelab.redhat.com validation: Sufficient CPU cores
DEBUG Host worker02.cdp.rdu2.scaleLab.redhat.com validation: Sufficient minimum RAM
DEBUG Host worker02.cdp.rdu2.scalelab.redhat.com validation: Sufficient disk capacity
DEBUG Host worker02.cdp.rdu2.scalelab.redhat.com validation: Sufficient CPU cores for role worker
DEBUG Host worker02.cdp.rdu2.scalelab.redhat.com validation: Sufficient RAM for role worker
DEBUG Host worker02.cdp.rdu2.scalelab.redhat.com validation: Hostname worker02.cdp.rdu2.scalelab.redhat.com is unique in cluster
DEBUG Host worker02.cdp.rdu2.scalelab.redhat.com validation: Hostname worker02.cdp.rdu2.scalelab.redhat.com is allowed
DEBUG Host worker02.cdp.rdu2.scalelab.redhat.com validation: Speed of installation disk has not yet been measured
DEBUG Host worker02.cdp.rdu2.scalelab.redhat.com validation: Host is compatible with cluster platform baremetal
DEBUG Host worker02.cdp.rdu2.scalelab.redhat.com validation: Host agent is compatible with the service
DEBUG Host worker02.cdp.rdu2.scalelab.redhat.com validation: No request to skip formatting of the installation disk
DEBUG Host worker02.cdp.rdu2.scalelab.redhat.com validation: All disks that have skipped formatting are present in the host inventory
DEBUG Host worker02.cdp.rdu2.scalelab.redhat.com validation: Host is connected
DEBUG Host worker02.cdp.rdu2.scalelab.redhat.com validation: Media device is connected
DEBUG Host worker02.cdp.rdu2.scalelab.redhat.com validation: Machine Network CIDR is defined
DEBUG Host worker02.cdp.rdu2.scalelab.redhat.com validation: Host belongs to all machine network CIDRs
DEBUG Host worker02.cdp.rdu2.scalelab.redhat.com validation: Host has connectivity to the majority of hosts in the cluster
DEBUG Host worker02.cdp.rdu2.scaleLab.redhat.com validation: Platform OpenShift Virtualization is allowed
WARNING Host worker02.cdp.rdu2.scalelab.redhat.com validation: Host couldn't synchronize with any NTP server
DEBUG Host worker02.cdp.rdu2.scalelab.redhat.com validation: Host clock is synchronized with service
DEBUG Host worker02.cdp.rdu2.scalelab.redhat.com validation: All required container images were either pulled successfully or no attempt was made to pull them
DEBUG Host worker02.cdp.rdu2.scalelab.redhat.com validation: Network latency requirement has been satisfied.
DEBUG Host worker02.cdp.rdu2.scalelab.redhat.com validation: Packet loss requirement has been satisfied.
DEBUG Host worker02.cdp.rdu2.scalelab.redhat.com validation: Host has been configured with at least one default route.
DEBUG Host worker02.cdp.rdu2.scalelab.redhat.com validation: Domain name resolution for the api.cdp.rdu2.scalelab.redhat.com domain was successful or not required
DEBUG Host worker02.cdp.rdu2.scalelab.redhat.com validation: Domain name resolution for the api-int.cdp.rdu2.scalelab.redhat.com domain was successful or not required
DEBUG Host worker02.cdp.rdu2.scalelab.redhat.com validation: Domain name resolution for the *.apps.cdp.rdu2.scalelab.redhat.com domain was successful or not required
DEBUG Host worker02.cdp.rdu2.scalelab.redhat.com validation: DNS wildcard check was successful
DEBUG Host worker02.cdp.rdu2.scalelab.redhat.com validation: Host subnets are not overlapping
DEBUG Host worker02.cdp.rdu2.scalelab.redhat.com validation: No IP collisions were detected by host 00089653-134d-5a4c-bc25-e3b77da8fbec
DEBUG Host worker02.cdp.rdu2.scalelab.redhat.com validation: cnv is disabled
DEBUG Host worker02.cdp.rdu2.scalelab.redhat.com validation: lso is disabled
DEBUG Host worker02.cdp.rdu2.scalelab.redhat.com validation: lvm is disabled
DEBUG Host worker02.cdp.rdu2.scalelab.redhat.com validation: mce is disabled
DEBUG Host worker02.cdp.rdu2.scalelab.redhat.com validation: odf is disabled
DEBUG Host master01.cdp.rdu2.scalelab.redhat.com validation: Valid inventory exists for the host
DEBUG Host master01.cdp.rdu2.scalelab.redhat.com validation: Sufficient CPU cores
DEBUG Host master01.cdp.rdu2.scalelab.redhat.com validation: Sufficient minimum RAM
DEBUG Host master01.cdp.rdu2.scalelab.redhat.com validation: Sufficient disk capacity
DEBUG Host master01.cdp.rdu2.scalelab.redhat.com validation: Sufficient CPU cores for role master
DEBUG Host master01.cdp.rdu2.scalelab.redhat.com validation: Sufficient RAM for role master
DEBUG Host master01.cdp.rdu2.scalelab.redhat.com validation: Hostname master01.cdp.rdu2.scalelab.redhat.com is unique in cluster
DEBUG Host master01.cdp.rdu2.scalelab.redhat.com validation: Hostname master01.cdp.rdu2.scalelab.redhat.com is allowed
DEBUG Host master01.cdp.rdu2.scalelab.redhat.com validation: Speed of installation disk has not yet been measured
DEBUG Host master01.cdp.rdu2.scalelab.redhat.com validation: Host is compatible with cluster platform baremetal
DEBUG Host master01.cdp.rdu2.scalelab.redhat.com validation: Host agent is compatible with the service
DEBUG Host master01.cdp.rdu2.scalelab.redhat.com validation: No request to skip formatting of the installation disk
DEBUG Host master01.cdp.rdu2.scalelab.redhat.com validation: All disks that have skipped formatting are present in the host inventory
DEBUG Host master01.cdp.rdu2.scalelab.redhat.com validation: Host is connected
DEBUG Host master01.cdp.rdu2.scalelab.redhat.com validation: Media device is connected
DEBUG Host master01.cdp.rdu2.scalelab.redhat.com validation: Machine Network CIDR is defined
DEBUG Host master01.cdp.rdu2.scalelab.redhat.com validation: Host belongs to all machine network CIDRs
DEBUG Host master01.cdp.rdu2.scalelab.redhat.com validation: Host has connectivity to the majority of hosts in the cluster
DEBUG Host master01.cdp.rdu2.scalelab.redhat.com validation: Platform OpenShift Virtualization is allowed
WARNING Host master01.cdp.rdu2.scalelab.redhat.com validation: Host couldn't synchronize with any NTP server
DEBUG Host master01.cdp.rdu2.scalelab.redhat.com validation: Host clock is synchronized with service
DEBUG Host master01.cdp.rdu2.scalelab.redhat.com validation: All required container images were either pulled successfully or no attempt was made to pull them
DEBUG Host master01.cdp.rdu2.scalelab.redhat.com validation: Network latency requirement has been satisfied.
DEBUG Host master01.cdp.rdu2.scalelab.redhat.com validation: Packet loss requirement has been satisfied.
DEBUG Host master01.cdp.rdu2.scalelab.redhat.com validation: Host has been configured with at least one default route.
```

```
DEBUG Host master01.cdp.rdu2.scalelab.redhat.com validation: Domain name resolution for the
api.cdp.rdu2.scalelab.redhat.com domain was successful or not required
DEBUG Host master01.cdp.rdu2.scalelab.redhat.com validation: Domain name resolution for the
api-int.cdp.rdu2.scalelab.redhat.com domain was successful or not required
DEBUG Host master01.cdp.rdu2.scalelab.redhat.com validation: Domain name resolution for the
*.apps.cdp.rdu2.scalelab.redhat.com domain was successful or not required
DEBUG Host master01.cdp.rdu2.scalelab.redhat.com validation: DNS wildcard check was successful
DEBUG Host master01.cdp.rdu2.scalelab.redhat.com validation: Host subnets are not overlapping
DEBUG Host master01.cdp.rdu2.scalelab.redhat.com validation: No IP collisions were detected by host
462a303e-0cb3-5fe4-a266-a580b4fcfa76
DEBUG Host master01.cdp.rdu2.scalelab.redhat.com validation: cnv is disabled
DEBUG Host master01.cdp.rdu2.scalelab.redhat.com validation: lso is disabled
DEBUG Host master01.cdp.rdu2.scalelab.redhat.com validation: lvm is disabled
DEBUG Host master01.cdp.rdu2.scalelab.redhat.com validation: mce is disabled
DEBUG Host master01.cdp.rdu2.scalelab.redhat.com validation: odf is disabled
DEBUG Host worker06.cdp.rdu2.scalelab.redhat.com validation: Valid inventory exists for the host
DEBUG Host worker06.cdp.rdu2.scalelab.redhat.com validation: Sufficient CPU cores
DEBUG Host worker06.cdp.rdu2.scalelab.redhat.com validation: Sufficient minimum RAM
DEBUG Host worker06.cdp.rdu2.scalelab.redhat.com validation: Sufficient disk capacity
DEBUG Host worker06.cdp.rdu2.scalelab.redhat.com validation: Sufficient CPU cores for role worker
DEBUG Host worker06.cdp.rdu2.scalelab.redhat.com validation: Sufficient RAM for role worker
DEBUG Host worker06.cdp.rdu2.scalelab.redhat.com validation: Hostname worker06.cdp.rdu2.scalelab.redhat.com is unique in
cluster
DEBUG Host worker06.cdp.rdu2.scalelab.redhat.com validation: Hostname worker06.cdp.rdu2.scalelab.redhat.com is allowed
DEBUG Host worker06.cdp.rdu2.scalelab.redhat.com validation: Speed of installation disk has not yet been measured
DEBUG Host worker06.cdp.rdu2.scalelab.redhat.com validation: Host is compatible with cluster platform baremetal
DEBUG Host worker06.cdp.rdu2.scalelab.redhat.com validation: Host agent is compatible with the service
DEBUG Host worker06.cdp.rdu2.scalelab.redhat.com validation: No request to skip formatting of the installation disk
DEBUG Host worker06.cdp.rdu2.scalelab.redhat.com validation: All disks that have skipped formatting are present in the
host inventory
DEBUG Host worker06.cdp.rdu2.scalelab.redhat.com validation: Host is connected
DEBUG Host worker06.cdp.rdu2.scalelab.redhat.com validation: Media device is connected
DEBUG Host worker06.cdp.rdu2.scalelab.redhat.com validation: Machine Network CIDR is defined
DEBUG Host worker06.cdp.rdu2.scalelab.redhat.com validation: Host belongs to all machine network CIDRS
DEBUG Host worker06.cdp.rdu2.scalelab.redhat.com validation: Host has connectivity to the majority of hosts in the
cluster
DEBUG Host worker06.cdp.rdu2.scalelab.redhat.com validation: Platform OpenShift Virtualization is allowed
WARNING Host worker06.cdp.rdu2.scalelab.redhat.com validation: Host couldn't synchronize with any NTP server
DEBUG Host worker06.cdp.rdu2.scalelab.redhat.com validation: Host clock is synchronized with service
DEBUG Host worker06.cdp.rdu2.scalelab.redhat.com validation: All required container images were either pulled
successfully or no attempt was made to pull them
DEBUG Host worker06.cdp.rdu2.scalelab.redhat.com validation: Network latency requirement has been satisfied.
DEBUG Host worker06.cdp.rdu2.scalelab.redhat.com validation: Packet loss requirement has been satisfied.
DEBUG Host worker06.cdp.rdu2.scalelab.redhat.com validation: Host has been configured with at least one default route.
DEBUG Host worker06.cdp.rdu2.scalelab.redhat.com validation: Domain name resolution for the
api.cdp.rdu2.scalelab.redhat.com domain was successful or not required
DEBUG Host worker06.cdp.rdu2.scalelab.redhat.com validation: Domain name resolution for the
api-int.cdp.rdu2.scalelab.redhat.com domain was successful or not required
DEBUG Host worker06.cdp.rdu2.scalelab.redhat.com validation: Domain name resolution for the
*.apps.cdp.rdu2.scalelab.redhat.com domain was successful or not required
DEBUG Host worker06.cdp.rdu2.scalelab.redhat.com validation: DNS wildcard check was successful
DEBUG Host worker06.cdp.rdu2.scalelab.redhat.com validation: Host subnets are not overlapping
DEBUG Host worker06.cdp.rdu2.scalelab.redhat.com validation: No IP collisions were detected by host
eb194bfd-9bbb-5e78-852b-b9649fe65211
DEBUG Host worker06.cdp.rdu2.scalelab.redhat.com validation: cnv is disabled
DEBUG Host worker06.cdp.rdu2.scalelab.redhat.com validation: lso is disabled
DEBUG Host worker06.cdp.rdu2.scalelab.redhat.com validation: lvm is disabled
DEBUG Host worker06.cdp.rdu2.scalelab.redhat.com validation: mce is disabled
DEBUG Host worker06.cdp.rdu2.scalelab.redhat.com validation: odf is disabled
DEBUG Host worker05.cdp.rdu2.scalelab.redhat.com validation: Valid inventory exists for the host
DEBUG Host worker05.cdp.rdu2.scalelab.redhat.com validation: Sufficient CPU cores
DEBUG Host worker05.cdp.rdu2.scalelab.redhat.com validation: Sufficient minimum RAM
DEBUG Host worker05.cdp.rdu2.scalelab.redhat.com validation: Sufficient disk capacity
DEBUG Host worker05.cdp.rdu2.scalelab.redhat.com validation: Sufficient CPU cores for role worker
DEBUG Host worker05.cdp.rdu2.scalelab.redhat.com validation: Sufficient RAM for role worker
DEBUG Host worker05.cdp.rdu2.scalelab.redhat.com validation: Hostname worker05.cdp.rdu2.scalelab.redhat.com is unique in
cluster
DEBUG Host worker05.cdp.rdu2.scalelab.redhat.com validation: Hostname worker05.cdp.rdu2.scalelab.redhat.com is allowed
DEBUG Host worker05.cdp.rdu2.scalelab.redhat.com validation: Speed of installation disk has not yet been measured
DEBUG Host worker05.cdp.rdu2.scalelab.redhat.com validation: Host is compatible with cluster platform baremetal
DEBUG Host worker05.cdp.rdu2.scalelab.redhat.com validation: Host agent is compatible with the service
DEBUG Host worker05.cdp.rdu2.scalelab.redhat.com validation: No request to skip formatting of the installation disk
```

