FlashStack with Red Hat OpenShift Container and Virtualization Platform using Cisco UCS X-Series

Manual Configuration Deployment for FlashStack with Red Hat Bare Metal OpenShift Container and Virtualization Platform using Cisco UCS X-Series, Pure Storage FlashArray//XL170, and Portworx Enterprise

Published: November 2024

In partnership with:

About the Cisco Validated Design Program

The Cisco Validated Design (CVD) program consists of systems and solutions designed, tested, and documented to facilitate faster, more reliable, and more predictable customer deployments. For more information, go to: http://www.cisco.com/go/designzone.

Executive Summary

The FlashStack solution is a validated, converged infrastructure developed jointly by Cisco and Pure Storage. The solution offers a predesigned data center architecture that incorporates compute, storage, and network to reduce IT risk by validating the architecture and helping ensure compatibility among the components. The FlashStack solution is successful because of its ability to evolve and incorporate both technology and product innovations in the areas of management, compute, storage, and networking. This document covers the deployment details of Red Hat OpenShift Container Platform (OCP) and Red Hat OpenShift Virtualization on FlashStack Bare Metal infrastructure. Some of the most important advantages of FlashStack with Red Hat OpenShift Container Platform and Red hat OpenShift Virtualization on Bare Metal are:

- Simplify IT operations with a unified platform: With Red Hat OpenShift Container Platform and Virtualization, containers and virtual machines can be run side-by-side within a cluster avoiding operational, complexity and challenges of maintaining separate platforms for running these workloads.
- Consistent infrastructure configuration: Cisco Intersight and UCS help bring up the entire server farm with standardized methods and consistent configuration tools that helps to improve the compute availability, avoid human configuration errors and achieve higher Return on Investments (ROI).
- Simpler and programmable Infrastructure: The entire underlying infrastructure can be configured using infrastructure as code delivered using Red Hat Ansible.
- End-to-End 100Gbps Ethernet: This solution offers 100Gbps connectivity among the servers and storage using 5th Gen Cisco UCS VIC, Fabric Interconnect and 100Gbps adapters on storage controllers.
- Single Storage platform for both virtual and containerized workloads: Using the Pure Storage FlashArray as backend storage, Portworx Enterprise by Pure Storage provides persistent and container-native data platform with enterprise grade features such as snapshots, clones, replication, compression, de-duplication and so on, for the workloads running inside containers and virtual machines.

In addition to the compute-specific hardware and software innovations, integration of the Cisco Intersight cloud platform with Pure Storage FlashArray and Cisco Nexus delivers monitoring, orchestration, and workload optimization capabilities for different layers of the FlashStack solution.

If you are interested in understanding the FlashStack design and deployment details, including configuration of various elements of design and associated best practices, refer to Cisco Validated Designs for FlashStack here: https://www.cisco.com/c/en/us/solutions/design-zone/data-center-design-guides/data-center-design-guides-all.html#FlashStack

Note: This document serves as the deployment guide for the solution. The design guide for the solution will be available soon.

Solution Overview

This chapter contains the following:

Introduction

The FlashStack solution with Red Hat OpenShift on Bare Metal configuration represents a cohesive and flexible infrastructure solution that combines computing hardware, networking, and storage resources into a single, integrated architecture. Designed as a collaborative effort between Cisco and Pure Storage, this converged infrastructure platform is engineered to deliver high levels of efficiency, scalability, and performance, suitable for a multitude of datacenter workloads. By standardizing on a validated design, organizations can accelerate deployment, reduce operational complexities, and confidently scale their IT operations to meet evolving business demands. The FlashStack architecture leverages Cisco's Unified Computing System (Cisco UCS) servers, Cisco Nexus networking, and Pure's innovative storage systems, providing a robust foundation for containerized, virtualized and non-virtualized environments.

Audience

The intended audience for this document includes, but is not limited to IT architects, sales engineers, field consultants, professional services, IT managers, IT engineers, partners, and customers who are interested to take the advantage of an infrastructure built to deliver IT efficiency and enable IT innovation.

Purpose of this Document

This document provides deployment guidance for bringing up the FlashStack solution with Red Hat OpenShift Container Platform and virtualization on Bare Metal Infrastructure. This document introduces various design elements and explains various considerations and best practices for a successful Red Hat OpenShift deployment.

Highlights of this Solution

The highlights of this solution are:

- Red Hat OpenShift Bare Metal deployment on FlashStack solution enabling customers to run both containerized and virtualized workloads running alongside each other with in a cluster.
- Configuration guidelines for compute, network, storage and OCP, including the OpenShift Virtualization on the entire stack.
- Portworx Enterprise for enabling container-native persistent storage and Portworx's dynamic console plugin for OpenShift for monitoring different storage resources running on OpenShift clusters.
- Support for NVIDIA GPUs for running AI/ML inference workloads.

Technology Overview

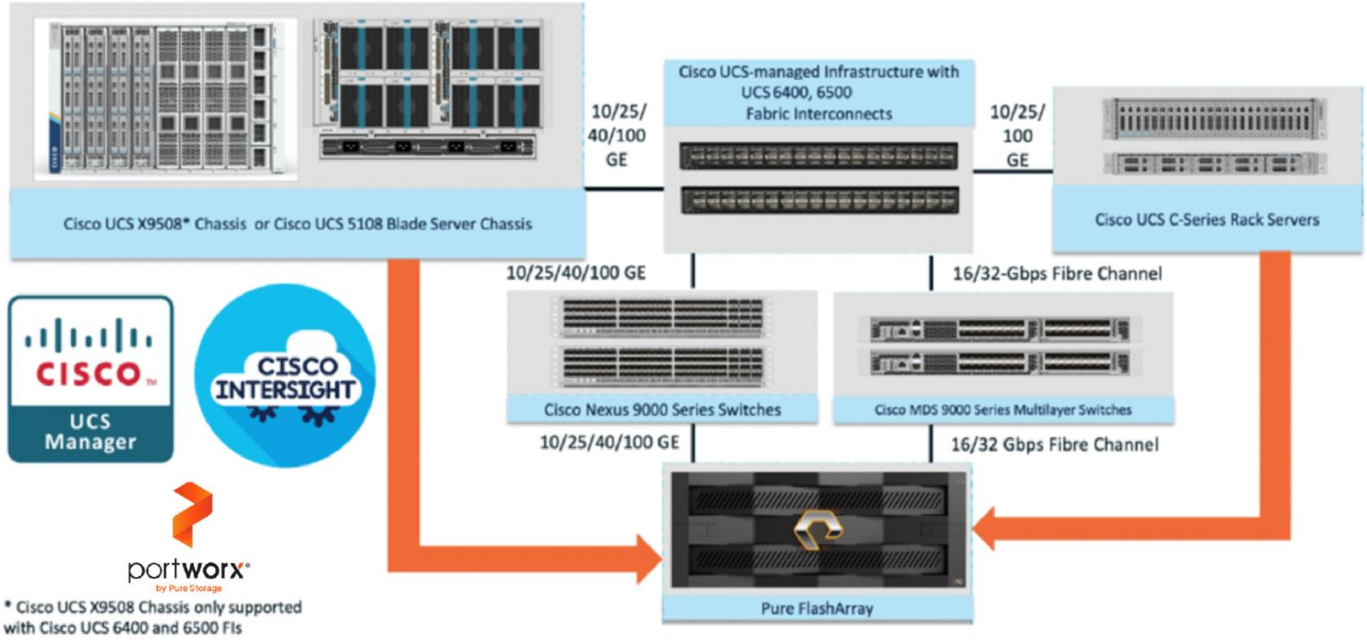
This chapter contains the following:

- FlashStack Components
- Benefits of Portworx Enterprise with OpenShift Virtualization
- Benefits of Portworx Enterprise with FlashArray

FlashStack Components

The FlashStack architecture was jointly developed by Cisco and Pure Storage. All FlashStack components are integrated, allowing customers to deploy the solution quickly and economically while eliminating many of the risks associated with researching, designing, building, and deploying similar solutions from the foundation. One of the main benefits of FlashStack is its ability to maintain consistency at scale. Figure 1 illustrates the series of hardware components used for building the FlashStack architectures.

Figure 1. FlashStack Components


Cisco UCS X9508* Chassis or Cisco UCS 5108 Blade Server Chassis
10/25/
40/100
GE
Cisco UCS-managed Infrastructure with
UCS 6400, 6500
Fabric Interconnects
10/25/
100
GE
Cisco UCS C-Series Rack Servers
10/25/40/100 GE
16/32-Gbps Fibre Channel
UCS Manager
CISCO INTERSIGHT
Cisco Nexus 9000 Series Switches
Cisco MDS 9000 Series Multilayer Switches
10/25/40/100 GE
16/32 Gbps Fibre Channel
portworx
by Pure Storage
Pure FlashArray
* Cisco UCS X9508 Chassis only supported
with Cisco UCS 6400 and 6500 FIs

All FlashStack components are integrated, so you can deploy the solution quickly and economically while eliminating many of the risks associated with researching, designing, building, and deploying similar solutions from the foundation. One of the main benefits of FlashStack is its ability to maintain consistency at scale. Each of the component families shown in Figure 1 (Cisco UCS, Cisco Nexus, Cisco MDS, Portworx by Pure Storage and Pure Storage FlashArray systems) offers platform and resource options to scale up or scale out the infrastructure while supporting the same features and functions.

Refer to the Appendix for more detailed information of the above components used in this solution.