```
DEBUG Host worker05.cdp.rdu2.scalelab.redhat.com validation: All disks that have skipped formatting are present in the host inventory
DEBUG Host worker05.cdp.rdu2.scalelab.redhat.com validation: Host is connected
DEBUG Host worker05.cdp.rdu2.scalelab.redhat.com validation: Media device is connected
DEBUG Host worker05.cdp.rdu2.scalelab.redhat.com validation: Machine Network CIDR is defined
DEBUG Host worker05.cdp.rdu2.scalelab.redhat.com validation: Host belongs to all machine network CIDRs
DEBUG Host worker05.cdp.rdu2.scalelab.redhat.com validation: Host has connectivity to the majority of hosts in the cluster
DEBUG Host worker05.cdp.rdu2.scalelab.redhat.com validation: Platform OpenShift Virtualization is allowed
WARNING Host worker05.cdp.rdu2.scalelab.redhat.com validation: Host couldn't synchronize with any NTP server
DEBUG Host worker05.cdp.rdu2.scalelab.redhat.com validation: Host clock is synchronized with service
DEBUG Host worker05.cdp.rdu2.scalelab.redhat.com validation: All required container images were either pulled successfully or no attempt was made to pull them
DEBUG Host worker05.cdp.rdu2.scalelab.redhat.com validation: Network latency requirement has been satisfied.
DEBUG Host worker05.cdp.rdu2.scalelab.redhat.com validation: Packet loss requirement has been satisfied.
DEBUG Host worker05.cdp.rdu2.scalelab.redhat.com validation: Host has been configured with at least one default route.
DEBUG Host worker05.cdp.rdu2.scalelab.redhat.com validation: Domain name resolution for the api.cdp.rdu2.scalelab.redhat.com domain was successful or not required
DEBUG Host worker05.cdp.rdu2.scalelab.redhat.com validation: Domain name resolution for the api-int.cdp.rdu2.scalelab.redhat.com domain was successful or not required
DEBUG Host worker05.cdp.rdu2.scalelab.redhat.com validation: Domain name resolution for the *.apps.cdp.rdu2.scalelab.redhat.com domain was successful or not required
DEBUG Host worker05.cdp.rdu2.scalelab.redhat.com validation: DNS wildcard check was successful
DEBUG Host worker05.cdp.rdu2.scalelab.redhat.com validation: Host subnets are not overlapping
DEBUG Host worker05.cdp.rdu2.scalelab.redhat.com validation: No IP collisions were detected by host f7b7f1b9-817a-5e9e-85ee-4d05d0d8f3be
DEBUG Host worker05.cdp.rdu2.scalelab.redhat.com validation: cnv is disabled
DEBUG Host worker05.cdp.rdu2.scalelab.redhat.com validation: lso is disabled
DEBUG Host worker05.cdp.rdu2.scalelab.redhat.com validation: lvm is disabled
DEBUG Host worker05.cdp.rdu2.scalelab.redhat.com validation: mce is disabled
DEBUG Host worker05.cdp.rdu2.scalelab.redhat.com validation: odf is disabled
DEBUG Host worker04.cdp.rdu2.scalelab.redhat.com validation: Valid inventory exists for the host
DEBUG Host worker04.cdp.rdu2.scalelab.redhat.com validation: Sufficient CPU cores
DEBUG Host worker04.cdp.rdu2.scalelab.redhat.com validation: Sufficient minimum RAM
DEBUG Host worker04.cdp.rdu2.scalelab.redhat.com validation: Sufficient disk capacity
DEBUG Host worker04.cdp.rdu2.scalelab.redhat.com validation: Sufficient CPU cores for role worker
DEBUG Host worker04.cdp.rdu2.scalelab.redhat.com validation: Sufficient RAM for role worker
DEBUG Host worker04.cdp.rdu2.scalelab.redhat.com validation: Hostname worker04.cdp.rdu2.scalelab.redhat.com is unique in cluster
DEBUG Host worker04.cdp.rdu2.scalelab.redhat.com validation: Hostname worker04.cdp.rdu2.scalelab.redhat.com is allowed
DEBUG Host worker04.cdp.rdu2.scalelab.redhat.com validation: Speed of installation disk has not yet been measured
DEBUG Host worker04.cdp.rdu2.scalelab.redhat.com validation: Host is compatible with cluster platform baremetal
DEBUG Host worker04.cdp.rdu2.scalelab.redhat.com validation: Host agent is compatible with the service
DEBUG Host worker04.cdp.rdu2.scalelab.redhat.com validation: No request to skip formatting of the installation disk
DEBUG Host worker04.cdp.rdu2.scalelab.redhat.com validation: All disks that have skipped formatting are present in the host inventory
DEBUG Host worker04.cdp.rdu2.scalelab.redhat.com validation: Host is connected
DEBUG Host worker04.cdp.rdu2.scalelab.redhat.com validation: Media device is connected
DEBUG Host worker04.cdp.rdu2.scalelab.redhat.com validation: Machine Network CIDR is defined
DEBUG Host worker04.cdp.rdu2.scalelab.redhat.com validation: Host belongs to all machine network CIDRs
DEBUG Host worker04.cdp.rdu2.scalelab.redhat.com validation: Host has connectivity to the majority of hosts in the cluster
DEBUG Host worker04.cdp.rdu2.scalelab.redhat.com validation: Platform OpenShift Virtualization is allowed
WARNING Host worker04.cdp.rdu2.scalelab.redhat.com validation: Host couldn't synchronize with any NTP server
DEBUG Host worker04.cdp.rdu2.scalelab.redhat.com validation: Host clock is synchronized with service
DEBUG Host worker04.cdp.rdu2.scalelab.redhat.com validation: All required container images were either pulled successfully or no attempt was made to pull them
DEBUG Host worker04.cdp.rdu2.scalelab.redhat.com validation: Network latency requirement has been satisfied.
DEBUG Host worker04.cdp.rdu2.scalelab.redhat.com validation: Packet loss requirement has been satisfied.
DEBUG Host worker04.cdp.rdu2.scalelab.redhat.com validation: Host has been configured with at least one default route.
DEBUG Host worker04.cdp.rdu2.scalelab.redhat.com validation: Domain name resolution for the api.cdp.rdu2.scalelab.redhat.com domain was successful or not required
DEBUG Host worker04.cdp.rdu2.scalelab.redhat.com validation: Domain name resolution for the api-int.cdp.rdu2.scalelab.redhat.com domain was successful or not required
DEBUG Host worker04.cdp.rdu2.scalelab.redhat.com validation: Domain name resolution for the *.apps.cdp.rdu2.scalelab.redhat.com domain was successful or not required
DEBUG Host worker04.cdp.rdu2.scalelab.redhat.com validation: DNS wildcard check was successful
DEBUG Host worker04.cdp.rdu2.scalelab.redhat.com validation: Host subnets are not overlapping
DEBUG Host worker04.cdp.rdu2.scalelab.redhat.com validation: No IP collisions were detected by host 2ec0e61a-c7ec-5f37-917c-88a8a486ace9
DEBUG Host worker04.cdp.rdu2.scalelab.redhat.com validation: cnv is disabled
DEBUG Host worker04.cdp.rdu2.scalelab.redhat.com validation: lso is disabled
```

```

DEBUG Host worker04.cdp.rdu2.scalelab.redhat.com validation: lvm is disabled
DEBUG Host worker04.cdp.rdu2.scaleLab.redhat.com validation: mce is disabled
DEBUG Host worker04.cdp.rdu2.scalelab.redhat.com validation: odf is disabled
DEBUG Host worker10.cdp.rdu2.scalelab.redhat.com validation: Valid inventory exists for the host
DEBUG Host worker10.cdp.rdu2.scalelab.redhat.com validation: Sufficient CPU cores
DEBUG Host worker10.cdp.rdu2.scalelab.redhat.com validation: Sufficient minimum RAM
DEBUG Host worker10.cdp.rdu2.scalelab.redhat.com validation: Sufficient disk capacity
DEBUG Host worker10.cdp.rdu2.scalelab.redhat.com validation: Sufficient CPU cores for role worker
DEBUG Host worker10.cdp.rdu2.scalelab.redhat.com validation: Sufficient RAM for role worker
DEBUG Host worker10.cdp.rdu2.scalelab.redhat.com validation: Hostname worker10.cdp.rdu2.scalelab.redhat.com is unique in cluster
DEBUG Host worker10.cdp.rdu2.scaleLab.redhat.com validation: Hostname worker10.cdp.rdu2.scaleLab.redhat.com is allowed
DEBUG Host worker10.cdp.rdu2.scalelab.redhat.com validation: Speed of installation disk has not yet been measured
DEBUG Host worker10.cdp.rdu2.scalelab.redhat.com validation: Host is compatible with cluster platform baremetal
DEBUG Host worker10.cdp.rdu2.scalelab.redhat.com validation: Host agent is compatible with the service
DEBUG Host worker10.cdp.rdu2.scalelab.redhat.com validation: No request to skip formatting of the installation disk
DEBUG Host worker10.cdp.rdu2.scalelab.redhat.com validation: All disks that have skipped formatting are present in the host inventory
DEBUG Host worker10.cdp.rdu2.scaleLab.redhat.com validation: Host is connected
DEBUG Host worker10.cdp.rdu2.scalelab.redhat.com validation: Media device is connected
DEBUG Host worker10.cdp.rdu2.scalelab.redhat.com validation: Machine Network CIDR is defined
DEBUG Host worker10.cdp.rdu2.scalelab.redhat.com validation: Host belongs to all machine network CIDRs
DEBUG Host worker10.cdp.rdu2.scalelab.redhat.com validation: Host has connectivity to the majority of hosts in the cluster
DEBUG Host worker10.cdp.rdu2.scaleLab.redhat.com validation: Platform OpenShift Virtualization is allowed
WARNING Host worker10.cdp.rdu2.scalelab.redhat.com validation: Host couldn't synchronize with any NTP server
DEBUG Host worker10.cdp.rdu2.scalelab.redhat.com validation: Host clock is synchronized with service
DEBUG Host worker10.cdp.rdu2.scalelab.redhat.com validation: All required container images were either pulled successfully or no attempt was made to pull them
DEBUG Host worker10.cdp.rdu2.scaleLab.redhat.com validation: Network Latency requirement has been satisfied.
DEBUG Host worker10.cdp.rdu2.scalelab.redhat.com validation: Packet Loss requirement has been satisfied.
DEBUG Host worker10.cdp.rdu2.scalelab.redhat.com validation: Host has been configured with at least one default route.
DEBUG Host worker10.cdp.rdu2.scalelab.redhat.com validation: Domain name resolution for the api.cdp.rdu2.scalelab.redhat.com domain was successful or not required
DEBUG Host worker10.cdp.rdu2.scaleLab.redhat.com validation: Domain name resolution for the api-int.cdp.rdu2.scaleLab.redhat.com domain was successful or not required
DEBUG Host worker10.cdp.rdu2.scalelab.redhat.com validation: Domain name resolution for the *.apps.cdp.rdu2.scalelab.redhat.com domain was successful or not required
DEBUG Host worker10.cdp.rdu2.scaleLab.redhat.com validation: DNS wildcard check was successful
DEBUG Host worker10.cdp.rdu2.scalelab.redhat.com validation: Host subnets are not overlapping
DEBUG Host worker10.cdp.rdu2.scalelab.redhat.com validation: No IP collisions were detected by host dfd6e101-e113-5938-9c9f-b5598c386584
DEBUG Host worker10.cdp.rdu2.scalelab.redhat.com validation: cnv is disabled
DEBUG Host worker10.cdp.rdu2.scalelab.redhat.com validation: lso is disabled
DEBUG Host worker10.cdp.rdu2.scalelab.redhat.com validation: lvm is disabled
DEBUG Host worker10.cdp.rdu2.scalelab.redhat.com validation: mce is disabled
DEBUG Host worker10.cdp.rdu2.scalelab.redhat.com validation: odf is disabled
DEBUG Host master03.cdp.rdu2.scalelab.redhat.com validation: Valid inventory exists for the host
DEBUG Host master03.cdp.rdu2.scaleLab.redhat.com validation: Sufficient CPU cores
DEBUG Host master03.cdp.rdu2.scalelab.redhat.com validation: Sufficient minimum RAM
DEBUG Host master03.cdp.rdu2.scalelab.redhat.com validation: Sufficient disk capacity
DEBUG Host master03.cdp.rdu2.scaleLab.redhat.com validation: Sufficient CPU cores for role master
DEBUG Host master03.cdp.rdu2.scalelab.redhat.com validation: Sufficient RAM for role master
DEBUG Host master03.cdp.rdu2.scalelab.redhat.com validation: Hostname master03.cdp.rdu2.scalelab.redhat.com is unique in cluster
DEBUG Host master03.cdp.rdu2.scaleLab.redhat.com validation: Hostname master03.cdp.rdu2.scaleLab.redhat.com is allowed
DEBUG Host master03.cdp.rdu2.scalelab.redhat.com validation: Speed of installation disk has not yet been measured
DEBUG Host master03.cdp.rdu2.scalelab.redhat.com validation: Host is compatible with cluster platform baremetal
DEBUG Host master03.cdp.rdu2.scalelab.redhat.com validation: Host agent is compatible with the service
DEBUG Host master03.cdp.rdu2.scalelab.redhat.com validation: No request to skip formatting of the installation disk
DEBUG Host master03.cdp.rdu2.scalelab.redhat.com validation: All disks that have skipped formatting are present in the host inventory
DEBUG Host master03.cdp.rdu2.scaleLab.redhat.com validation: Host is connected
DEBUG Host master03.cdp.rdu2.scalelab.redhat.com validation: Media device is connected
DEBUG Host master03.cdp.rdu2.scalelab.redhat.com validation: Machine Network CIDR is defined
DEBUG Host master03.cdp.rdu2.scalelab.redhat.com validation: Host belongs to all machine network CIDRs
DEBUG Host master03.cdp.rdu2.scalelab.redhat.com validation: Host has connectivity to the majority of hosts in the cluster
DEBUG Host master03.cdp.rdu2.scaleLab.redhat.com validation: Platform OpenShift Virtualization is allowed
WARNING Host master03.cdp.rdu2.scalelab.redhat.com validation: Host couldn't synchronize with any NTP server
DEBUG Host master03.cdp.rdu2.scalelab.redhat.com validation: Host clock is synchronized with service

```

```
DEBUG Host master03.cdp.rdu2.scalelab.redhat.com validation: All required container images were either pulled successfully or no attempt was made to pull them
DEBUG Host master03.cdp.rdu2.scalelab.redhat.com validation: Network Latency requirement has been satisfied.
DEBUG Host master03.cdp.rdu2.scalelab.redhat.com validation: Packet Loss requirement has been satisfied.
DEBUG Host master03.cdp.rdu2.scalelab.redhat.com validation: Host has been configured with at least one default route.
DEBUG Host master03.cdp.rdu2.scalelab.redhat.com validation: Domain name resolution for the api.cdp.rdu2.scalelab.redhat.com domain was successful or not required
DEBUG Host master03.cdp.rdu2.scalelab.redhat.com validation: Domain name resolution for the api-int.cdp.rdu2.scalelab.redhat.com domain was successful or not required
DEBUG Host master03.cdp.rdu2.scalelab.redhat.com validation: Domain name resolution for the *.apps.cdp.rdu2.scalelab.redhat.com domain was successful or not required
DEBUG Host master03.cdp.rdu2.scalelab.redhat.com validation: DNS wildcard check was successful
DEBUG Host master03.cdp.rdu2.scalelab.redhat.com validation: Host subnets are not overlapping
DEBUG Host master03.cdp.rdu2.scalelab.redhat.com validation: No IP collisions were detected by host 3833b526-a7bf-5672-a028-f59d7d680ec8
DEBUG Host master03.cdp.rdu2.scalelab.redhat.com validation: cnv is disabled
DEBUG Host master03.cdp.rdu2.scalelab.redhat.com validation: lso is disabled
DEBUG Host master03.cdp.rdu2.scalelab.redhat.com validation: lvm is disabled
DEBUG Host master03.cdp.rdu2.scalelab.redhat.com validation: mce is disabled
DEBUG Host master03.cdp.rdu2.scalelab.redhat.com validation: odf is disabled
DEBUG Host worker01.cdp.rdu2.scalelab.redhat.com validation: Valid inventory exists for the host
DEBUG Host worker01.cdp.rdu2.scalelab.redhat.com validation: Sufficient CPU cores
DEBUG Host worker01.cdp.rdu2.scalelab.redhat.com validation: Sufficient minimum RAM
DEBUG Host worker01.cdp.rdu2.scalelab.redhat.com validation: Sufficient disk capacity
DEBUG Host worker01.cdp.rdu2.scalelab.redhat.com validation: Sufficient CPU cores for role worker
DEBUG Host worker01.cdp.rdu2.scalelab.redhat.com validation: Sufficient RAM for role worker
DEBUG Host worker01.cdp.rdu2.scalelab.redhat.com validation: Hostname worker01.cdp.rdu2.scalelab.redhat.com is unique in cluster
DEBUG Host worker01.cdp.rdu2.scalelab.redhat.com validation: Hostname worker01.cdp.rdu2.scalelab.redhat.com is allowed
DEBUG Host worker01.cdp.rdu2.scalelab.redhat.com validation: Speed of installation disk has not yet been measured
DEBUG Host worker01.cdp.rdu2.scalelab.redhat.com validation: Host is compatible with cluster platform baremetal
DEBUG Host worker01.cdp.rdu2.scalelab.redhat.com validation: Host agent is compatible with the service
DEBUG Host worker01.cdp.rdu2.scalelab.redhat.com validation: No request to skip formatting of the installation disk
DEBUG Host worker01.cdp.rdu2.scalelab.redhat.com validation: All disks that have skipped formatting are present in the host inventory
DEBUG Host worker01.cdp.rdu2.scalelab.redhat.com validation: Host is connected
DEBUG Host worker01.cdp.rdu2.scalelab.redhat.com validation: Media device is connected
DEBUG Host worker01.cdp.rdu2.scalelab.redhat.com validation: Machine Network CIDR is defined
DEBUG Host worker01.cdp.rdu2.scalelab.redhat.com validation: Host belongs to all machine network CIDRs
DEBUG Host worker01.cdp.rdu2.scalelab.redhat.com validation: Host has connectivity to the majority of hosts in the cluster
DEBUG Host worker01.cdp.rdu2.scalelab.redhat.com validation: Platform OpenShift Virtualization is allowed
WARNING Host worker01.cdp.rdu2.scalelab.redhat.com validation: Host couldn't synchronize with any NTP server
DEBUG Host worker01.cdp.rdu2.scalelab.redhat.com validation: Host clock is synchronized with service
DEBUG Host worker01.cdp.rdu2.scalelab.redhat.com validation: All required container images were either pulled successfully or no attempt was made to pull them
DEBUG Host worker01.cdp.rdu2.scalelab.redhat.com validation: Network Latency requirement has been satisfied.
DEBUG Host worker01.cdp.rdu2.scalelab.redhat.com validation: Packet Loss requirement has been satisfied.
DEBUG Host worker01.cdp.rdu2.scalelab.redhat.com validation: Host has been configured with at least one default route.
DEBUG Host worker01.cdp.rdu2.scalelab.redhat.com validation: Domain name resolution for the api.cdp.rdu2.scalelab.redhat.com domain was successful or not required
DEBUG Host worker01.cdp.rdu2.scalelab.redhat.com validation: Domain name resolution for the api-int.cdp.rdu2.scalelab.redhat.com domain was successful or not required
DEBUG Host worker01.cdp.rdu2.scalelab.redhat.com validation: Domain name resolution for the *.apps.cdp.rdu2.scalelab.redhat.com domain was successful or not required
DEBUG Host worker01.cdp.rdu2.scalelab.redhat.com validation: DNS wildcard check was successful
DEBUG Host worker01.cdp.rdu2.scalelab.redhat.com validation: Host subnets are not overlapping
DEBUG Host worker01.cdp.rdu2.scalelab.redhat.com validation: No IP collisions were detected by host b1cf435-1098-50fe-a95f-ab3765e345e9
DEBUG Host worker01.cdp.rdu2.scalelab.redhat.com validation: cnv is disabled
DEBUG Host worker01.cdp.rdu2.scalelab.redhat.com validation: lso is disabled
DEBUG Host worker01.cdp.rdu2.scalelab.redhat.com validation: lvm is disabled
DEBUG Host worker01.cdp.rdu2.scalelab.redhat.com validation: mce is disabled
DEBUG Host worker01.cdp.rdu2.scalelab.redhat.com validation: odf is disabled
DEBUG Host infra01.cdp.rdu2.scalelab.redhat.com validation: Valid inventory exists for the host
DEBUG Host infra01.cdp.rdu2.scalelab.redhat.com validation: Sufficient CPU cores
DEBUG Host infra01.cdp.rdu2.scalelab.redhat.com validation: Sufficient minimum RAM
DEBUG Host infra01.cdp.rdu2.scalelab.redhat.com validation: Sufficient disk capacity
DEBUG Host infra01.cdp.rdu2.scalelab.redhat.com validation: Sufficient CPU cores for role worker
DEBUG Host infra01.cdp.rdu2.scalelab.redhat.com validation: Sufficient RAM for role worker
DEBUG Host infra01.cdp.rdu2.scalelab.redhat.com validation: Hostname infra01.cdp.rdu2.scalelab.redhat.com is unique in cluster
```

```

DEBUG Host infra01.cdp.rdu2.scalelab.redhat.com validation: Hostname infra01.cdp.rdu2.scalelab.redhat.com is allowed
DEBUG Host infra01.cdp.rdu2.scalelab.redhat.com validation: Speed of installation disk has not yet been measured
DEBUG Host infra01.cdp.rdu2.scalelab.redhat.com validation: Host is compatible with cluster platform baremetal
DEBUG Host infra01.cdp.rdu2.scalelab.redhat.com validation: Host agent is compatible with the service
DEBUG Host infra01.cdp.rdu2.scalelab.redhat.com validation: No request to skip formatting of the installation disk
DEBUG Host infra01.cdp.rdu2.scalelab.redhat.com validation: All disks that have skipped formatting are present in the host inventory
DEBUG Host infra01.cdp.rdu2.scaleLab.redhat.com validation: Host is connected
DEBUG Host infra01.cdp.rdu2.scalelab.redhat.com validation: Media device is connected
DEBUG Host infra01.cdp.rdu2.scalelab.redhat.com validation: Machine Network CIDR is defined
DEBUG Host infra01.cdp.rdu2.scalelab.redhat.com validation: Host belongs to all machine network CIDRs
WARNING Host infra01.cdp.rdu2.scaleLab.redhat.com validation: No connectivity to the majority of hosts in the cluster
DEBUG Host infra01.cdp.rdu2.scaleLab.redhat.com validation: Platform OpenShift Virtualization is allowed
WARNING Host infra01.cdp.rdu2.scaleLab.redhat.com validation: Host couldn't synchronize with any NTP server
DEBUG Host infra01.cdp.rdu2.scaleLab.redhat.com validation: Host clock is synchronized with service
DEBUG Host infra01.cdp.rdu2.scaleLab.redhat.com validation: All required container images were either pulled successfully or no attempt was made to pull them
DEBUG Host infra01.cdp.rdu2.scaleLab.redhat.com validation: Host has been configured with at least one default route.
DEBUG Host infra01.cdp.rdu2.scaleLab.redhat.com validation: Domain name resolution for the api.cdp.rdu2.scaleLab.redhat.com domain was successful or not required
DEBUG Host infra01.cdp.rdu2.scaleLab.redhat.com validation: Domain name resolution for the api-int.cdp.rdu2.scaleLab.redhat.com domain was successful or not required
DEBUG Host infra01.cdp.rdu2.scaleLab.redhat.com validation: Domain name resolution for the *.apps.cdp.rdu2.scaleLab.redhat.com domain was successful or not required
DEBUG Host infra01.cdp.rdu2.scaleLab.redhat.com validation: Host subnets are not overlapping
DEBUG Host infra01.cdp.rdu2.scaleLab.redhat.com validation: No IP collisions were detected by host 1ddf58df-3d52-50e0-bd87-9391b5b9f1dc
DEBUG Host infra01.cdp.rdu2.scaleLab.redhat.com validation: cnv is disabled
DEBUG Host infra01.cdp.rdu2.scaleLab.redhat.com validation: lso is disabled
DEBUG Host infra01.cdp.rdu2.scaleLab.redhat.com validation: lvm is disabled
DEBUG Host infra01.cdp.rdu2.scaleLab.redhat.com validation: mce is disabled
DEBUG Host infra01.cdp.rdu2.scaleLab.redhat.com validation: odf is disabled
DEBUG Host worker09.cdp.rdu2.scaleLab.redhat.com validation: Valid inventory exists for the host
DEBUG Host worker09.cdp.rdu2.scaleLab.redhat.com validation: Sufficient CPU cores
DEBUG Host worker09.cdp.rdu2.scaleLab.redhat.com validation: Sufficient minimum RAM
DEBUG Host worker09.cdp.rdu2.scaleLab.redhat.com validation: Sufficient disk capacity
DEBUG Host worker09.cdp.rdu2.scaleLab.redhat.com validation: Sufficient CPU cores for role worker
DEBUG Host worker09.cdp.rdu2.scaleLab.redhat.com validation: Sufficient RAM for role worker
DEBUG Host worker09.cdp.rdu2.scaleLab.redhat.com validation: Hostname worker09.cdp.rdu2.scaleLab.redhat.com is unique in cluster
DEBUG Host worker09.cdp.rdu2.scaleLab.redhat.com validation: Hostname worker09.cdp.rdu2.scaleLab.redhat.com is allowed
DEBUG Host worker09.cdp.rdu2.scaleLab.redhat.com validation: Speed of installation disk has not yet been measured
DEBUG Host worker09.cdp.rdu2.scaleLab.redhat.com validation: Host is compatible with cluster platform baremetal
DEBUG Host worker09.cdp.rdu2.scaleLab.redhat.com validation: Host agent is compatible with the service
DEBUG Host worker09.cdp.rdu2.scaleLab.redhat.com validation: No request to skip formatting of the installation disk
DEBUG Host worker09.cdp.rdu2.scaleLab.redhat.com validation: All disks that have skipped formatting are present in the host inventory
DEBUG Host worker09.cdp.rdu2.scaleLab.redhat.com validation: Host is connected
DEBUG Host worker09.cdp.rdu2.scaleLab.redhat.com validation: Media device is connected
DEBUG Host worker09.cdp.rdu2.scaleLab.redhat.com validation: Machine Network CIDR is defined
DEBUG Host worker09.cdp.rdu2.scaleLab.redhat.com validation: Host belongs to all machine network CIDRs
DEBUG Host worker09.cdp.rdu2.scaleLab.redhat.com validation: Host has connectivity to the majority of hosts in the cluster
DEBUG Host worker09.cdp.rdu2.scaleLab.redhat.com validation: Platform OpenShift Virtualization is allowed
WARNING Host worker09.cdp.rdu2.scaleLab.redhat.com validation: Host couldn't synchronize with any NTP server
DEBUG Host worker09.cdp.rdu2.scaleLab.redhat.com validation: Host clock is synchronized with service
DEBUG Host worker09.cdp.rdu2.scaleLab.redhat.com validation: All required container images were either pulled successfully or no attempt was made to pull them
DEBUG Host worker09.cdp.rdu2.scaleLab.redhat.com validation: Network latency requirement has been satisfied.
DEBUG Host worker09.cdp.rdu2.scaleLab.redhat.com validation: Packet loss requirement has been satisfied.
DEBUG Host worker09.cdp.rdu2.scaleLab.redhat.com validation: Host has been configured with at least one default route.
DEBUG Host worker09.cdp.rdu2.scaleLab.redhat.com validation: Domain name resolution for the api.cdp.rdu2.scaleLab.redhat.com domain was successful or not required
DEBUG Host worker09.cdp.rdu2.scaleLab.redhat.com validation: Domain name resolution for the api-int.cdp.rdu2.scaleLab.redhat.com domain was successful or not required
DEBUG Host worker09.cdp.rdu2.scaleLab.redhat.com validation: Domain name resolution for the *.apps.cdp.rdu2.scaleLab.redhat.com domain was successful or not required
DEBUG Host worker09.cdp.rdu2.scaleLab.redhat.com validation: DNS wildcard check was successful
DEBUG Host worker09.cdp.rdu2.scaleLab.redhat.com validation: Host subnets are not overlapping
DEBUG Host worker09.cdp.rdu2.scaleLab.redhat.com validation: No IP collisions were detected by host 422bcd9f-8ff0-57a3-9c2e-e575ff094535
DEBUG Host worker09.cdp.rdu2.scaleLab.redhat.com validation: cnv is disabled

```