Benefits of Portworx Enterprise with OpenShift Virtualization

Portworx with Red Hat OpenShift Virtualization and KubeVirt enhances data management for virtual machines and containers by offering integrated, enterprise-grade storage. Includes simplified storage operations through Kubernetes, high availability and resiliency across environments, advanced disaster recovery options, and automated scaling capabilities. This integration supports a unified infrastructure where traditional and modern workloads coexist, providing flexibility in deployment across diverse infrastructures and ensuring robust data security.

Portworx + Stork offers capabilities like VM migration between clusters, Synchronous Disaster Recovery, Ability to backup and restore VMs running on Red Hat OpenShift to comply with the service level agreements.

Benefits of Portworx Enterprise with FlashArray

Portworx on FlashArray offers flexible storage deployment options for Kubernetes. Using FlashArray as cloud drives enables automatic volume provisioning, cluster expansion, and supports PX Backup and Autopilot. Direct Access volumes allow for efficient on-premises storage management, offering file system operations, IOPS, and snapshot capabilities. Multi-tenancy features isolate storage access per user, enhancing security in shared environments.

Portworx on FlashArray enhances Kubernetes environments with robust data reduction, resiliency, simplicity, and support. It lowers storage costs through deduplication, compression, and thin provisioning, providing 2-10x data reduction. FlashArray's reliable infrastructure ensures high availability, reducing server-side rebuilds. Portworx simplifies Kubernetes deployment with minimal configuration and end-to-end visibility via Pure1. Additionally, unified support, powered by Pure1 telemetry, offers centralized, proactive assistance for both storage hardware and Kubernetes services, creating an efficient and scalable solution for enterprise needs.

Deployment Hardware and Software

This chapter contains the following:

- Design Considerations
- Requirements

Design Considerations

The FlashStack Datacenter with Cisco UCS and Cisco Intersight meets the following general design requirements:

- Resilient design across all the layers of infrastructure with no single point of failure
- Scalable design with the flexibility to add compute capacity, storage, or network bandwidth as needed
- Modular design that can be replicated to expand and grow as the needs of the business grow
- Flexible design that can support different models of various components with ease
- Simplified design with the ability to integrate and automate with external automation tools
- AI-Ready design to support required NVIDIA GPUs for running AI/ML based workloads
- Cloud-enabled design which can be configured, managed, and orchestrated from the cloud using GUI or APIs

To deliver a solution which meets all these design requirements, various solution components are connected and configured as explained in later sections.

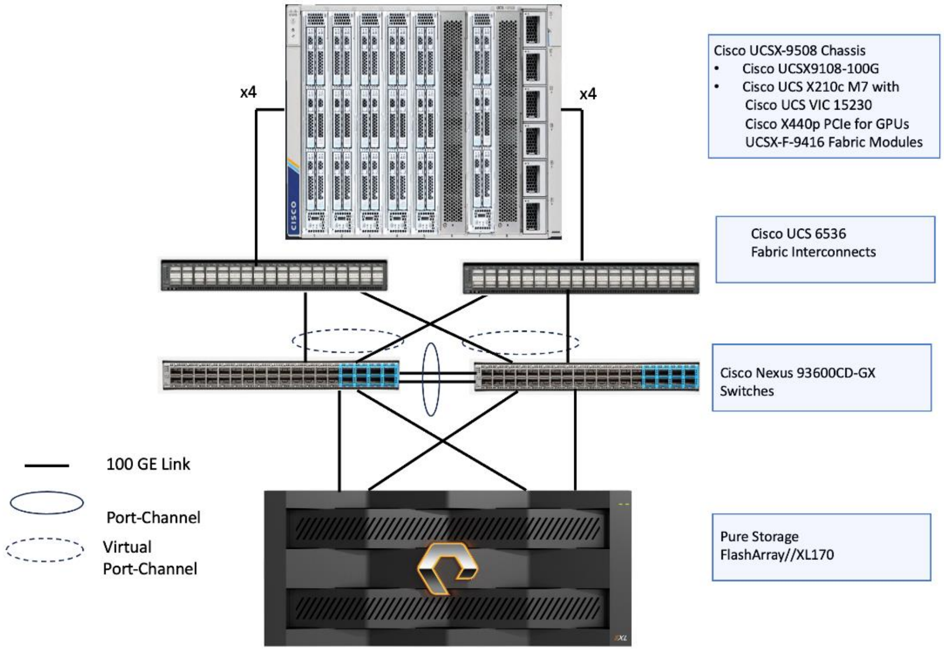
Requirements

Physical Topology

FlashStack with Cisco UCS X-Series supports both Ethernet and Fibre Channel (FC) storage access. This Red Hat OpenShift Bare Metal deployment is built over Ethernet-based design. For this solution, Cisco Nexus 93699CD-GX switches are used to provide the connectivity between the servers and storage. ISCSI configuration on the Cisco UCS and Pure Storage FlashArray is utilized to set up storage access. The physical components and connectivity details for Ethernet -based design are covered below.

Figure 2 shows the physical topology and network connections used for this Ethernet-based FlashStack design.

Figure 2. Physical Topology



The reference hardware configuration includes:

- One Cisco UCS X9508 chassis, equipped with a pair of Cisco UCS X9108 100G IFMs, contains six Cisco UCS X210c M7 compute nodes and two Cisco UCS X440p PCIe nodes each with two NVDIA L40S GPUs. Other configurations of servers with and without GPUs are also supported. Each compute node is equipped with fifth-generation Cisco VIC card 15231 providing 100-G ethernet connectivity on each side of the fabric. A pair of Fabric Modules installed at the rear side of the chassis enables connectivity between the X440p PCIe nodes and X210c M7 nodes.
- Cisco fifth-generation 6536 fabric interconnects are used to provide connectivity to the compute nodes installed in the chassis.
- High-speed Cisco NXOS-based Nexus C93600CD-GX switching design to support up to 100 and 400-GE connectivity.
- Pure Storage FlashArray//X170 with 100Gigabit Ethernet connectivity. FlashArray introduces native block and file architectures built on a single global storage pool, simplifying management and treating both services as equal citizens–the first truly Unified Block and File Platform in the market.

The software components consist of:

- Cisco Intersight platform to deploy, maintain, and support the FlashStack components.
- Cisco Intersight Assist virtual appliance to help connect the Pure Storage FlashArray and Cisco Nexus Switches with the Cisco Intersight platform to enable visibility into these platforms from Intersight.
- Red Hat OpenShift Container Platform for providing a consistent hybrid cloud foundation for building and scaling containerized and virtualized applications.
- Portworx by Pure Storage (Portworx Enterprise) data platform for providing enterprise grade storage for containerized and virtualized workloads hosted on OpenShift platform.

Red Hat OpenShift Container Platform on Bare Metal Server Configuration

A simple Red Hat OpenShift cluster consists of at least five servers – 3 Master or Control Plane Nodes and 2 or more Worker or compute Nodes where applications and VMs are run. In this lab validation 3 Worker Nodes were utilized. Based on published Red Hat requirements, the three Master Nodes were configured with 64GB RAM, and the three Worker Nodes were configured with 1024GB memory to handle containerized applications and Virtual Machines. Each Node was booted from RAID1 disk created using two M.2 SSD drives. Also, the servers paired with X440p PCIe Nodes were configured as Workers. From a networking perspective, both the Masters and the Workers were configured with a single vNIC with UCS Fabric Failover in the Bare Metal or Management VLAN. The Workers were configured with extra NICs (vNICs) to allow storage attachment to the Workers and for virtual machine's management and storage traffic purposes.

Each worker had two additional vNICs with the iSCSI A and B VLANs configured as native to allow iSCSI persistent storage attachment and future iSCSI boot. Each worker node is also configured with three additional vNICs for Virtual Machine's management traffic and direct storage access using In-Guest iSCSI. The following sections provide more details on the network configuration of worker nodes.

FlashStack Cabling

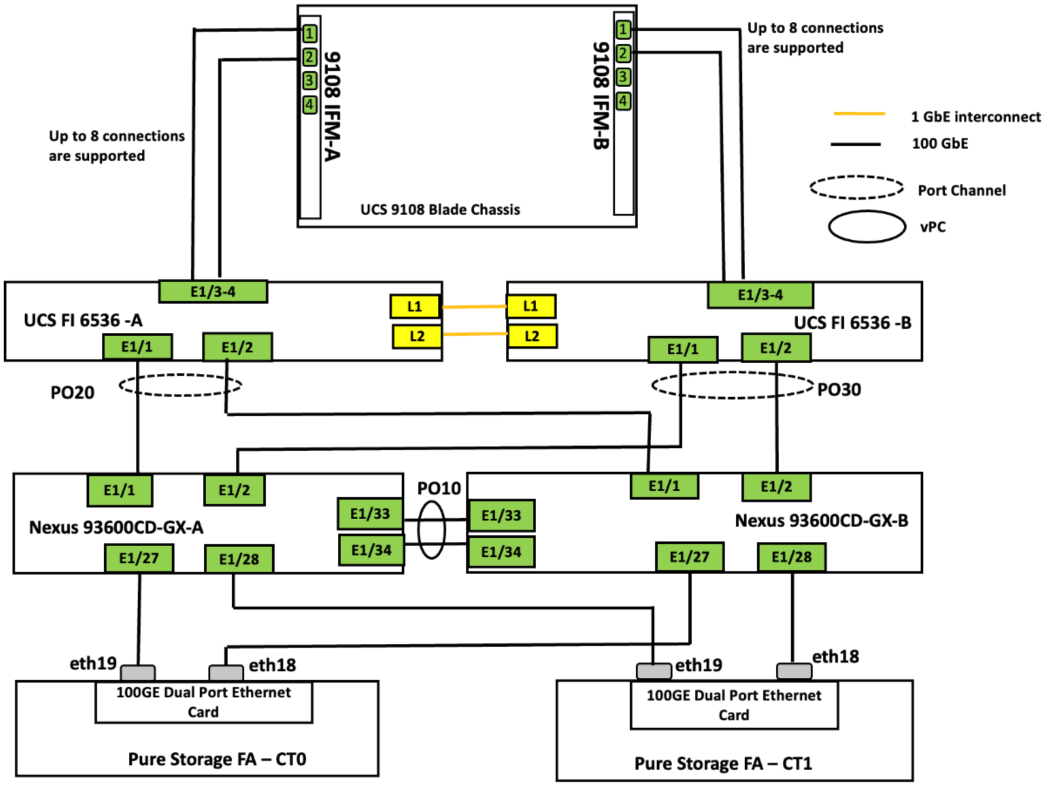
The information in this section is provided as a reference for cabling the physical equipment in a FlashStack environment. Figure 3 illustrates how all the hardware components are connected.