```

DEBUG Host worker09.cdp.rdu2.scalelab.redhat.com validation: lso is disabled
DEBUG Host worker09.cdp.rdu2.scalelab.redhat.com validation: lvm is disabled
DEBUG Host worker09.cdp.rdu2.scalelab.redhat.com validation: mce is disabled
DEBUG Host worker09.cdp.rdu2.scalelab.redhat.com validation: odf is disabled
DEBUG Host worker03.cdp.rdu2.scalelab.redhat.com validation: Valid inventory exists for the host
DEBUG Host worker03.cdp.rdu2.scalelab.redhat.com validation: Sufficient CPU cores
DEBUG Host worker03.cdp.rdu2.scalelab.redhat.com validation: Sufficient minimum RAM
DEBUG Host worker03.cdp.rdu2.scalelab.redhat.com validation: Sufficient disk capacity
DEBUG Host worker03.cdp.rdu2.scalelab.redhat.com validation: Sufficient CPU cores for role worker
DEBUG Host worker03.cdp.rdu2.scalelab.redhat.com validation: Sufficient RAM for role worker
DEBUG Host worker03.cdp.rdu2.scalelab.redhat.com validation: Hostname worker03.cdp.rdu2.scalelab.redhat.com is unique in cluster
DEBUG Host worker03.cdp.rdu2.scalelab.redhat.com validation: Hostname worker03.cdp.rdu2.scalelab.redhat.com is allowed
DEBUG Host worker03.cdp.rdu2.scalelab.redhat.com validation: Speed of installation disk has not yet been measured
DEBUG Host worker03.cdp.rdu2.scalelab.redhat.com validation: Host is compatible with cluster platform baremetal
DEBUG Host worker03.cdp.rdu2.scalelab.redhat.com validation: Host agent is compatible with the service
DEBUG Host worker03.cdp.rdu2.scalelab.redhat.com validation: No request to skip formatting of the installation disk
DEBUG Host worker03.cdp.rdu2.scalelab.redhat.com validation: ALL disks that have skipped formatting are present in the host inventory
DEBUG Host worker03.cdp.rdu2.scalelab.redhat.com validation: Host is connected
DEBUG Host worker03.cdp.rdu2.scalelab.redhat.com validation: Media device is connected
DEBUG Host worker03.cdp.rdu2.scalelab.redhat.com validation: Machine Network CIDR is defined
DEBUG Host worker03.cdp.rdu2.scalelab.redhat.com validation: Host belongs to all machine network CIDRs
DEBUG Host worker03.cdp.rdu2.scalelab.redhat.com validation: Host has connectivity to the majority of hosts in the cluster
DEBUG Host worker03.cdp.rdu2.scalelab.redhat.com validation: Platform OpenShift Virtualization is allowed
WARNING Host worker03.cdp.rdu2.scalelab.redhat.com validation: Host couldn't synchronize with any NTP server
DEBUG Host worker03.cdp.rdu2.scalelab.redhat.com validation: Host clock is synchronized with service
DEBUG Host worker03.cdp.rdu2.scalelab.redhat.com validation: All required container images were either pulled successfully or no attempt was made to pull them
DEBUG Host worker03.cdp.rdu2.scalelab.redhat.com validation: Network latency requirement has been satisfied.
DEBUG Host worker03.cdp.rdu2.scalelab.redhat.com validation: Packet loss requirement has been satisfied.
DEBUG Host worker03.cdp.rdu2.scalelab.redhat.com validation: Host has been configured with at least one default route.
DEBUG Host worker03.cdp.rdu2.scalelab.redhat.com validation: Domain name resolution for the api.cdp.rdu2.scalelab.redhat.com domain was successful or not required
DEBUG Host worker03.cdp.rdu2.scalelab.redhat.com validation: Domain name resolution for the api-int.cdp.rdu2.scalelab.redhat.com domain was successful or not required
DEBUG Host worker03.cdp.rdu2.scalelab.redhat.com validation: Domain name resolution for the *.apps.cdp.rdu2.scalelab.redhat.com domain was successful or not required
DEBUG Host worker03.cdp.rdu2.scalelab.redhat.com validation: DNS wildcard check was successful
DEBUG Host worker03.cdp.rdu2.scalelab.redhat.com validation: Host subnets are not overlapping
DEBUG Host worker03.cdp.rdu2.scalelab.redhat.com validation: No IP collisions were detected by host def36ddf-7a48-5a98-860a-1ca00fe0ffec
DEBUG Host worker03.cdp.rdu2.scalelab.redhat.com validation: cnv is disabled
DEBUG Host worker03.cdp.rdu2.scalelab.redhat.com validation: lso is disabled
DEBUG Host worker03.cdp.rdu2.scalelab.redhat.com validation: lvm is disabled
DEBUG Host worker03.cdp.rdu2.scalelab.redhat.com validation: mce is disabled
DEBUG Host worker03.cdp.rdu2.scalelab.redhat.com validation: odf is disabled
DEBUG Host worker07.cdp.rdu2.scalelab.redhat.com validation: Host is connected
DEBUG Host worker07.cdp.rdu2.scalelab.redhat.com validation: Media device is connected
DEBUG Host worker07.cdp.rdu2.scalelab.redhat.com validation: Machine Network CIDR is defined
DEBUG Host worker07.cdp.rdu2.scalelab.redhat.com validation: Host belongs to all machine network CIDRs
DEBUG Host worker07.cdp.rdu2.scalelab.redhat.com validation: Host has connectivity to the majority of hosts in the cluster
DEBUG Host worker07.cdp.rdu2.scalelab.redhat.com validation: Platform OpenShift Virtualization is allowed
WARNING Host worker07.cdp.rdu2.scalelab.redhat.com validation: Host couldn't synchronize with any NTP server
DEBUG Host worker07.cdp.rdu2.scalelab.redhat.com validation: Host clock is synchronized with service
DEBUG Host worker07.cdp.rdu2.scalelab.redhat.com validation: All required container images were either pulled successfully or no attempt was made to pull them
DEBUG Host worker07.cdp.rdu2.scalelab.redhat.com validation: Network latency requirement has been satisfied.
DEBUG Host worker07.cdp.rdu2.scalelab.redhat.com validation: Packet loss requirement has been satisfied.
DEBUG Host worker07.cdp.rdu2.scalelab.redhat.com validation: Host has been configured with at least one default route.
DEBUG Host worker07.cdp.rdu2.scalelab.redhat.com validation: Domain name resolution for the api.cdp.rdu2.scalelab.redhat.com domain was successful or not required
DEBUG Host worker07.cdp.rdu2.scalelab.redhat.com validation: Domain name resolution for the api-int.cdp.rdu2.scalelab.redhat.com domain was successful or not required
DEBUG Host worker07.cdp.rdu2.scalelab.redhat.com validation: Domain name resolution for the *.apps.cdp.rdu2.scalelab.redhat.com domain was successful or not required
DEBUG Host worker07.cdp.rdu2.scalelab.redhat.com validation: DNS wildcard check was successful
DEBUG Host worker07.cdp.rdu2.scalelab.redhat.com validation: Host subnets are not overlapping
DEBUG Host worker07.cdp.rdu2.scalelab.redhat.com validation: No IP collisions were detected by host 4ae3ae26-a3ef-5845-8a63-f0481cb9bff4

```

```

DEBUG Host worker07.cdp.rdu2.scalelab.redhat.com validation: cnv is disabled
DEBUG Host worker07.cdp.rdu2.scaleLab.redhat.com validation: lso is disabled
DEBUG Host worker07.cdp.rdu2.scalelab.redhat.com validation: lvm is disabled
DEBUG Host worker07.cdp.rdu2.scalelab.redhat.com validation: mce is disabled
DEBUG Host worker07.cdp.rdu2.scalelab.redhat.com validation: odf is disabled
DEBUG Host worker07.cdp.rdu2.scalelab.redhat.com validation: Valid inventory exists for the host
DEBUG Host worker07.cdp.rdu2.scalelab.redhat.com validation: Sufficient CPU cores
DEBUG Host worker07.cdp.rdu2.scalelab.redhat.com validation: Sufficient minimum RAM
DEBUG Host worker07.cdp.rdu2.scalelab.redhat.com validation: Sufficient disk capacity
DEBUG Host worker07.cdp.rdu2.scalelab.redhat.com validation: Sufficient CPU cores for role worker
DEBUG Host worker07.cdp.rdu2.scalelab.redhat.com validation: Sufficient RAM for role worker
DEBUG Host worker07.cdp.rdu2.scaleLab.redhat.com validation: Hostname worker07.cdp.rdu2.scalelab.redhat.com is unique in cluster
DEBUG Host worker07.cdp.rdu2.scalelab.redhat.com validation: Hostname worker07.cdp.rdu2.scalelab.redhat.com is allowed
DEBUG Host worker07.cdp.rdu2.scalelab.redhat.com validation: Speed of installation disk has not yet been measured
DEBUG Host worker07.cdp.rdu2.scalelab.redhat.com validation: Host is compatible with cluster platform baremetal
DEBUG Host worker07.cdp.rdu2.scalelab.redhat.com validation: Host agent is compatible with the service
DEBUG Host worker07.cdp.rdu2.scalelab.redhat.com validation: No request to skip formatting of the installation disk
DEBUG Host worker07.cdp.rdu2.scalelab.redhat.com validation: All disks that have skipped formatting are present in the host inventory
INFO Host infra01.cdp.rdu2.scaleLab.redhat.com: calculated role is worker
DEBUG Host infra02.cdp.rdu2.scaleLab.redhat.com validation: Host is connected
DEBUG Host infra02.cdp.rdu2.scaleLab.redhat.com validation: Media device is connected
DEBUG Host infra02.cdp.rdu2.scaleLab.redhat.com validation: Machine Network CIDR is defined
DEBUG Host infra02.cdp.rdu2.scaleLab.redhat.com validation: Host belongs to all machine network CIDRs
WARNING Host infra02.cdp.rdu2.scaleLab.redhat.com validation: No connectivity to the majority of hosts in the cluster
DEBUG Host infra02.cdp.rdu2.scaleLab.redhat.com validation: Platform OpenShift Virtualization is allowed
WARNING Host infra02.cdp.rdu2.scaleLab.redhat.com validation: Host couldn't synchronize with any NTP server
DEBUG Host infra02.cdp.rdu2.scaleLab.redhat.com validation: Host clock is synchronized with service
DEBUG Host infra02.cdp.rdu2.scaleLab.redhat.com validation: All required container images were either pulled successfully or no attempt was made to pull them
DEBUG Host infra02.cdp.rdu2.scaleLab.redhat.com validation: Network latency requirement has been satisfied.
DEBUG Host infra02.cdp.rdu2.scaleLab.redhat.com validation: Packet loss requirement has been satisfied.
DEBUG Host infra02.cdp.rdu2.scaleLab.redhat.com validation: Host has been configured with at least one default route.
DEBUG Host infra02.cdp.rdu2.scaleLab.redhat.com validation: Domain name resolution for the api.cdp.rdu2.scaleLab.redhat.com domain was successful or not required
DEBUG Host infra02.cdp.rdu2.scaleLab.redhat.com validation: Domain name resolution for the api-int.cdp.rdu2.scaleLab.redhat.com domain was successful or not required
DEBUG Host infra02.cdp.rdu2.scaleLab.redhat.com validation: Domain name resolution for the *.apps.cdp.rdu2.scaleLab.redhat.com domain was successful or not required
DEBUG Host infra02.cdp.rdu2.scaleLab.redhat.com validation: Host subnets are not overlapping
DEBUG Host infra02.cdp.rdu2.scaleLab.redhat.com validation: No IP collisions were detected by host fd61237d-f691-5b68-8def-e0e42f360e98
DEBUG Host infra02.cdp.rdu2.scaleLab.redhat.com validation: cnv is disabled
DEBUG Host infra02.cdp.rdu2.scaleLab.redhat.com validation: lso is disabled
DEBUG Host infra02.cdp.rdu2.scaleLab.redhat.com validation: lvm is disabled
DEBUG Host infra02.cdp.rdu2.scaleLab.redhat.com validation: mce is disabled
DEBUG Host infra02.cdp.rdu2.scaleLab.redhat.com validation: odf is disabled
DEBUG Host infra02.cdp.rdu2.scaleLab.redhat.com validation: Valid inventory exists for the host
DEBUG Host infra02.cdp.rdu2.scaleLab.redhat.com validation: Sufficient CPU cores
DEBUG Host infra02.cdp.rdu2.scaleLab.redhat.com validation: Sufficient minimum RAM
DEBUG Host infra02.cdp.rdu2.scaleLab.redhat.com validation: Sufficient disk capacity
DEBUG Host infra02.cdp.rdu2.scaleLab.redhat.com validation: Sufficient CPU cores for role auto-assign
DEBUG Host infra02.cdp.rdu2.scaleLab.redhat.com validation: Sufficient RAM for role auto-assign
DEBUG Host infra02.cdp.rdu2.scaleLab.redhat.com validation: Hostname infra02.cdp.rdu2.scaleLab.redhat.com is unique in cluster
DEBUG Host infra02.cdp.rdu2.scaleLab.redhat.com validation: Hostname infra02.cdp.rdu2.scaleLab.redhat.com is allowed
DEBUG Host infra02.cdp.rdu2.scaleLab.redhat.com validation: Speed of installation disk has not yet been measured
DEBUG Host infra02.cdp.rdu2.scaleLab.redhat.com validation: Host is compatible with cluster platform baremetal
DEBUG Host infra02.cdp.rdu2.scaleLab.redhat.com validation: Host agent is compatible with the service
DEBUG Host infra02.cdp.rdu2.scaleLab.redhat.com validation: No request to skip formatting of the installation disk
DEBUG Host infra02.cdp.rdu2.scaleLab.redhat.com validation: All disks that have skipped formatting are present in the host inventory
INFO Host infra02.cdp.rdu2.scaleLab.redhat.com: calculated role is worker
INFO Host infra01.cdp.rdu2.scaleLab.redhat.com validation: Host has connectivity to the majority of hosts in the cluster
INFO Host infra01.cdp.rdu2.scaleLab.redhat.com: updated status from insufficient to known (Host is ready to be installed)
INFO Host 590a24c7-af35-5db9-bddc-03da5310c1f0: Successfully registered
DEBUG Host infra03.cdp.rdu2.scaleLab.redhat.com validation: Valid inventory exists for the host
DEBUG Host infra03.cdp.rdu2.scaleLab.redhat.com validation: Sufficient CPU cores
DEBUG Host infra03.cdp.rdu2.scaleLab.redhat.com validation: Sufficient minimum RAM
DEBUG Host infra03.cdp.rdu2.scaleLab.redhat.com validation: Sufficient disk capacity
DEBUG Host infra03.cdp.rdu2.scaleLab.redhat.com validation: Sufficient CPU cores for role auto-assign

```

```

DEBUG Host infra03.cdp.rdu2.scalelab.redhat.com validation: Sufficient RAM for role auto-assign
DEBUG Host infra03.cdp.rdu2.scalelab.redhat.com validation: Hostname infra03.cdp.rdu2.scalelab.redhat.com is unique in
cluster
DEBUG Host infra03.cdp.rdu2.scaleLab.redhat.com validation: Hostname infra03.cdp.rdu2.scaleLab.redhat.com is allowed
DEBUG Host infra03.cdp.rdu2.scaleLab.redhat.com validation: Speed of installation disk has not yet been measured
DEBUG Host infra03.cdp.rdu2.scaleLab.redhat.com validation: Host is compatible with cluster platform baremetal
DEBUG Host infra03.cdp.rdu2.scaleLab.redhat.com validation: Host agent is compatible with the service
DEBUG Host infra03.cdp.rdu2.scaleLab.redhat.com validation: No request to skip formatting of the installation disk
DEBUG Host infra03.cdp.rdu2.scaleLab.redhat.com validation: All disks that have skipped formatting are present in the
host inventory
DEBUG Host infra03.cdp.rdu2.scaleLab.redhat.com validation: Host is connected
DEBUG Host infra03.cdp.rdu2.scaleLab.redhat.com validation: Media device is connected
DEBUG Host infra03.cdp.rdu2.scaleLab.redhat.com validation: Machine Network CIDR is defined
DEBUG Host infra03.cdp.rdu2.scaleLab.redhat.com validation: Host belongs to all machine network CIDRs
WARNING Host infra03.cdp.rdu2.scaleLab.redhat.com validation: No connectivity to the majority of hosts in the cluster
DEBUG Host infra03.cdp.rdu2.scaleLab.redhat.com validation: Platform OpenShift Virtualization is allowed
WARNING Host infra03.cdp.rdu2.scaleLab.redhat.com validation: Host couldn't synchronize with any NTP server
DEBUG Host infra03.cdp.rdu2.scaleLab.redhat.com validation: Host clock is synchronized with service
DEBUG Host infra03.cdp.rdu2.scaleLab.redhat.com validation: All required container images were either pulled successfully
or no attempt was made to pull them
DEBUG Host infra03.cdp.rdu2.scaleLab.redhat.com validation: Network Latency requirement has been satisfied.
DEBUG Host infra03.cdp.rdu2.scaleLab.redhat.com validation: Packet Loss requirement has been satisfied.
DEBUG Host infra03.cdp.rdu2.scaleLab.redhat.com validation: Host has been configured with at least one default route.
DEBUG Host infra03.cdp.rdu2.scaleLab.redhat.com validation: Domain name resolution for the
api.cdp.rdu2.scaleLab.redhat.com domain was successful or not required
DEBUG Host infra03.cdp.rdu2.scaleLab.redhat.com validation: Domain name resolution for the
api-int.cdp.rdu2.scaleLab.redhat.com domain was successful or not required
DEBUG Host infra03.cdp.rdu2.scaleLab.redhat.com validation: Domain name resolution for the
*.apps.cdp.rdu2.scaleLab.redhat.com domain was successful or not required
DEBUG Host infra03.cdp.rdu2.scaleLab.redhat.com validation: Host subnets are not overlapping
DEBUG Host infra03.cdp.rdu2.scaleLab.redhat.com validation: No IP collisions were detected by host
590a24c7-af35-5db9-bddc-03da5310c1f0
DEBUG Host infra03.cdp.rdu2.scaleLab.redhat.com validation: cnv is disabled
DEBUG Host infra03.cdp.rdu2.scaleLab.redhat.com validation: lso is disabled
DEBUG Host infra03.cdp.rdu2.scaleLab.redhat.com validation: lvm is disabled
DEBUG Host infra03.cdp.rdu2.scaleLab.redhat.com validation: mce is disabled
DEBUG Host infra03.cdp.rdu2.scaleLab.redhat.com validation: odf is disabled
INFO Host infra03.cdp.rdu2.scaleLab.redhat.com: calculated role is worker
INFO Host infra02.cdp.rdu2.scaleLab.redhat.com validation: Host has connectivity to the majority of hosts in the cluster
INFO Host infra02.cdp.rdu2.scaleLab.redhat.com: updated status from insufficient to known (Host is ready to be installed)
INFO Host infra03.cdp.rdu2.scaleLab.redhat.com validation: Host has connectivity to the majority of hosts in the cluster
INFO Host infra03.cdp.rdu2.scaleLab.redhat.com: updated status from insufficient to known (Host is ready to be installed)
INFO Cluster is ready for install
INFO Cluster validation: All hosts in the cluster are ready to install.
INFO Preparing cluster for installation
INFO Host infra01.cdp.rdu2.scaleLab.redhat.com: updated status from known to preparing-for-installation (Host finished
successfully to prepare for installation)
INFO Host worker05.cdp.rdu2.scaleLab.redhat.com: New image status
quay.io/openshift-release-dev/ocp-v4.0-art-dev@sha256:eba47e7d1ab87091b9a4657e28d203a5bb96f03e118c863ce9ed846d44088cc5.
result: success. time: 1.38 seconds; size: 447.71 Megabytes; download rate: 338.98 MBps
INFO Host worker10.cdp.rdu2.scaleLab.redhat.com: New image status
quay.io/openshift-release-dev/ocp-v4.0-art-dev@sha256:eba47e7d1ab87091b9a4657e28d203a5bb96f03e118c863ce9ed846d44088cc5.
result: success. time: 1.05 seconds; size: 447.71 Megabytes; download rate: 449.24 MBps
INFO Host worker08.cdp.rdu2.scaleLab.redhat.com: New image status
quay.io/openshift-release-dev/ocp-v4.0-art-dev@sha256:eba47e7d1ab87091b9a4657e28d203a5bb96f03e118c863ce9ed846d44088cc5.
result: success. time: 1.73 seconds; size: 447.71 Megabytes; download rate: 271.09 MBps
INFO Host worker03.cdp.rdu2.scaleLab.redhat.com: New image status
quay.io/openshift-release-dev/ocp-v4.0-art-dev@sha256:eba47e7d1ab87091b9a4657e28d203a5bb96f03e118c863ce9ed846d44088cc5.
result: success. time: 1.25 seconds; size: 447.71 Megabytes; download rate: 375.09 MBps
INFO Host worker01.cdp.rdu2.scaleLab.redhat.com: New image status
quay.io/openshift-release-dev/ocp-v4.0-art-dev@sha256:eba47e7d1ab87091b9a4657e28d203a5bb96f03e118c863ce9ed846d44088cc5.
result: success. time: 1.16 seconds; size: 447.71 Megabytes; download rate: 404.46 MBps
INFO Host worker04.cdp.rdu2.scaleLab.redhat.com: updated status from preparing-for-installation to preparing-successful
(Host finished successfully to prepare for installation)
INFO Host infra02.cdp.rdu2.scaleLab.redhat.com: New image status
quay.io/openshift-release-dev/ocp-v4.0-art-dev@sha256:eba47e7d1ab87091b9a4657e28d203a5bb96f03e118c863ce9ed846d44088cc5.
result: success. time: 1.13 seconds; size: 447.71 Megabytes; download rate: 413.80 MBps
INFO Host worker08.cdp.rdu2.scaleLab.redhat.com: updated status from preparing-for-installation to preparing-successful
(Host finished successfully to prepare for installation)
INFO Host worker06.cdp.rdu2.scaleLab.redhat.com: updated status from preparing-for-installation to preparing-successful
(Host finished successfully to prepare for installation)

```

```

INFO Host worker03.cdp.rdu2.scalelab.redhat.com: updated status from preparing-for-installation to preparing-successful
(Host finished successfully to prepare for installation)
INFO Host infra01.cdp.rdu2.scalelab.redhat.com: New image status
quay.io/openshift-release-dev/ocp-v4.0-art-dev@sha256:eba47e7d1ab87091b9a4657e28d203a5bb96f03e118c863ce9ed846d44088cc5.
result: success. time: 1.06 seconds; size: 447.71 Megabytes; download rate: 442.97 MBps
INFO Host master02.cdp.rdu2.scalelab.redhat.com: New image status
quay.io/openshift-release-dev/ocp-v4.0-art-dev@sha256:eba47e7d1ab87091b9a4657e28d203a5bb96f03e118c863ce9ed846d44088cc5.
result: success. time: 1.28 seconds; size: 447.71 Megabytes; download rate: 366.99 MBps
INFO Host worker07.cdp.rdu2.scalelab.redhat.com: updated status from preparing-for-installation to preparing-successful
(Host finished successfully to prepare for installation)
INFO Host infra01.cdp.rdu2.scalelab.redhat.com: updated status from preparing-for-installation to preparing-successful
(Host finished successfully to prepare for installation)
INFO Cluster installation in progress
INFO Host master02.cdp.rdu2.scalelab.redhat.com: updated status from preparing-for-installation to preparing-successful
(Host finished successfully to prepare for installation)
INFO Host infra02.cdp.rdu2.scalelab.redhat.com: updated status from preparing-successful to installing (Installation is
in progress)
INFO Host: master02.cdp.rdu2.scalelab.redhat.com, reached installation stage Writing image to disk: 6%
INFO Host: master02.cdp.rdu2.scalelab.redhat.com, reached installation stage Writing image to disk: 14%
INFO Host: master02.cdp.rdu2.scalelab.redhat.com, reached installation stage Writing image to disk: 21%
INFO Host: worker09.cdp.rdu2.scalelab.redhat.com, reached installation stage Writing image to disk
INFO Host: worker09.cdp.rdu2.scalelab.redhat.com, reached installation stage Writing image to disk: 7%
INFO Host: master02.cdp.rdu2.scalelab.redhat.com, reached installation stage Writing image to disk: 45%
INFO Host: worker05.cdp.rdu2.scalelab.redhat.com, reached installation stage Writing image to disk: 16%
INFO Host: master02.cdp.rdu2.scalelab.redhat.com, reached installation stage Writing image to disk: 67%
INFO Host: worker08.cdp.rdu2.scalelab.redhat.com, reached installation stage Writing image to disk: 10%
INFO Host: worker03.cdp.rdu2.scalelab.redhat.com, reached installation stage Writing image to disk: 14%
INFO Host: worker03.cdp.rdu2.scalelab.redhat.com, reached installation stage Writing image to disk: 20%
INFO Host: worker06.cdp.rdu2.scalelab.redhat.com, reached installation stage Writing image to disk: 22%
INFO Host: worker03.cdp.rdu2.scalelab.redhat.com, reached installation stage Writing image to disk: 30%
INFO Host: master03.cdp.rdu2.scalelab.redhat.com, reached installation stage Writing image to disk: 6%
INFO Host: worker10.cdp.rdu2.scalelab.redhat.com, reached installation stage Writing image to disk: 81%
INFO Host: worker10.cdp.rdu2.scalelab.redhat.com, reached installation stage Writing image to disk: 93%
INFO Host: infra03.cdp.rdu2.scalelab.redhat.com, reached installation stage Writing image to disk: 41%
INFO Host: worker01.cdp.rdu2.scalelab.redhat.com, reached installation stage Writing image to disk: 74%
INFO Host: master01.cdp.rdu2.scalelab.redhat.com, reached installation stage Writing image to disk: 32%
INFO Host: worker08.cdp.rdu2.scalelab.redhat.com, reached installation stage Writing image to disk: 43%
INFO Host: infra02.cdp.rdu2.scalelab.redhat.com, reached installation stage Writing image to disk: 71%
INFO Host: infra02.cdp.rdu2.scalelab.redhat.com, reached installation stage Writing image to disk: 79%
INFO Host: worker08.cdp.rdu2.scalelab.redhat.com, reached installation stage Writing image to disk: 60%
INFO Host: master01.cdp.rdu2.scalelab.redhat.com, reached installation stage Writing image to disk: 61%
INFO Host: worker07.cdp.rdu2.scalelab.redhat.com, reached installation stage Writing image to disk: 58%
INFO Host: infra01.cdp.rdu2.scalelab.redhat.com, reached installation stage Writing image to disk: 69%
INFO Host: worker08.cdp.rdu2.scalelab.redhat.com, reached installation stage Writing image to disk: 77%
INFO Host: infra01.cdp.rdu2.scalelab.redhat.com, reached installation stage Writing image to disk: 86%
INFO Host: infra01.cdp.rdu2.scalelab.redhat.com, reached installation stage Writing image to disk: 97%
INFO Host: master01.cdp.rdu2.scalelab.redhat.com, reached installation stage Writing image to disk: 93%
INFO Host: master01.cdp.rdu2.scalelab.redhat.com, reached installation stage Writing image to disk: 100%
INFO Host: worker08.cdp.rdu2.scalelab.redhat.com, reached installation stage Writing image to disk: 100%
INFO Host: worker08.cdp.rdu2.scalelab.redhat.com, reached installation stage Waiting for control plane
INFO Bootstrap Kube API Initialized
INFO Host: master03.cdp.rdu2.scalelab.redhat.com, reached installation stage Rebooting
INFO Host: master02.cdp.rdu2.scalelab.redhat.com, reached installation stage Rebooting
INFO Host: master01.cdp.rdu2.scalelab.redhat.com, reached installation stage Waiting for control plane: Waiting for
masters to join bootstrap control plane
INFO Host: master02.cdp.rdu2.scalelab.redhat.com, reached installation stage Configuring
INFO Host: master01.cdp.rdu2.scalelab.redhat.com, reached installation stage Waiting for bootkube
INFO Host: master03.cdp.rdu2.scalelab.redhat.com, reached installation stage Done
INFO Node master03.cdp.rdu2.scalelab.redhat.com has been rebooted 1 times before completing installation
INFO Node master02.cdp.rdu2.scalelab.redhat.com has been rebooted 1 times before completing installation
INFO Host: master01.cdp.rdu2.scalelab.redhat.com, reached installation stage Waiting for bootkube: waiting for ETCD
bootstrap to be complete
INFO Bootstrap configMap status is complete
INFO Bootstrap is complete
INFO cluster bootstrap is complete
DEBUG Still waiting for the cluster to initialize: Working towards 4.17.42: 791 of 901 done (87% complete)
DEBUG Still waiting for the cluster to initialize: Multiple errors are preventing progress:
DEBUG * Cluster operators authentication, image-registry, ingress, insights, kube-apiserver, machine-api, monitoring,
openshift-apiserver, openshift-samples, operator-lifecycle-manager-packageserver are not available
DEBUG * Could not update imagestream "openshift/driver-toolkit" (611 of 901): resource may have been deleted
DEBUG * Could not update oauthclient "console" (545 of 901): the server does not recognize this resource, check extension
API servers

```





```

DEBUG Route found in openshift-console namespace: console
DEBUG OpenShift console route is admitted
DEBUG Still waiting for the cluster to initialize: Cluster operators authentication, monitoring are not available
DEBUG Still waiting for the cluster to initialize: Cluster operators authentication, monitoring are not available
DEBUG Still waiting for the cluster to initialize: Cluster operator authentication is not available
INFO Cluster is installed
INFO Install complete!
INFO To access the cluster as the system:admin user when using 'oc', run
INFO     export KUBECONFIG=/root/ocp-agent-install/ocp-install-dir/auth/kubeconfig
INFO Access the OpenShift web-console here:
https://console-openshift-console.apps.cdp.rdu2.scalelab.redhat.com
INFO Login to the console with user: "kubeadmin", and password: "j8As6-TJCvN-AADQL-KFd7Z"

```

## NTP

Since we didn't use the OpenShift Assisted Installer, we will need to configure the NTP server that will be used. Until then, there may be NodeClockNotSynchronising messages logged.

Create the following .bu file and use the butane utility to convert it to the format used for the MachineConfig.

**vi 99-worker-chrony.bu**