Note: This document assumes that out-of-band management ports are plugged into an existing management infrastructure at the deployment site. These interfaces will be used in various configuration steps.

Note: Be sure to use the cabling directions in this section as a guide.

Figure 3 details the cable connections used in the validation lab for the FlashStack topology based on the 5th generation Cisco UCS 6536 fabric interconnect. On each side of the Fabric Interconnect, two 100G ports on each UCS 9108 100G IFMs are used to connect the Cisco UCS X9508 chassis to the Fabric Interconnects. Two 100G port on each FI are connected to the pair of Cisco Nexus 93600CD-GX switches that are configured with a vpc domain. Each Pure Storage FlashArray//XI170 controller is connected to the pair of Nexus 93600CD-GX switches over 100G ports. Additional 1Gb management connections will be needed for one or more out-of-band network switches that sit apart from the FlashStack infrastructure. Each Cisco UCS fabric interconnect and Cisco Nexus switch is connected to the out-of-band network switches, and each Pure Storage FlashArray controller has a connection to the out-of-band network switches. Layer 3 network connectivity is required between the Out-of-Band (OOB) and In-Band (IB) Management Subnets.

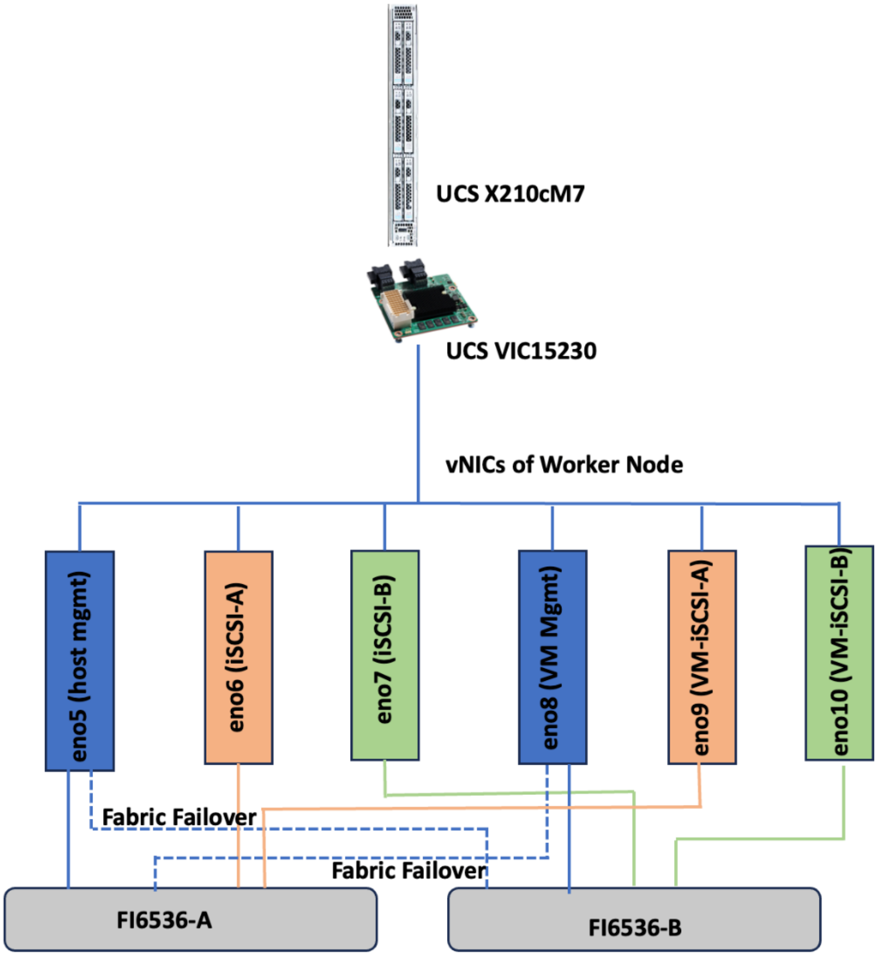
Figure 3. FlashStack Cabling



Worker Node Network Configuration

The worker node is configured with six vNICs. First three vNICs (eno5, eno6 and eno7) are used by worker node for OpenShift cluster management traffic and storage traffic using iSCSI protocol. While the last three vNICs (eno8, eno9 and eno10) will be used for Virtual machines networking. Eno6 will be used for virtual machine management traffic while eno9 and eno10 are used for enabling direct storage access to the Pure Storage FlashArray using in-guest iscsi. vNICs eno5 and eno8 are configured with Fabric-Failover option. vNICs eno6 and eno9 are used for iSCSI storage traffic via Fabric-A while vNICs eno7 and eno10 are used for iSCSI storage traffic via Fabric-B. Figure 4 illustrates the vNIC configuration of OpenShift worker nodes that are planned for hosting both PODs and Virtual Machines.

Figure 4. Worker Node vNIC configuration



Note: The control plane (master node) will just have one vNIC eno5 with Fabric-Failover option enabled.

VLAN Configuration

Table 2 lists the VLANs configured for setting up the FlashStack environment along with their usage.

Table 1. VLAN Usage

VLAN ID	NAME	Usage	IP Subnet used in this deployment
2	Native-VLAN	VLAN2 is used as native VLAN instead of default VLAN1	
1060	OOB-Mgmt-VLAN	Out-of-band management VLAN to connect management port for various devices	10.106.0.24/0 BW: 10.106.0.254
1061	IB-Mgmt-VLAN	Routable Bare Metal VLAN used for OpenShift cluster and node management	10.106.1.0/24 GW: 10.106.1.254
3010	OCP-iSCSI-A	Used for OpenShift iSCSI persistent storage via Fabric-A Also, for In-Guest iSCSI storage access directly from VMs	192.168.51.0/24

VLAN ID	NAME	Usage	IP Subnet used in this deployment
3020	OCP-iSCSI-B	Used for OpenShift iSCSI persistent storage via Fabric-B Also, for In-Guest iSCSI storage access directly from VMs	192.168.52.0/24
1062	VM-Mgmt-VLAN	Routable VLAN used for VM management network	10.106.2.0/24 GW: 10.106.2.254

Table 2 lists the infrastructure services running on either virtual machines or bar mental servers required for deployment as outlined in the document. All these services are hosted on pre-existing infrastructure with in the FlashStack

Table 2. Infrastructure services

Service Description	VLAN	IP Address
AD/DNS-1 & DHCP	1061	10.106.1.21
AD/DNS-2	1061	10.106.1.22
OCP installer/bastion node	1061	10.106.1.23
Cisco Intersight Assist Virtual Appliance	1061	10.106.1.24

Software Revisions

The FlashStack Solution with Red Hat OpenShift on Bare Metal infrastructure configuration is built using the following components.

Table 3 lists the required software revisions for various components of the solution.

Table 3. Software Revisions

Layer	Device	Image Bundle version	Comments
Compute	Pair of Cisco UCS Fabric Interconnect - 6530	4.3(4.240066)	
	6x Cisco UCS X210 M7 with Cisco VIC 15230	5.2(2.240053)	
Network	Cisco Nexus 93699CD-GX-NX-OS	10.3(5)(M)	
Storage	Pure Storage FlashArray Purity //FA	6.6.10	
Software	Red Hat OpenShift	4.16	
	Portworx Enterprise	3.1.6	
	Cisco Intersight Assist Appliance	1.1.1-0	

Layer	Device	Image Bundle version	Comments
	NVIDIA L40S Driver	550.90.07	

Network Switch Configuration

This chapter contains the following:

- Physical Connectivity
- Cisco Nexus Switch Manual Configuration
- Claim Cisco Nexus Switches into Cisco Intersight

Physical Connectivity

Physical cabling should be completed by following the diagram and table references in section FlashStack Cabling.

The following procedures describe how to configure the Cisco Nexus 93600CD-GX switches for use in a FlashStack environment. This procedure assumes the use of Cisco Nexus 9000 10.1(2), the Cisco suggested Nexus switch release at the time of this validation.

Note: The procedure includes the setup of NTP distribution on both the mgmt0 port and the in-band management VLAN. The interface-vlan feature and ntp commands are used to set this up. This procedure also assumes that the default VRF is used to route the in-band management VLAN.