```

variant: openshift
version: 4.16.0
metadata:
  name: 99-worker-chrony
  labels:
    machineconfiguration.openshift.io/role: worker
storage:
  files:
    - path: /etc/chrony.conf
      mode: 0644
      overwrite: true

```

```
contents:
  inline: |
    pool foreman.rdu2.scalelab.redhat.com iburst
    driftfile /var/lib/chrony/drift
    makestep 1.0 3
    rtcsync
    logdir /var/log/chrony
```

```
./butane 99-worker-chrony.bu -o 99-worker-chrony.yaml
```

```
cat 99-worker-chrony.yaml
```

```
# Generated by Butane; do not edit
apiVersion: machineconfiguration.openshift.io/v1
kind: MachineConfig
metadata:
  labels:
    machineconfiguration.openshift.io/role: worker
  name: 99-worker-chrony
spec:
  config:
    ignition:
      version: 3.4.0
    storage:
      files:
        - contents:
            compression: ""
            source:
data:,pool%20foreman.rdu2.scalelab.redhat.com%20iburst%20%0Adriftfile%20%2Fva
r%2Flib%2Fchrony%2Fdrift%0Amakestep%201.0%203%0Artcsync%0Alogdir%20%2Fvar%2F1
og%2Fchrony%0A
            mode: 420
            overwrite: true
            path: /etc/chrony.conf
```

Apply the MachineConfig using the following command:

```
oc create -f 99-worker-chrony.yaml
```

This will initiate a rolling reboot to update each worker node in the cluster. You can use the following commands to view the status of the nodes during the update:

```
oc get mcp
oc get nodes

oc describe machineconfig 99-worker-chrony
```

```
vi 99-master-chrony.bu
variant: openshift
version: 4.16.0
metadata:
  name: 99-master-chrony
  labels:
    machineconfiguration.openshift.io/role: master
storage:
  files:
    - path: /etc/chrony.conf
      mode: 0644
      overwrite: true
      contents:
        inline: |
          pool foreman.rdu2.scalelab.redhat.com iburst
          driftfile /var/lib/chrony/drift
          makestep 1.0 3
          rtcsync
          logdir /var/log/chrony
```

```
./butane 99-worker-chrony.bu -o 99-worker-chrony.yaml
```

```
cat 99-worker-chrony.yaml
# Generated by Butane; do not edit
apiVersion: machineconfiguration.openshift.io/v1
kind: MachineConfig
metadata:
  labels:
    machineconfiguration.openshift.io/role: master
  name: 99-master-chrony
spec:
  config:
    ignition:
      version: 3.4.0
    storage:
      files:
        - contents:
            compression: ""
            source:
data:,pool%20foreman.rdu2.scalelab.redhat.com%20iburst%20%0Adriftfile%20%2Fva
r%2Flib%2Fchrony%2Fdrift%0Amakestep%201.0%203%0Artcsync%0Alogdir%20%2Fvar%2F1
og%2Fchrony%0A
            mode: 420
            overwrite: true
            path: /etc/chrony.conf
```

```
oc create -f 99-master-chrony.yaml
```

This will initiate a rolling reboot to update each master node in the cluster. You can use the following commands to view the status of the nodes during the update:

```
oc get mcp  
oc get nodes
```

```
oc describe machineconfig 99-master-chrony
```

---

## ALTERNATIVE CONFIGURATION STEPS

### OPTIONAL Method to create VMs

STEP 1. Select Virtualization > Catalog > Template Catalog > Red Hat Enterprise Linux 9 VM

Project: pvcbase ▾

### Create new VirtualMachine

Select an option to create a VirtualMachine from.

InstanceTypes  Template catalog

Template project  
All projects ▾

All templates  Filter by keyword... 12 items

Default templates

User templates

Boot source available

- Operating system
  - CentOS
  - Fedora
  - Other
  - RHEL
  - Windows
- Workload
  - Desktop
  - High performance
  - Server

 CentOS Stream 8 VM centos-stream8-server-small	 CentOS Stream 9 VM centos-stream9-server-small <small>Source available</small>	 CentOS 7 VM centos7-server-small	 Fedora VM fedora-server-small <small>Source available</small>
 Red Hat Enterprise Linux 7 VM rhel7-server-small	 Red Hat Enterprise Linux 8 VM rhel8-server-small <small>Source available</small>	 Red Hat Enterprise Linux 9 VM rhel9-server-small <small>Source available</small>	 Microsoft Windows 10 VM windows10-desktop-medium
<small>Project openshift Boot source PVC Workload Server CPU 1 Memory 2 GiB</small>	<small>Project openshift Boot source PVC (auto import) Workload Server CPU 1 Memory 2 GiB</small>	<small>Project openshift Boot source PVC (auto import) Workload Server CPU 1 Memory 2 GiB</small>	<small>Project openshift Boot source PVC (auto import) Workload Server CPU 1 Memory 2 GiB</small>
<small>Project openshift Boot source PVC Workload Server CPU 1 Memory 2 GiB</small>	<small>Project openshift Boot source PVC (auto import) Workload Server CPU 1 Memory 2 GiB</small>	<small>Project openshift Boot source PVC (auto import) Workload Server CPU 1 Memory 2 GiB</small>	<small>Project openshift Boot source PVC Workload Desktop CPU 1 Memory 4 GiB</small>

**STEP 2. Edit the following parameters: VirtualMachine name, CPU | Memory, Disk size, Optional Parameters-CLOUD\_USER\_PASSWORD:**

### Red Hat Enterprise Linux 9 VM

×

**rhel9-server-small**

**Description**

Template for Red Hat Enterprise Linux 9 VM or newer. A PVC with the RHEL disk image must be available.

**Disk size \***

- 1508 +

GiB

---

**Documentation**

[Refer to documentation](#)

**Drivers**

Mount Windows drivers disk

---

**CPU | Memory**

32 CPU | 64 GiB Memory

**CLOUD\_USER\_PASSWORD**

---

**Network interfaces (1)**

Name	Network	Type

**Project**

pvcbase

**Public SSH key**

cdp-ssh

---

Start this VirtualMachine after creation

**Quick create VirtualMachine**

**Customize VirtualMachine**

**Cancel**

#### VM configuration parameters:

Parameter	Setting/Value
VirtualMachine name	cldr-mngr.redhat.local
Project name	pvc-base
Operating System	Red Hat Enterprise Linux 9 VM
CPU	32 CPU
Memory	64 GiB
Storage:	
Disk source	Blank
Disk Size	1500 GiB

CLOUD\_USER\_PASSWORD

r3dh4t!

### STEP 3. Select Customize Virtualmachine.

**IMPORTANT:** From the screen that follows, make sure that you edit several settings *before* you select **Create Virtual Machine**.

Project: pvcbase ▾

Catalog

Customize and create VirtualMachine  YAML

Template: Red Hat Enterprise Linux 9 VM

Overview    YAML    Scheduling    Environment    Network interfaces    Disks    Scripts    Metadata

---

Name	Network interfaces (1)		
ipaserver	Name	Network	Type
	default	Pod networking	Masquerade
Namespace	Disks (2)		
pvcbase	Name	Drive	Size
	rootdisk	Disk	250 GiB
	cloudinitdisk	Disk	-
Description	Hardware devices		
Not available	GPU devices		
Operating system	Not available		
Red Hat Enterprise Linux 9 VM	Host devices		
CPU   Memory	Not available		
16 CPU   32 GiB Memory			
Machine type	Headless mode		
pc-q35-rhel9.4.0	<input checked="" type="checkbox"/>		
Boot mode	Hostname		
UEFI (secure)			
<input checked="" type="checkbox"/> Start this VirtualMachine after creation			

**Create VirtualMachine**    Cancel

**STEP 4.** From the “*Network Interfaces*” screen, click on the three vertical dots to the far right of the “*default*” network interface, and click on “*Delete*”.

Project: pvc-ds

Catalog Customize and create VirtualMachine  YAML  
Template: Red Hat Enterprise Linux 9 VM

Overview YAML Scheduling Environment **Network interfaces** Disks Scripts Metadata

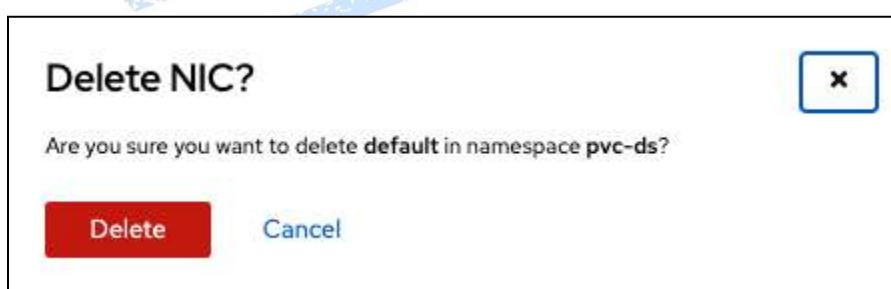
Add network interface

Name Filter Name Search by name...

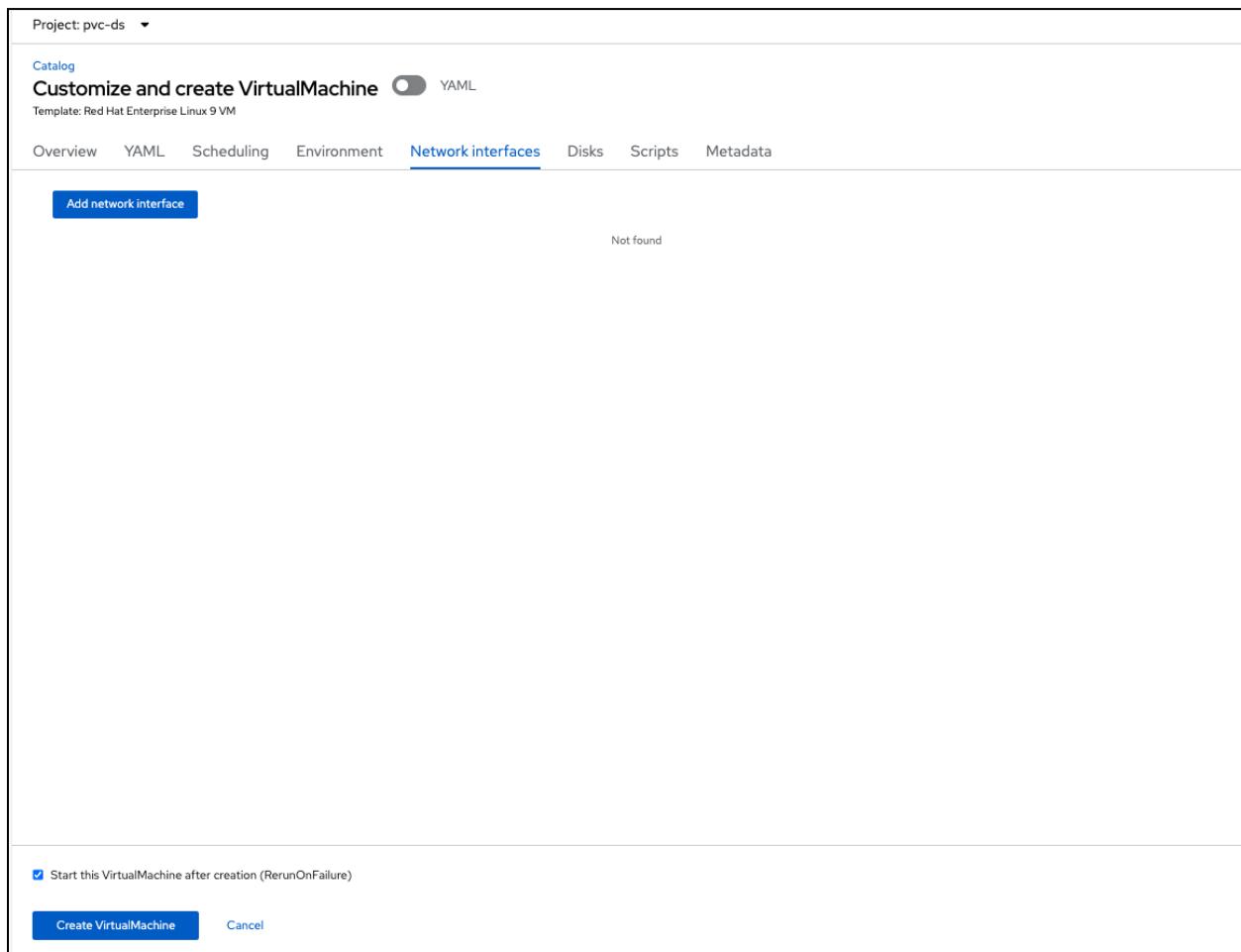
Name	Model	Network	Type	MAC address
default	virtio	Pod networking	Masquerade	02:7a:32:00:00:04

Start this VirtualMachine after creation (RerunOnFailure)

**STEP 5.** Then, confirm by clicking on “Delete”.



**STEP 6.** From the “*Network Interfaces*” screen, click on “*Add network interface*”.



**STEP 7.** From the “Add network interface” screen, enter the following parameters and then click on “Save”.

Network interface configuration parameters:

Parameter	Setting/Value
Network interface name	eth0
Model	virtio
Network	default/localnet-network

Add network interface

Name \*

Model

Network \*

default/localnet-network

Advanced

Save Cancel

STEP 8. Next, select the Scripts tab; and click on Edit in the Cloud-init section:

Project: pvcbase ▾

Catalog

**Customize and create VirtualMachine**

Template: Red Hat Enterprise Linux 9 VM

Overview YAML Scheduling Environment Network interfaces Disks Scripts Metadata

**Cloud-init**

You can use cloud-init to initialize the operating system with a specific configuration when the VirtualMachine is started.

The cloud-init service is enabled by default in Fedora and RHEL images. [Learn more](#)

User	Password	Network data
cloud-user	*****	Default

**Public SSH key** Linux only

Store the key in a project secret.  
The key will be stored after the machine is created

**cdp-ssh**

Start this VirtualMachine after creation

**STEP 9.** From the **Cloud-init** menu, select the checkbox next to **Add network data**, then enter the parameters as shown in the following screen for the **Ethernet name** and **Gateway address**.

Refer to the table provided at the beginning of this activity for the **IP address** to use for each VM. Select **Apply** when done.

## Cloud-init

X

You can use cloud-init to initialize the operating system with a specific configuration when the VirtualMachine is started.

The cloud-init service is enabled by default in Fedora and RHEL images.

[Learn more](#) 

Configure via:  Form view  Script

User \*

cloud-user

Password

r3dh4t!



Password for this username - [generate password](#)

Add network data

check this option to add network data section to the cloud-init script.

Ethernet name

eth0

IP addresses

10.6.131.20/26

Use commas to separate between IP addresses

Gateway address

10.6.131.62

**Apply**

[Cancel](#)

Use the following table as a reference to create the VirtualMachines for CDP Base:

Hostname	CPU	RAM	Disk	MAC Address	IP Address
cldr-mngr.redhat.local	32	64G	1.5T	02:7c:71:05:00:01	192.168.2.190/24
pvcbase-master.redhat.local	32	64G	1T	02:7c:71:05:00:02	192.168.2.191/24
pvcbase-worker1.redhat.local	32	64G	1T	02:7c:71:05:00:03	192.168.2.192/24
pvcbase-worker2.redhat.local	32	64G	1T	02:7c:71:05:00:04	192.168.2.193/24
pvcbase-worker3.redhat.local	32	64G	1T	02:7c:71:05:00:05	192.168.2.194/24
pvcbase-worker4.redhat.local	32	64G	1T	02:7c:71:05:00:06	192.168.2.195/24
pvcbase-worker5.redhat.local	32	64G	1T	02:7c:71:05:00:07	192.168.2.196/24

Project: pvcbase ▾

Catalog

Customize and create VirtualMachine  YAML

Template: Red Hat Enterprise Linux 9 VM

Overview   YAML   Scheduling   Environment   Network interfaces   Disks   Scripts   Metadata

**Cloud-init** [Edit](#)

You can use cloud-init to initialize the operating system with a specific configuration when the VirtualMachine is started.

The cloud-init service is enabled by default in Fedora and RHEL images. [Learn more](#)

User	Password	Network data
cloud-user	*****	Custom

**Public SSH key** [Edit](#)

Store the key in a project secret.

The key will be stored after the machine is created

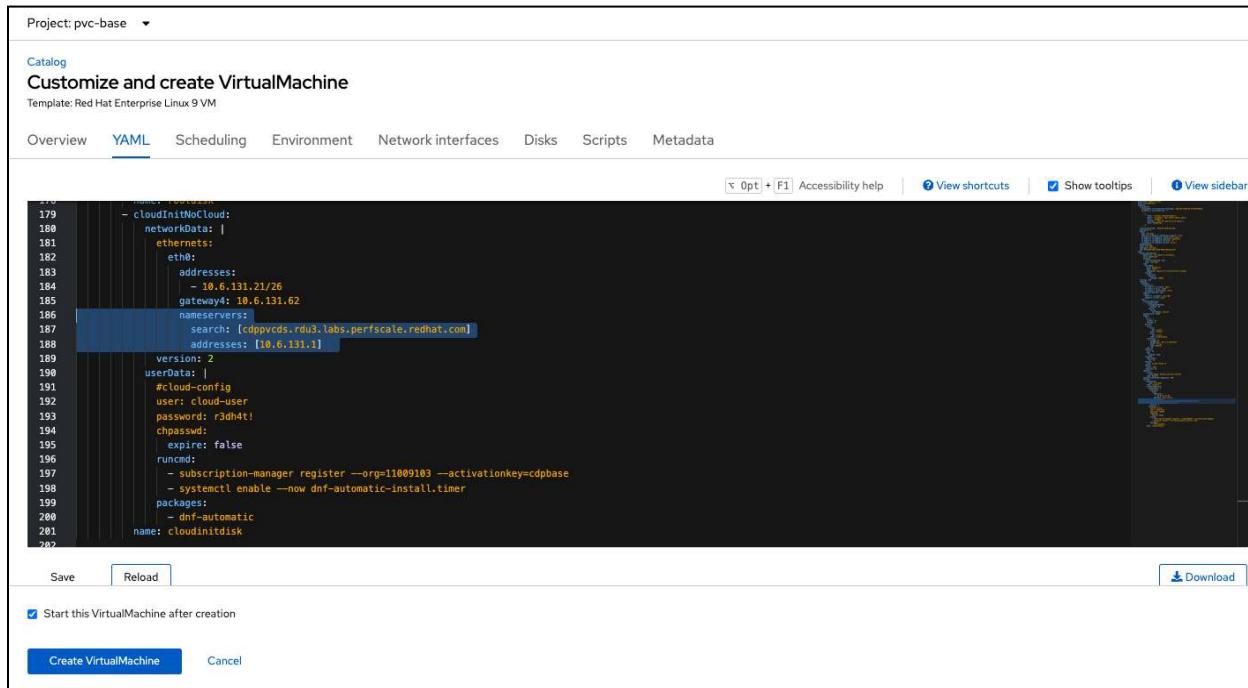
cdp-ssh

Start this VirtualMachine after creation

[Create VirtualMachine](#) [Cancel](#)

**STEP 10.** Select the **YAML** tab to edit the YAML definition for the VM. Add the highlighted **nameservers** section for DNS servers and search domains to match the provided section,

**IMPORTANT:** Make sure you click on *Save*, before you click on *Create VirtualMachine*. Otherwise, the changes made to the YAML will NOT be saved.



The screenshot shows the 'Customize and create VirtualMachine' page in Cloudera Manager. The 'YAML' tab is selected. The YAML code is displayed, with the 'CloudInitNoCloud' section highlighted in blue. The 'nameservers' and 'addresses' fields within this section are also highlighted in blue. At the bottom, there are 'Save' and 'Create VirtualMachine' buttons.

```
170     - cloudInitNoCloud:
171       networkData: |
172         eternets:
173           eth0:
174             addresses:
175               - 10.6.131.21/26
176             gateway4: 10.6.131.62
177             nameservers:
178               search: [cdppvcds.rdu3.labs.perfscal.redhat.com]
179               addresses: [10.6.131.1]
180
181             version: 2
182             userData: |
183               #cloud-config
184               user: cloud-user
185               password: r3dh4t!
186               chpasswd:
187                 expire: false
188               runcmd:
189                 - subscription-manager register --org=l1009103 --activationkey=cdpbase
190                 - systemctl enable --now dnf-automatic-install.timer
191               packages:
192                 - dnf-automatic
193               name: cloudinitdisk
194
195
196
197
198
199
200
201
202
```

Edit the YAML to include the following, under the **spec.template.spec.volumes.cloudInitNoCloud** section for the DNS servers and search domains:

```
spec:
  template:
    spec:
      volumes:
        - dataVolume:
            name: cldr-mngr
            name: rootdisk
        - cloudInitNoCloud:
            networkData: |
              eternets:
                eth0:
                  addresses:
                    - 192.168.2.190/26
                  gateway4: 192.168.2.1
                  nameservers:
                    search: [redhat.local]
                    addresses: [192.168.1.210]
            version: 2
            userData: |
              #cloud-config
              user: cloud-user
```

```
password: r3dh4t!
chpasswd:
    expire: false
runcmd:
    - subscription-manager register --org=11009103
--activationkey=cdpbase
    - systemctl enable --now dnf-automatic-install.timer
packages:
    - dnf-automatic
name: cloudinitdisk
```

## NOTE: In case the VM does not register with Red Hat Insights, you can use the following command to manually register the VM: rhc connect

Example:

```
# rhc connect --activation-key cdpbase --organization 11009103
```

**STEP 11.** When you have finished editing the VM, select the *Create VirtualMachine* button.

**### IMPORTANT:** When editing the YAML, make sure you click on the **Save** button before you click on *Create VirtualMachine*. Otherwise, the changes made in the YAML editor will NOT be saved.

**IMPORTANT:** Some of the parameters, such as CloudInit only take effect after the *first* boot of the virtual machine.

If there are settings that still need to be made before this, then make sure to unselect the option to '*Start this VirtualMachine after creation*'.

Start this VirtualMachine after creation

**Create VirtualMachine**

Cancel

**STEP 12.** From the **Overview** tab, you can view the VM provisioning status.

## ADD NODES TO THE CLUSTER

### 6.4.3. Adding nodes to your cluster

vi nodes-config.yaml file

```

hosts:
- hostname: f36-h03-000-r640
  rootDeviceHints:
    deviceName: /dev/sda
  interfaces:
    - name: eno2np1
      macAddress: bc:97:e1:79:19:f1 # Change for each host
  networkConfig:
    interfaces:
      - name: eno2np1
        type: ethernet
        state: up
        mac-address: bc:97:e1:79:19:f1 # Change for each host
        ipv4:
          enabled: true
          address:
            - ip: 10.1.48.113 # Change for each host
              prefix-length: 23
        dhcp: false

```

```
dns-resolver:
  config:
    server:
      - 10.1.49.1
routes:
  config:
    - destination: 0.0.0.0/0
      next-hop-address: 10.1.49.254
      next-hop-interface: eno2np1
      table-id: 254
- hostname: f36-h05-000-r640
rootDeviceHints:
  deviceName: /dev/sda
interfaces:
  - name: eno2np1
    macAddress: bc:97:e1:7c:f6:31 # Change for each host
networkConfig:
  interfaces:
    - name: eno2np1
      type: ethernet
      state: up
      mac-address: bc:97:e1:7c:f6:31 # Change for each host
      ipv4:
        enabled: true
        address:
          - ip: 10.1.48.114 # Change for each host
            prefix-length: 23
        dhcp: false
  dns-resolver:
    config:
      server:
        - 10.1.49.1
routes:
  config:
    - destination: 0.0.0.0/0
      next-hop-address: 10.1.49.254
      next-hop-interface: eno2np1
      table-id: 254
- hostname: f36-h07-000-r640
rootDeviceHints:
  deviceName: /dev/sda
interfaces:
  - name: eno2np1
    macAddress: bc:97:e1:79:2c:11 # Change for each host
networkConfig:
  interfaces:
```

```
- name: eno2np1
  type: ethernet
  state: up
  mac-address: bc:97:e1:79:2c:11 # Change for each host
  ipv4:
    enabled: true
    address:
      - ip: 10.1.48.115 # Change for each host
        prefix-length: 23
    dhcp: false
  dns-resolver:
    config:
      server:
        - 10.1.49.1
  routes:
    config:
      - destination: 0.0.0.0/0
        next-hop-address: 10.1.49.254
        next-hop-interface: eno2np1
        table-id: 254
- hostname: f36-h09-000-r640
  rootDeviceHints:
    deviceName: /dev/sda
  interfaces:
    - name: eno2np1
      macAddress: bc:97:e1:78:e8:61 # Change for each host
  networkConfig:
    interfaces:
      - name: eno2np1
        type: ethernet
        state: up
        mac-address: bc:97:e1:78:e8:61 # Change for each host
        ipv4:
          enabled: true
          address:
            - ip: 10.1.48.116 # Change for each host
              prefix-length: 23
          dhcp: false
    dns-resolver:
      config:
        server:
          - 10.1.49.1
    routes:
      config:
        - destination: 0.0.0.0/0
          next-hop-address: 10.1.49.254
```

```
        next-hop-interface: eno2np1
        table-id: 254
- hostname: f36-h10-000-r640
  rootDeviceHints:
    deviceName: /dev/sda
  interfaces:
    - name: eno2np1
      macAddress: bc:97:e1:78:e7:a1 # Change for each host
networkConfig:
  interfaces:
    - name: eno2np1
      type: ethernet
      state: up
      mac-address: bc:97:e1:78:e7:a1 # Change for each host
    ipv4:
      enabled: true
      address:
        - ip: 10.1.48.117 # Change for each host
          prefix-length: 23
      dhcp: false
  dns-resolver:
    config:
      server:
        - 10.1.49.1
  routes:
    config:
      - destination: 0.0.0.0/0
        next-hop-address: 10.1.49.254
        next-hop-interface: eno2np1
        table-id: 254
- hostname: f36-h11-000-r640
  rootDeviceHints:
    deviceName: /dev/sda
  interfaces:
    - name: eno2np1
      macAddress: bc:97:e1:69:92:31 # Change for each host
networkConfig:
  interfaces:
    - name: eno2np1
      type: ethernet
      state: up
      mac-address: bc:97:e1:69:92:31 # Change for each host
    ipv4:
      enabled: true
      address:
        - ip: 10.1.48.118 # Change for each host
```

```
prefix-length: 23
dhcp: false
dns-resolver:
  config:
    server:
      - 10.1.49.1
routes:
  config:
    - destination: 0.0.0.0/0
      next-hop-address: 10.1.49.254
      next-hop-interface: eno2np1
      table-id: 254
```

```
[root@ipaserver]# oc adm node-image create nodes-config.yaml
[root@ipaserver]# ll *.iso
-rw-r--r--. 1 root root 1326184448 Nov  4 18:01 node.x86_64.iso
cp node.x86_64.iso /var/www/html/.
```

Verify that the discovery image is available at:  
[http://10.1.49.1/node.x86\\_64.iso](http://10.1.49.1/node.x86_64.iso)

**BADFISH**

Configure the servers to boot from the Discovery image:

```
export USER=quads
export PASS=rdu2@4618
```

```
dnf install -y podman
```

```
vi workerlist
10.1.64.196
10.1.65.226
10.1.65.96
10.1.65.139
10.1.64.17
10.1.65.3
10.1.64.139
10.1.65.220
```

10.1.65.210

```
for HOST in `cat ~/workerlist`;do podman run -it --rm quay.io/quads/badfish  
-H $HOST -u $USER -p $PASS --power-off;done  
  
for HOST in `cat ~/workerlist`;do podman run -it --rm quay.io/quads/badfish  
-H $HOST -u $USER -p $PASS --unmount-virtual-media;done  
  
for HOST in `cat ~/workerlist`;do podman run -it --rm quay.io/quads/badfish  
-H $HOST -u $USER -p $PASS --mount-virtual-media  
http://10.1.49.1/node.x86_64.iso;done  
  
for HOST in `cat ~/workerlist`;do podman run -it --rm quay.io/quads/badfish  
-H $HOST -u $USER -p $PASS --boot-to-virtual-media;done  
  
for HOST in `cat ~/workerlist`;do podman run -it --rm quay.io/quads/badfish  
-H $HOST -u $USER -p $PASS --power-on;done  
  
for HOST in `cat ~/workerlist`;do podman run -it --rm quay.io/quads/badfish  
-H $HOST -u $USER -p $PASS --power-state;done
```

## NodeNetworkConfigurationPolicy

```
apiVersion: nmstate.io/v1  
kind: NodeNetworkConfigurationPolicy  
metadata:  
  name: f36-h03-000-r640  
  #namespace:  
spec:  
  nodeSelector:  
    node-role.kubernetes.io/worker: ""  
    kubernetes.io/hostname: f36-h03-000-r640  
  desiredState:  
    interfaces:  
      - name: eno1np0  
        description: iSCSI network  
        type: ethernet  
        ethernet:  
          state: down  
          ipv4:  
            enabled: false  
          ipv6:  
            enabled: false  
---
```