Note: This document assumes that initial day-0 switch configuration is already done using switch console ports and ready to use the switches using their management IPs.

Cisco Nexus Switch Manual Configuration

Procedure 1. Enabling features on Cisco Nexus A and Cisco Nexus B

Step 1. Log into both Nexus switches as admin using ssh.

Step 2. Enable the switch features as described below:

```
config t
feature nxapi
cfs eth distribute
feature udld
feature interface-vlan
feature netflow
feature hsrp
feature lacp
feature vpc
feature lldp
```

Procedure 2. Set Global Configurations on Cisco Nexus A and Cisco Nexus B

Step 1. Log into both Nexus switches as admin using ssh.

Step 2. Run the following commands to set the global configurations:

```
spanning-tree port type edge bpduguard default
spanning-tree port type edge bpdufilter default
spanning-tree port type network default
system default switchport
system default switchport shutdown
port-channel load-balance src-dst l4port
ntp server <Global-ntp-server-ip> use-vrf default
ntp master 3
clock timezone <timezone> <hour-offset> <minute-Offset>
clock summer-time <timezone> <start-weekk> <start-day> <start-month> <start-time> <end-week> <end-day> <enb-
month> <end-time> <offset-minutes>
ip route 0.0.0.0/0 <IB-Mgmt-VLAN-gatewayIP>
```

```
copy run start
```

Note: It is important to configure the local time so that logging time alignment and any backup schedules are correct. For more information on configuring the timezone and daylight savings time or summer time, please see https://www.cisco.com/c/en/us/td/docs/dcn/nx-os/nexus9000/102x/configuration/fundamentals/cisco-nexus-9000-nx-os-fundamentals-configuration-guide-102x/m-basic-device-management.html#task_1231769

Sample clock commands for the United States Eastern timezone are:

clock timezone EST -5 0

clock summer-time EDT 2 Sunday March 02:00 1 Sunday November 02:00 60

Procedure 3. Create VLANs on Cisco Nexus A and Cisco Nexus B

Step 1. From the global configuration mode, run the following

```
Vlan <oob-mgmt-vlan-id>
name OOB-Mgmt-VLAN
Vlan <ib-mgmt-vlan-id>
name IB-Mgmt-VLAN
Vlan <native-vlan-id>
name Native-VLAN
Vlan <ocp-iscsi-a-vlan-id>
name OCP-iSCSI-A
Vlan <ocp-iscsi-b-vlan-id>
name OCP-iSCSI-B
Vlan <vm-mgmt-vlan-id>
name VM-Mgmt-VLAN
```

Procedure 4. Add NTP Distribution Interface

Cisco Nexus - A

Step 1. From the global configuration mode, run the following commands:

```
interface vlan <ib-mgmt-vlan-id>
ip address <switch-a-ntp-ip>/<ib-mgmt-vlan-netmask-length>
no shut
exit
ntp peer <switch-b-ntp-ip> use-vrf default
```

Cisco Nexus - B

Step 2. From the global configuration mode, run the following commands:

```
interface vlan <ib-mgmt-vlan-id>
ip address <switch-b-ntp-ip>/<ib-mgmt-vlan-netmask-length>
no shut
exit
ntp peer <switch-a-ntp-ip> use-vrf default
```

Procedure 5. Define Port Channels on Cisco Nexus A and Cisco Nexus B

Cisco Nexus - A and B

Step 1. From the global configuration mode, run the following commands:

```

interface port-channel 10
description vPC Peer Link
switchport mode trunk
switchport trunk native vlan 2
switchport trunk allowed vlan 1060-1062,3010,3020
spanning-tree port type network
```

```
interface port-channel 20
switchport mode trunk
switchport trunk native vlan 2
switchport trunk allowed vlan 1060-1062,3010,3020
spanning-tree port type edge trunk
mtu 9216

interface port-channel 30
switchport mode trunk
switchport trunk native vlan 2
switchport trunk allowed vlan 1060-1062,3010,3020
spanning-tree port type edge trunk
mtu 9216

### Optional: The below port channels is for connecting the Nexus switches to the existing customer network
interface port-channel 106
description connectting-to-customer-Core-Switches
switchport mode trunk
switchport trunk native vlan 2
switchport trunk allowed vlan 1060-1062
spanning-tree port type normal
mtu 9216
```

Procedure 6. Configure Virtual Port Channel Domain on Nexus A and Cisco Nexus B

Cisco Nexus - A

Step 1. From the global configuration mode, run the following commands:

```
vpc domain <nexus-vpc-domain-id>
peer-switch
role priority 10
peer-keepalive destination 10.106.0.6 source 10.106.0.5
delay restore 150
peer-gateway
auto-recovery
ip arp synchronize
```

Cisco Nexus - B

Step 1. From the global configuration mode, run the following commands:

```
vpc domain <nexus-vpc-domain-id>
peer-switch
role priority 20
peer-keepalive destination 10.106.0.5 source 10.106.0.6
delay restore 150
peer-gateway
auto-recovery
ip arp synchronize
```

Procedure 7. Configure individual Interfaces

Cisco Nexus-A

Step 1. From the global configuration mode, run the following commands:

```
interface Ethernet1/1
description FI6536-A-uplink-Eth1
channel-group 20 mode active
no shutdown

interface Ethernet1/2
description FI6536-B-uplink-Eth1
channel-group 30 mode active
no shutdown

interface Ethernet1/33
description Nexus-B-33
```

```
channel-group 10 mode active
no shutdown

interface Ethernet1/34
description Nexus-B-34
channel-group 10 mode active
no shutdown

## Optional: Configuration for interfaces that connected to the customer existing management network
interface Ethernet1/35/1
description customer-Core-1:Eth1/37
channel-group 106 mode active
no shutdown

interface Ethernet1/35/2
description customer-Core-2:Eth1/37
channel-group 106 mode active
no shutdown
```

Cisco Nexus-B

Step 1. From the global configuration mode, run the following commands:

```
interface Ethernet1/1
description FI6536-A-uplink-Eth2
channel-group 20 mode active
no shutdown

interface Ethernet1/2
description FI6536-B-uplink-Eth2
channel-group 30 mode active
no shutdown

interface Ethernet1/33
description Nexus-A-33
channel-group 10 mode active
no shutdown

interface Ethernet1/34
description Nexus-A-34
channel-group 10 mode active
no shutdown

## Optional: Configuration for interfaces that connected to the customer existing management network
interface Ethernet1/35/1
description customer-Core-1:Eth1/38
channel-group 106 mode active
no shutdown

interface Ethernet1/35/2
description customer-Core-2:Eth1/38
channel-group 106 mode active
no shutdown
```

Procedure 8. Update the port channels

Cisco Nexus-A and B

Step 1. From the global configuration mode, run the following commands:

```
interface port-channel 10
vpc peer-link
interface port-channel 20
vpc 20
interface port-channel 30
vpc 30
interface port-channel 106
vpc 106

copy run start
```

Step 2. The following commands can be used to check for correct switch configuration:

```
Show run
show vpc
show port-channel summary
show ntp peer-status
show cdp neighbours
show lldp neighbours
show udld neighbours
show run int
show int
show int status
```

Cisco Nexus iSCSI Configuration

Procedure 1. Configure Interfaces for Pure Storage on Cisco Nexus and Cisco Nexus B

Cisco Nexus - A

Step 1. From the global configuration mode, run the following commands:

```
interface Ethernet1/27
description PureXL170-ct0-eth19
switchport access vlan 3010
spanning-tree port type edge
mtu 9216
no shutdown

interface Ethernet1/28
description PureXL170-ct1-eth19
switchport access vlan 3010
spanning-tree port type edge
mtu 9216
no shutdown
copy run start
```

Cisco Nexus - B

Step 1. From the global configuration mode, run the following commands:

```
interface Ethernet1/27
description PureXL170-ct0-eth18
switchport access vlan 3020
spanning-tree port type edge
mtu 9216
no shutdown

interface Ethernet1/28
description PureXL170-ct1-eth18
switchport access vlan 3020
spanning-tree port type edge
mtu 9216
no shutdown
copy run start
```

Claim Cisco Nexus Switches into Cisco Intersight

Cisco Nexus switches can be claimed into the Cisco Intersight either using Cisco Intersight Assist or Direct claim using Device ID and Claim Codes.

This section provides the steps to claim the Cisco Nexus switches using Cisco Intersight Assist.

Note: This procedure assumes that Cisco Intersight is already hosted outside the OpenShift cluster and claimed into the Intersight.com.