```
apiVersion: nmstate.io/v1
kind: NodeNetworkConfigurationPolicy
metadata:
  name: f36-h05-000-r640
  #namespace:
spec:
  nodeSelector:
    node-role.kubernetes.io/worker: ""
    kubernetes.io/hostname: f36-h05-000-r640
  desiredState:
    interfaces:
      - name: eno1np0
        description: iSCSI network
        type: ethernet
        ethernet:
          state: down
        ipv4:
          enabled: false
        ipv6:
          enabled: false
---
apiVersion: nmstate.io/v1
kind: NodeNetworkConfigurationPolicy
metadata:
  name: f36-h07-000-r640
  #namespace:
spec:
  nodeSelector:
    node-role.kubernetes.io/worker: ""
    kubernetes.io/hostname: f36-h07-000-r640
  desiredState:
    interfaces:
      - name: eno1np0
        description: iSCSI network
        type: ethernet
        ethernet:
          state: down
        ipv4:
          enabled: false
        ipv6:
          enabled: false
---
apiVersion: nmstate.io/v1
kind: NodeNetworkConfigurationPolicy
metadata:
  name: f36-h09-000-r640
```

```
#namespace:  
spec:  
  nodeSelector:  
    node-role.kubernetes.io/worker: ""  
    kubernetes.io/hostname: f36-h09-000-r640  
  desiredState:  
    interfaces:  
      - name: eno1np0  
        description: iSCSI network  
        type: ethernet  
        ethernet:  
          state: down  
        ipv4:  
          enabled: false  
        ipv6:  
          enabled: false  
---  
apiVersion: nmstate.io/v1  
kind: NodeNetworkConfigurationPolicy  
metadata:  
  name: f36-h10-000-r640  
  #namespace:  
spec:  
  nodeSelector:  
    node-role.kubernetes.io/worker: ""  
    kubernetes.io/hostname: f36-h10-000-r640  
  desiredState:  
    interfaces:  
      - name: eno1np0  
        description: iSCSI network  
        type: ethernet  
        ethernet:  
          state: down  
        ipv4:  
          enabled: false  
        ipv6:  
          enabled: false  
---  
apiVersion: nmstate.io/v1  
kind: NodeNetworkConfigurationPolicy  
metadata:  
  name: f36-h11-000-r640  
  #namespace:  
spec:  
  nodeSelector:  
    node-role.kubernetes.io/worker: ""
```

```
kubernetes.io/hostname: f36-h11-000-r640
desiredState:
  interfaces:
    - name: eno1np0
      description: iSCSI network
      type: ethernet
      ethernet:
        state: down
      ipv4:
        enabled: false
      ipv6:
        enabled: false
---
apiVersion: nmstate.io/v1
kind: NodeNetworkConfigurationPolicy
metadata:
  name: f36-h13-000-r640
  #namespace:
spec:
  nodeSelector:
    node-role.kubernetes.io/worker: ""
    kubernetes.io/hostname: f36-h13-000-r640
  desiredState:
    interfaces:
      - name: eno1np0
        description: iSCSI network
        type: ethernet
        ethernet:
          state: down
        ipv4:
          enabled: false
        ipv6:
          enabled: false
---
apiVersion: nmstate.io/v1
kind: NodeNetworkConfigurationPolicy
metadata:
  name: f36-h14-000-r640
  #namespace:
spec:
  nodeSelector:
    node-role.kubernetes.io/worker: ""
    kubernetes.io/hostname: f36-h14-000-r640
  desiredState:
    interfaces:
      - name: eno1np0
```

```

description: iSCSI network
type: ethernet
ethernet:
state: down
ipv4:
  enabled: false
ipv6:
  enabled: false
---
apiVersion: nmstate.io/v1
kind: NodeNetworkConfigurationPolicy
metadata:
  name: f36-h15-000-r640
  #namespace:
spec:
  nodeSelector:
    node-role.kubernetes.io/worker: ""
    kubernetes.io/hostname: f36-h15-000-r640
  desiredState:
    interfaces:
      - name: eno1np0
        description: iSCSI network
        type: ethernet
ethernet:
        state: down
        ipv4:
          enabled: false
        ipv6:
          enabled: false
---

```

## OPTIONAL METHOD TO CREATE VM YAML

**STEP 6.** Switch to the OpenShift console, and navigate to Virtualization > Catalog > Create a VM > from Catalog > RHEL 9. Enter the VM configuration parameters from the table provided below, and then select “*Customize VirtualMachine*” to further customize the VM.

The screenshot shows the 'Create new VirtualMachine' dialog in the Cloudera Manager interface. On the left, there's a sidebar with 'Template project' dropdowns for 'All projects' and 'Default templates'. Below these are sections for 'Default templates' (CentOS Stream 9 VM, Red Hat Enterprise Linux 9 VM, Microsoft Windows Server 2019 VM), 'Operating system' (CentOS, Fedora, Other, RHEL, Windows), 'Workload' (Desktop, High performance, Server), and 'CPU | Memory' (4 CPU | 16 GiB Memory). The main right panel is titled 'Red Hat Enterprise Linux 9 VM' and contains the following configuration fields:

- Template info**: Operating system is set to 'Red Hat Enterprise Linux 9 VM'.
- Storage**: Boot from CD is selected, and the URL is set to 'http://192.168.1.210/discovery\_image\_ocp-ds.iso'.
- CD source**: URL is set to 'http://192.168.1.210/discovery\_image\_ocp-ds.iso'.
- Description**: Template for Red Hat Enterprise Linux 9 VM or newer. A PVC with the RHEL disk image must be available.
- Documentation**: Refer to documentation link.
- CPU | Memory**: Set to 4 CPU | 16 GiB Memory.
- Network interfaces**: Default network settings.
- Disk size**: Disk size is set to 2 GiB.
- Disk source**: Disk source is set to 'Blank'.
- Disk size**: Disk size is set to 120 GiB.
- Drivers**: Mount Windows drivers disk checkbox is unchecked.
- Quick create VirtualMachine**: VirtualMachine name is 'master0.ocp-ds.redhat.local', Project is 'pvc-ds', Public SSH key is 'cdp-ssh'.
- Start this VirtualMachine after creation (RerunOnFailure)**: This checkbox is checked.
- Buttons**: Quick create VirtualMachine, Customize VirtualMachine, Cancel.

VM configuration parameters:

Parameter	Setting/Value
VirtualMachine name	master0.cdp.rdu2.scalelab.redhat.com
Project name	cdp-ds
Operating System	Red Hat Enterprise Linux 9 VM
CPU	4 CPU
Memory	16 GiB
Storage:	
Boot from CD	Selected (enabled)
CD source	URL (creates PVC)
Image URL	http://10.1.49.1/discovery_image_cdpd.s.iso

Disk Size	2 GiB
Disk source	Blank
Disk Size	120 GiB

**STEP 7.** From the Customize and create VirtualMachine screen, select the “*Network Interfaces*” tab.

The screenshot shows the 'Customize and create VirtualMachine' interface. The 'Overview' tab is active. On the left, there's a sidebar with sections like 'Name', 'Namespace', 'Description', 'Operating system', 'CPU | Memory', 'Machine type', 'Boot mode', 'Start in pause mode', and 'Workload profile'. On the right, there are several tables and sections: 'Network interfaces (1)' showing one interface named 'default' (Pod networking, Masquerade); 'Disks (3)' listing 'rootdisk' (Disk, 120 GiB), 'installation-cdrom' (CD-ROM, 2 GiB), and 'cloudinitdisk' (Disk, -); 'Hardware devices' (GPU devices, Not available); 'Host devices' (Not available); 'Headless mode' (disabled); 'Hostname' (master0.ocp-ds.redhat.local); and 'Guest system log access' (disabled). At the bottom, there's a checkbox for 'Start this VirtualMachine after creation (RerunOnFailure)' and two buttons: 'Create VirtualMachine' and 'Cancel'.

**STEP 8.** From the “*Network Interfaces*” screen, click on the three vertical dots to the far right of the “*default*” network interface, and click on “*Delete*”.

Project: pvc-ds

Catalog Customize and create VirtualMachine  YAML  
Template: Red Hat Enterprise Linux 9 VM

Overview YAML Scheduling Environment **Network interfaces** Disks Scripts Metadata

Add network interface

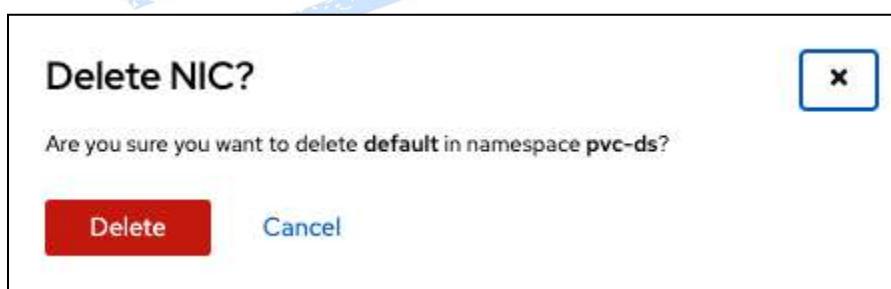
Name Filter Model Network Type MAC address

Name	Model	Network	Type	MAC address
default	virtio	Pod networking	Masquerade	02:7a:32:00:00:04

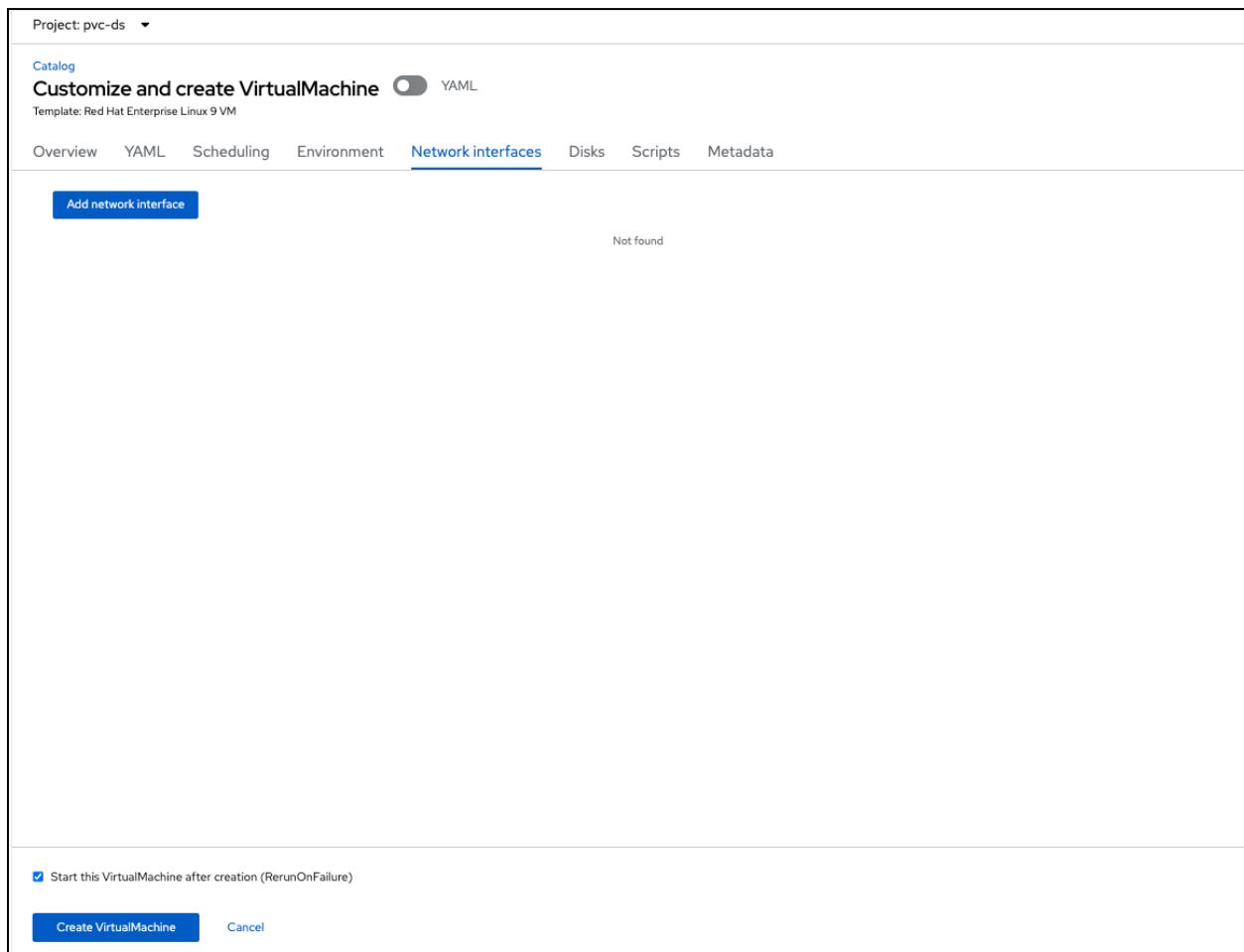
Start this VirtualMachine after creation (RerunOnFailure)

Create VirtualMachine Cancel

**STEP 9.** Then, confirm by clicking on “Delete”.



**STEP 10.** From the “*Network Interfaces*” screen, click on “*Add network interface*”.



**STEP 11.** From the “Add network interface” screen, enter the following parameters and then click on “Save”.

Network interface configuration parameters:

Parameter	Setting/Value
Network interface name	eth0
Model	virtio
Network	default/localnet-network

Add network interface

Name \*

Model

Network \*

Advanced

Save Cancel

STEP 12. Select the “Overview” tab, and confirm the configuration settings. Then click on “Create VirtualMachine”.

Project: pvc-ds ▾

Catalog Customize and create VirtualMachine  YAML

Template: Red Hat Enterprise Linux 9 VM

**Overview** **YAML** Scheduling Environment Network interfaces Disks Scripts Metadata

<b>Name</b>	<b>Network interfaces (1)</b>		
master0.ocp-ds.redhat.local	Name	Network	Type
	eth0	default/localnet-network	Bridge
<b>Namespace</b>	<b>Disks (3)</b>		
pvc-ds	Name	Drive	Size
	rootdisk	Disk	120 GiB
	installation-cdrom	CD-ROM	2 GiB
	cloudinitdisk	Disk	-
<b>Description</b>	<b>Hardware devices</b>		
Not available	GPU devices	Not available	
<b>Operating system</b>	<b>Host devices</b>		
Red Hat Enterprise Linux 9 VM	Not available		
<b>CPU   Memory</b>	<b>Headless mode</b>		
4 CPU   16 GiB Memory			
<b>Machine type</b>	<b>Hostname</b>		
pc-q35-rhel9.4.0	master0.ocp-ds.redhat.local		
<b>Boot mode</b>	<b>Guest system log access</b>		
UEFI (secure)			
<b>Start in pause mode</b>			
<b>Workload profile</b>			
Server			

Start this VirtualMachine after creation (RerunOnFailure)

**Create VirtualMachine** **Cancel**