Procedure 1. Claiming Cisco Nexus Switches into Cisco Intersight using Cisco Intersight Assist

Cisco Nexus - A

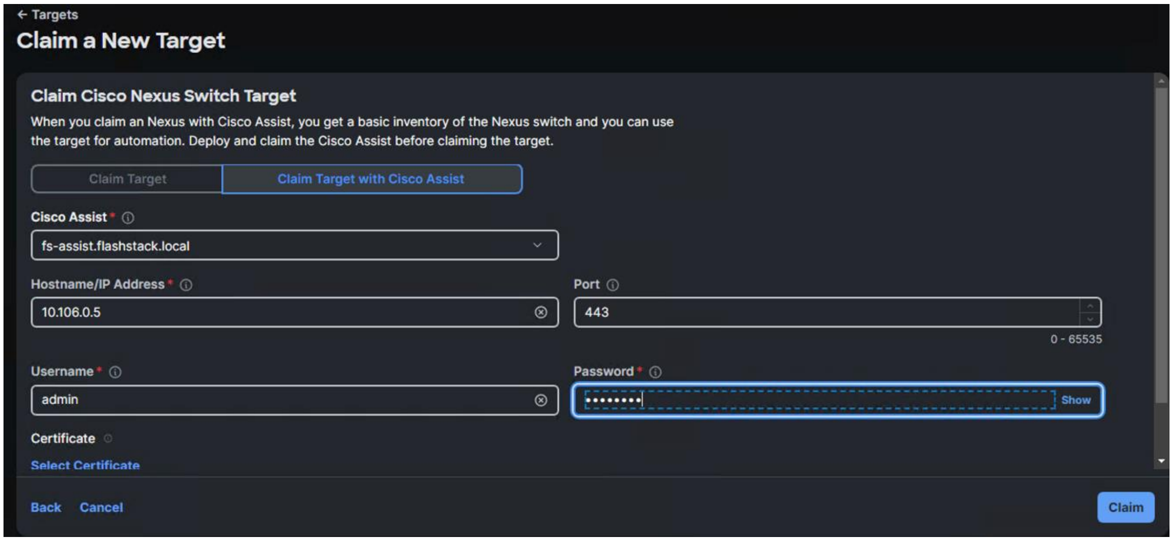
Step 1. Log into Nexus Switches and confirm **nxapi** feature is enabled.

```
show nxapi
nxapi enabled
NXAPI timeout 10
HTTPS Listen on port 443
Certificate Information:
    Issuer:   issuer=C = US, ST = CA, L = San Jose, O = Cisco Systems Inc., OU = dcnxos, CN = nxos
    Expires:  Sep 12 06:08:58 2024 GMT
```

Step 2. Log into Cisco Intersight with your login credentials. From the drop-down list located on the left top, select **System.**

Step 3. Under **Admin**, click **Target** and click **Claim a New Target**. Under Categories, select **Network**, then click **Cisco Nexus Switch** and then click **Start**.

Step 4. Select the Cisco Assist name which is already deployed and configured. Provide the Cisco Nexus Switch management IP address, username and password details and click **Claim.**



← Targets

Claim a New Target

Claim Cisco Nexus Switch Target

When you claim an Nexus with Cisco Assist, you get a basic inventory of the Nexus switch and you can use the target for automation. Deploy and claim the Cisco Assist before claiming the target.

Claim Target | Claim Target with Cisco Assist

Cisco Assist *: fs-assist.flashstack.local

Hostname/IP Address *: 10.106.0.5

Port: 443 (0 - 65535)

Username *: admin

Password *: •••••••• Show

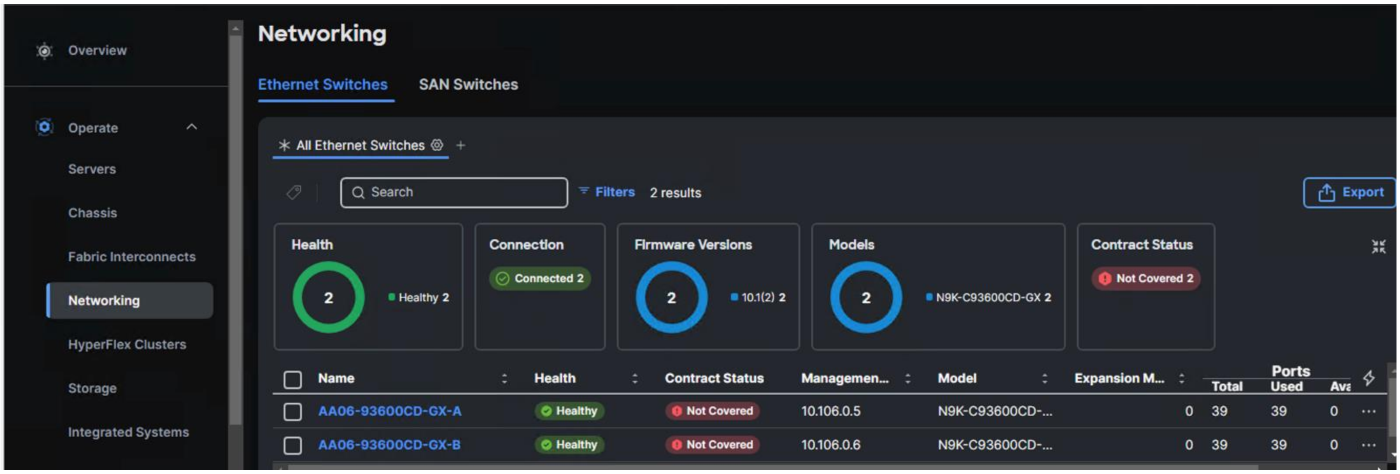
Certificate

Select Certificate

Back Cancel Claim

Step 5. Repeat steps 1 through 4 to claim the remaining Switch B.

Step 6. When the storage is successfully claim, from the top left drop-down list, select **Infrastructure Services**. Under Operate, click **Networking** tab. On the right you will find the newly claimed Cisco Nexus switch details and browse through the Switches for viewing the inventory details.



Networking

Ethernet Switches | SAN Switches

All Ethernet Switches

Search | Filters | 2 results | Export

Health: 2 — Healthy 2

Connection: Connected 2

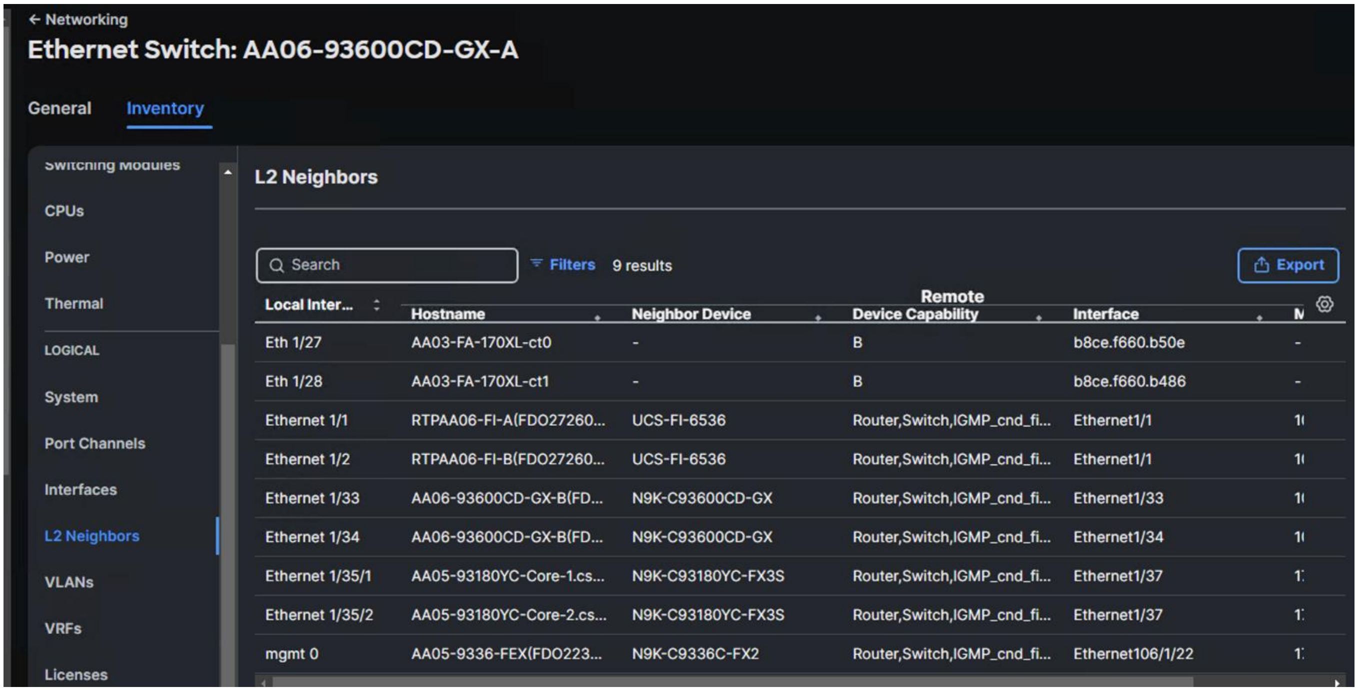
Firmware Versions: 2 — 10.1(2) 2

Models: 2 — N9K-C93600CD-GX 2

Contract Status: Not Covered 2

Name	Health	Contract Status	Managemen...	Model	Expansion M...	Ports Total	Ports Used	Ports Ava
AA06-93600CD-GX-A	Healthy	Not Covered	10.106.0.5	N9K-C93600CD-...	0	39	39	0
AA06-93600CD-GX-B	Healthy	Not Covered	10.106.0.6	N9K-C93600CD-...	0	39	39	0

The following screenshot shows the L2 neighbors of the Cisco Nexus Switch-A:



Ethernet Switch: AA06-93600CD-GX-A

L2 Neighbors

9 results

Local Inter...			Remote		
	Hostname	Neighbor Device	Device Capability	Interface	N
Eth 1/27	AA03-FA-170XL-ct0	-	B	b8ce.f660.b50e	-
Eth 1/28	AA03-FA-170XL-ct1	-	B	b8ce.f660.b486	-
Ethernet 1/1	RTPAA06-FI-A(FDO27260...	UCS-FI-6536	Router,Switch,IGMP_cnd_fi...	Ethernet1/1	1
Ethernet 1/2	RTPAA06-FI-B(FDO27260...	UCS-FI-6536	Router,Switch,IGMP_cnd_fi...	Ethernet1/1	1
Ethernet 1/33	AA06-93600CD-GX-B(FD...	N9K-C93600CD-GX	Router,Switch,IGMP_cnd_fi...	Ethernet1/33	1
Ethernet 1/34	AA06-93600CD-GX-B(FD...	N9K-C93600CD-GX	Router,Switch,IGMP_cnd_fi...	Ethernet1/34	1
Ethernet 1/35/1	AA05-93180YC-Core-1.cs...	N9K-C93180YC-FX3S	Router,Switch,IGMP_cnd_fi...	Ethernet1/37	1
Ethernet 1/35/2	AA05-93180YC-Core-2.cs...	N9K-C93180YC-FX3S	Router,Switch,IGMP_cnd_fi...	Ethernet1/37	1
mgmt 0	AA05-9336-FEX(FDO223...	N9K-C9336C-FX2	Router,Switch,IGMP_cnd_fi...	Ethernet106/1/22	1

Cisco Intersight Managed Mode Configuration for Cisco UCS

This chapter contains the following:

- Fabric Interconnect Domain Profile and Policies
- Server Profile Templates and Polices
- Create Pools
- vNIC Templates and vNICs
- Ethernet Adapter Policy for iSCSI Traffic
- Storage Policy
- Compute Configuration Policies
- Management Configuration Policies

The procedures in this section describe how to configure a Cisco UCS domain for use in a base FlashStack environment. A Cisco UCS domain is defined as a pair for Cisco UCS FIs and all the servers connected to it. These can be managed using two methods: UCSM and IMM. The procedures outlined below are for Cisco UCS Fabric Interconnects running in Intersight managed mode (IMM).

The Cisco Intersight platform is a management solution delivered as a service with embedded analytics for Cisco and third-party IT infrastructures. The Cisco Intersight Managed Mode (also referred to as Cisco IMM or Intersight Managed Mode) is an architecture that manages Cisco Unified Computing System (Cisco UCS) fabric interconnect-attached systems through a Redfish-based standard model. Cisco Intersight managed mode standardizes both policy and operation management for Cisco UCS C-Series M7 and Cisco UCS X210c M7 compute nodes used in this deployment guide.

Note: This deployment guide assumes an Intersight account is already created, configured with required licenses and ready to use. Intersight Default Resource Group and Default Organizations are used for claiming all the physical components of the FlashStack solution.

Note: This deployment guide assumes that the initial day-0 configuration of Fabric Interconnects is already done in the IMM mode and claimed into the Intersight account.

Procedure 1. Fabric Interconnect Domain Profile and Policies

Step 1. Log into the Intersight portal and select **Infrastructure Service**. On the left select Profiles then under **Profiles** select **UCS Domain Profiles**.

Step 2. Click **Create UCS Domain Profile** to create a new domain profile for Fabric Interconnects. Under the General tab, select the Default Organization, enter name and descriptions of the profile.

Step 3. Click **Next** to go to UCS Domain Assignment. Click **Assign Later**.

Step 4. Click **Next** to go to VLAN & VSAN Configuration.

Step 5. Under VLAN & VSAN Configuration > VLAN Configuration, click **select Policy** and click **Create New**.

Step 6. On the Create VLAN page, General tab, enter a name (**AA06-FI-VLANs**)and click **Next** to go to Policy Details.

Step 7. To add a VLAN, click **Add VLANs**.

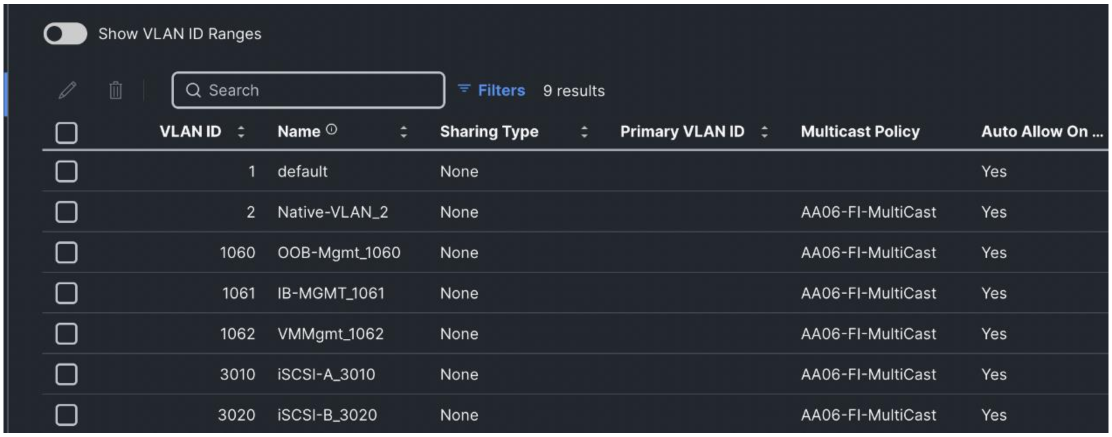
Step 8. For the Prefix, enter the VLAN name as OOB-Mgmt-VLAN. For the VLAN ID, enter the VLAN id 1061. Leave Auto Allow on Uplinks enabled and Enable VLAN Sharing disabled.

Step 9. Under Multicast Policy, click **Select Policy** and select **Create New** to create a Multicast policy.

Step 10. On the **Create Multicast Policy** page, enter name (**AA06-FI-MultiCast**) of the policy and click **Next** to go to **Policy Details**. Leave the **Snooping State** and **Source IP Proxy state** checked/enabled and click **Create.** Now select the newly created Multicast policy.

Step 11. Repeat steps 1 through 10 to add all the required VLANs to the VLAN policy.

Step 12. After adding all the VLANs, click **Set Native VLAN ID** and enter native VLANs (for example 2) and click **Create**. The following screenshot shows the VLANs used for this solution:



VLAN ID	Name	Sharing Type	Primary VLAN ID	Multicast Policy	Auto Allow On ...
1	default	None			Yes
2	Native-VLAN_2	None		AA06-FI-MultiCast	Yes
1060	OOB-Mgmt_1060	None		AA06-FI-MultiCast	Yes
1061	IB-MGMT_1061	None		AA06-FI-MultiCast	Yes
1062	VMMgmt_1062	None		AA06-FI-MultiCast	Yes
3010	iSCSI-A_3010	None		AA06-FI-MultiCast	Yes
3020	iSCSI-B_3020	None		AA06-FI-MultiCast	Yes

Step 13. Select the newly created VLAN policy for both Fabric Interconnects A and B. Click **Next** to go to **Port Configuration.**

Step 14. Enter name of the policy (AA06-FI-PortConfig) and click **Next** twice to go to **Port Roles Page**.

Step 15. In the right pane, under ports, select **port 1** and **2** and click **Configure**.

Step 16. Set **Role** as Server and leave **Auto Negotiation** enabled and click **Save**.

Step 17. In the right pane click **Port Channel** tab and click **Create Port Channel.**

Step 18. Select **Ethernet Uplink Port Channel** for the Role**.** Enter 201 as **Port Channel ID**. Set **Admin speed** as 100Gbps and **FEC** as Cl91.

Step 19. Under Link Control, create a new link control policy with the following options. Once created, select the policy.

Table 4. UDLD policy

Policy Name	Setting Name
AA06-FI-LinkControll	UDLD Admin State: True UDLD mode: Normal

Step 20. Select **Ports 1** and **2** for the Uplink Port Channel and click **Create** to complete the Port Roles policy.

Step 21. Click **Next** to go to **UCS Domain Configuration** page.

The following Management and Network related policies are created and used.

Table 5. Tantp policy

Policy Name	Setting Name
AA06-FI-OCP-NTP	Enable ntp: on Server list: **172.20.10.11,172.20.10.12,172.20.10.13** Timezone: America/New_York

Table 6. Network Connectivity Policy

Policy Name	Setting Name
AA06-FS-OCP-NWPolicy	Proffered IPV4 DNS Server: 10.106.1.21 Alternate IPV4 DNS Server: 10.106.1.22

Table 7. SNMP Policy

Policy Name	Setting Name
AA06-FS-OCP-SNMP	Enable SNMP: On (select **Both v2c and v3**) Snmp Port: 161 System Contact: your snmp admin email address System location: Location details snmp user: Name: snmpadmin Security level: AuthPriv Set Auth and Privacy passwords.

Table 8. QoS Policy

Policy Name	Setting Name
AA06-FS-OCP-SystemQoS	Best Effort: Enable Weight: 5 MTU: 9216

Step 22. When the UCS Domain profile is created with the above mentioned policies, edit the policy and assign it to the Fabric Interconnects.

Intersight will go through the discovery process and discover all the Cisco UCS C and X -Series compute nodes attached to the Fabric Interconnects.

Procedure 2. Server Profile Templates and Policies

In the Cisco Intersight platform, a server profile enables resource management by simplifying policy alignment and server configuration. The server profiles are derived from a server profile template. A Server profile template and its associated policies can be created using the server profile template wizard. After creating the server profile template, you can derive multiple consistent server profiles from the template.

The server profile templates captured in this deployment guide supports Cisco UCS X210c M7 compute nodes with 5th Generation VICs and can be modified to support other Cisco UCS blades and rack mount servers. Server profile templates captured in this deployment guide supports Cisco UCS X210c M7 compute nodes with 5th Generation VICs.

Create Pools

The following pools need to be created before proceeding with server profile template creation.

MAC Pools

The following two MAC pools for the vNICs that will be configured in the templates.

MAC Pool Name	Addresses Ranges
AA06-OCP-IB-MGMT-IPPool-A	From: 00:25:B5:A6:0A:00 Size: 64
AA06-OCP-IB-MGMT-IPPool-B	From: 00:25:B5:A6:0B:00 Size: 64

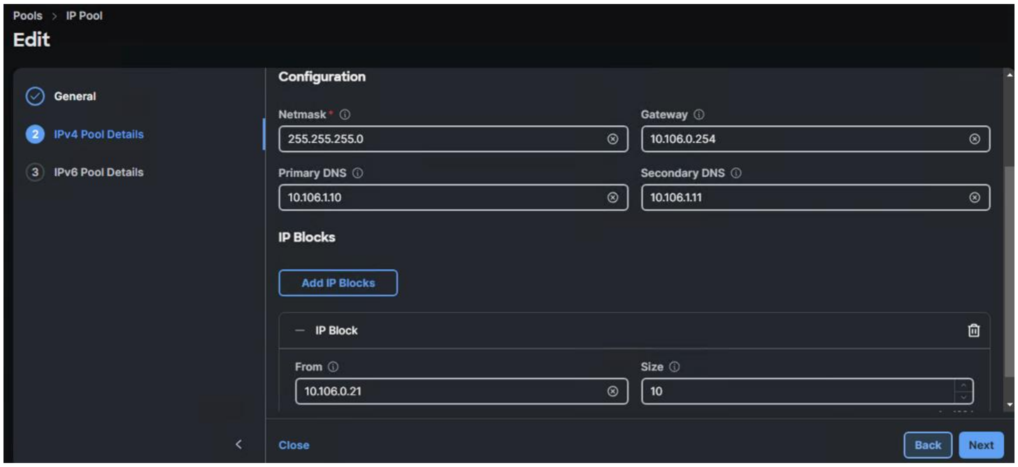
UUID pool

An UUID pool is created with the following settings:

UUID Pool Name	Settings
AA06-OCP-UUIDPool	UUID Prefix: AA060000-0000-0001 From: AA06-000000000001 To: AA06-000000000080 Size: 128
AA06-OCP-IB-MGMT-IPPool-B	From: 00:25:B5:A6:0B:00 Size: 64

Out-Of-Band (OOB) Management IP Pool

An OOB management IP pool (AA06-OCP-OOB-MGMT-IPPool) is created with following settings:



Configuration

Netmask*: 255.255.255.0

Gateway: 10.106.0.254

Primary DNS: 10.106.1.10

Secondary DNS: 10.106.1.11

IP Blocks

IP Block

From: 10.106.0.21

Size: 10

vNIC Templates and vNICs

In this deployment, separate server profile templates are created for Worker and Master Nodes where Worker Nodes have storage network interfaces to support workloads, but Master Nodes do not. The vNIC layout is covered below. While most of the policies are common across various templates, the LAN connectivity policies are unique and will use the information in the tables below.

The following vNIC templates are used for deriving the vNICs for OpenShift worker nodes for host management, VM management and iSCSI storage traffics.

Template Name	AA06-OCP-Mgmt-vNIC Template	AA06-OCP-iSCSIA-vNIC Template	AA06-OCP-iSCSIB-vNIC Template	AA06-VMMgmt-vNIC Template
Purpose	In-Band management of OpenShift hosts	iSCSI traffic through fabric-A (OpenShift - host and VM's In-Guest)	iSCSI traffic through fabric-B (OpenShift - host and VM's In-Guest)	VM management
Mac Pool	AA06-OCP-MACPool-A	AA06-OCP-MACPool-A	AA06-OCP-MACPool-B	AA06-OCP-MACPool-B
Switch ID	A	A	B	B
CDN Source setting	vNIC Name	vNIC Name	vNIC Name	vNIC Name
Fabric Failover setting	Yes	No	No	Yes
Network Group Policy name and Allowed VLANs and Native VLAN	AA06-OCP-BareMetal-NetGrp : Native and Allowed VLAN: 1061	AA06-OCP-iSCSI-A-NetGrp: Native and Allowed VLAN: 3010	AA06-OCP-iSCSIB-NetGrp: Native and Allowed VLAN: 3020	1062,1062
Network Control Policy Name and CDP and LLDP settings	AA06-OCP-CDPLLDP: CDP Enabled LLDP (Tx and Rx) Enable	AA06-OCP-CDPLLDP: CDP Enabled LLDP (Tx and Rx) Enable	AA06-OCP-CDPLLDP: CDP Enabled LLDP (Tx and Rx) Enable	AA06-OCP-CDPLLDP: CDP Enabled LLDP (Tx and Rx) Enable
QoS Policy name and Settings	AA06-OCP-MTU1500-MgmtQoS: Best Effort MTU: 1500 Rate Limit (Mbps): 100000	AA06-OCP-iSCSI-QoS: Best-effort MTU:9000 Rate Limit (Mbps): 100000	AA06-OCP-iSCSI-QoS: Best-effort MTU:9000 Rate Limit (Mbps): 100000	AA06-OCP-MTU1500-MgmtQoS: Best Effort MTU: 1500 Rate Limit (Mbps): 100000
Ethernet Adapter Policy Name and Settings	AA06-OCP-EthAdapter-Linux-v2: Uses system defined Policy: Linux-V2	AA06-OCP-EthAdapter-16RXQs-5G (refer below section)	AA06-OCP-EthAdapter-16RXQs-5G (refer below section)	AA06-OCP-EthAdapter-Linux-v2: Uses system defined Policy: Linux-V2

Note: If you are going to have many VMs added to the OpenShift cluster, then AA06-OCP-EthAdapter-16RXQs-5G adapter policy with MTU set to 1500 can be used for AA06-VMMgmt-vNIC template as well. This will provide more receive and transmit queues to the vNIC that carries the Virtual Machine management traffic.

Ethernet Adapter Policy for iSCSI Traffic

The ethernet adapter policy is used to set the interrupts, send and receive queues, and queue ring size. The values are set according to the best-practices guidance for the operating system in use. Cisco Intersight provides a default Linux Ethernet Adapter policy for typical Linux deployments.

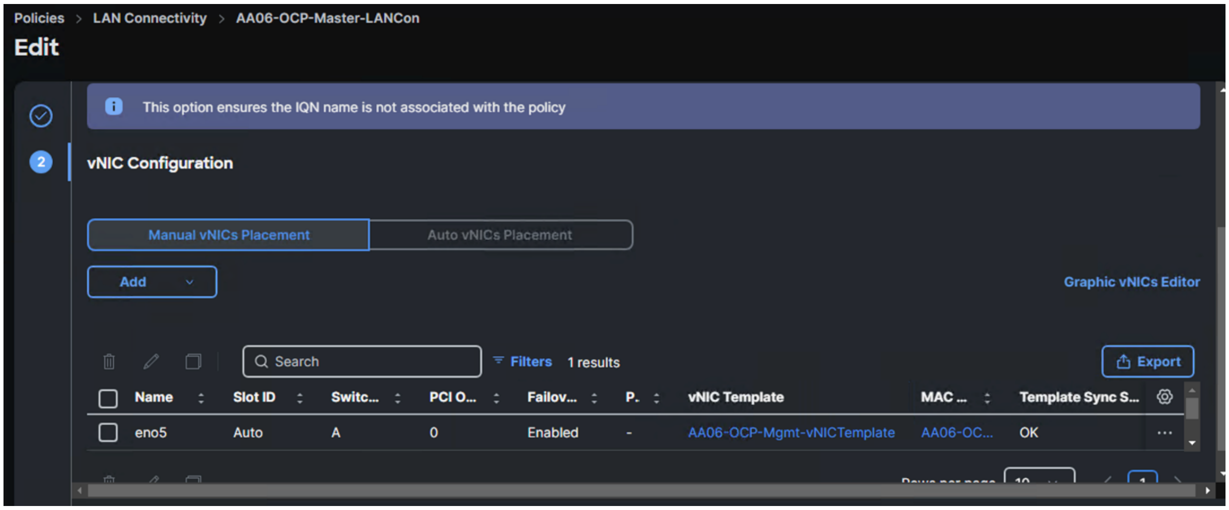
You can optionally configure a tweaked ethernet adapter policy for additional hardware receive queues handled by multiple CPUs in scenarios where there is a lot of traffic and multiple flows. In this deployment, a modified ethernet adapter policy, AA06-EthAdapter-16RXQs-5G, is created and attached to storage vNICs. Non-storage vNICs will use the default Linux-v2 Ethernet Adapter policy. Table 9 lists the settings that are changed from defaults in the Adapter policy used for the iSCSI traffic. The remaining settings are left at defaults.

Table 9. Settings and Values

Setting Name	Value
Name of the Policy	AA06-OCP-EthAdapter-16RXQs-5G
Interrupt Settings	Interrupts: 19, Interrupt Mode: MSX ,Interrupt Timer: 125
Receive	Receive Queue Count: 16, Receive Ring Size: 16384
Transmit	Transmit Queue Count: 1, Transmit Ring Size: 16384
Completion	Completion Queue Count: 17, Completion Ring Size: 1

Using the templates listed in Table 9, separate LAN connectivity policies are created for control and worker nodes.

Control nodes are configured with one vNIC which is derived from the AA06-OCP-Mgmt-vNIC template. Following screenshot shows the LAN connectivity policy (AA06-OCP-master-LANCon) created with one vNIC for control node.



Policies > LAN Connectivity > AA06-OCP-Master-LANCon

Edit

This option ensures the IQN name is not associated with the policy

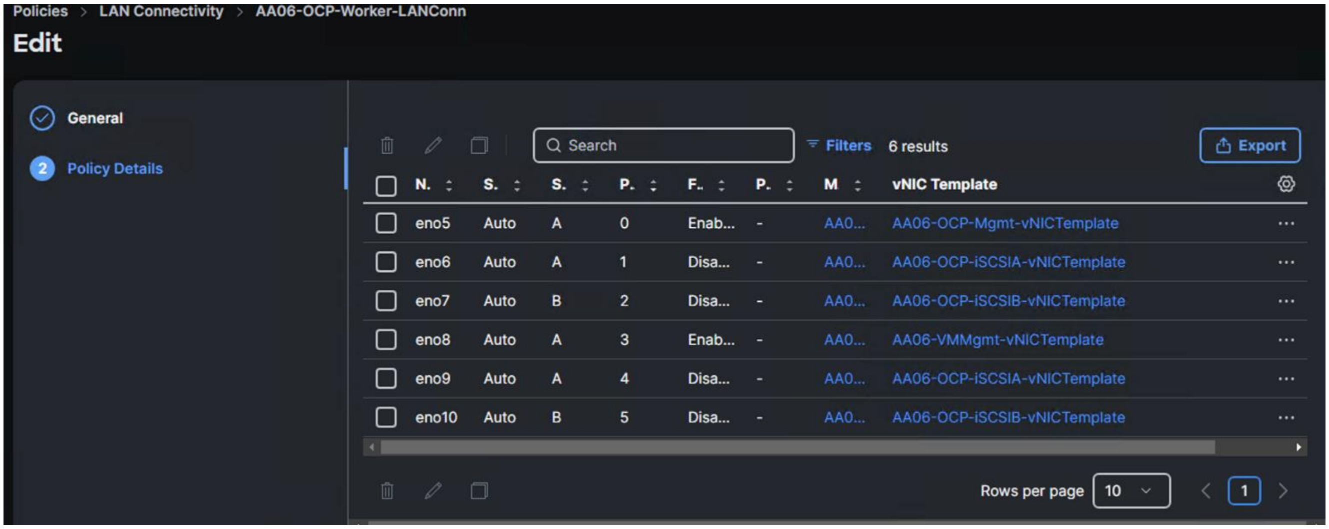
2 vNIC Configuration

Manual vNICs Placement | Auto vNICs Placement

1 results

Name	Slot ID	Switc...	PCI O...	Failov...	P.	vNIC Template	MAC ...	Template Sync S...
eno5	Auto	A	0	Enabled	-	AA06-OCP-Mgmt-vNICTemplate	AA06-OC...	OK

Worker nodes are configured with six vNICs which are derived from the templates discussed above. Following screenshot shows the LAN connectivity policy (AA06-OCP-Worker-LANConn) created with six vNICs for worker nodes.



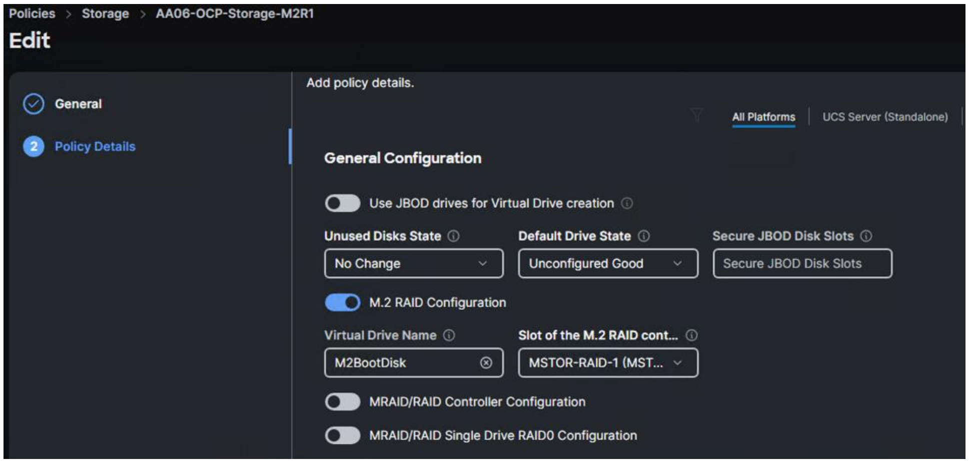
N.	S.	S.	P.	F..	P.	M	vNIC Template
eno5	Auto	A	0	Enab...	-	AA0...	AA06-OCP-Mgmt-vNICTemplate
eno6	Auto	A	1	Disa...	-	AA0...	AA06-OCP-iSCSIA-vNICTemplate
eno7	Auto	B	2	Disa...	-	AA0...	AA06-OCP-iSCSIB-vNICTemplate
eno8	Auto	A	3	Enab...	-	AA0...	AA06-VMMgmt-vNICTemplate
eno9	Auto	A	4	Disa...	-	AA0...	AA06-OCP-iSCSIA-vNICTemplate
eno10	Auto	B	5	Disa...	-	AA0...	AA06-OCP-iSCSIB-vNICTemplate

Note: vNICs eno5, eno6 and eno7 will be used to carry the management and iSCSI-A and iSCSI-B traffics of worker nodes while eno8,eno9, and eno10 will carry the virtual machine's management, iSCSI-A and iSCSI-B traffics.

Note: vNICs eno9 and eno10 are derived using the same vNIC templates as that of eno7 and eno8.

Storage Policy

For this solution, Cisco UCS X210c nodes are configured to boot from local M.2 SSD disks. Two M.2 disks are used and configured with RAID-1 configuration. Boot from SAN option will be supported in the next releases. The following screenshot shows the storage policy (AA06-OCP-Storage-M2R1), and the settings used for configuring the M.2 disks in RAID-1 mode.



General Configuration

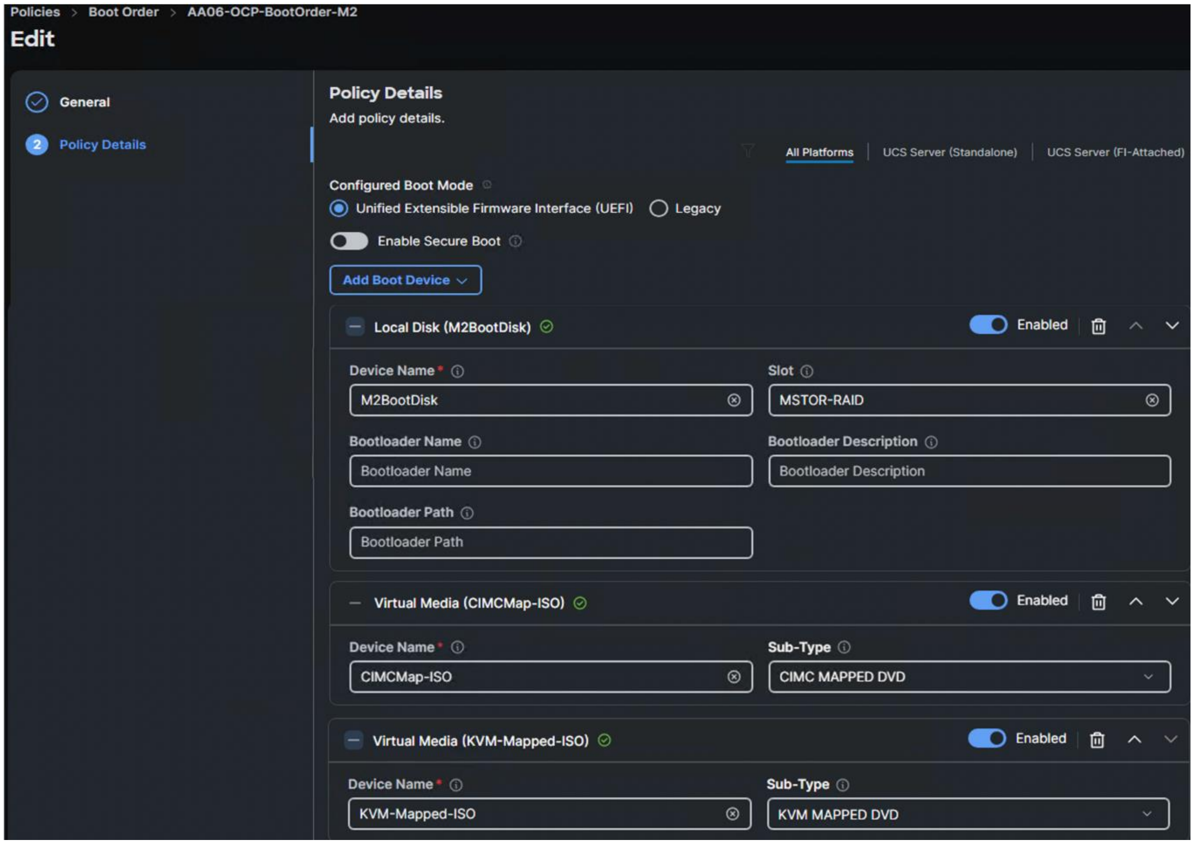
- Use JBOD drives for Virtual Drive creation: off
- Unused Disks State: No Change
- Default Drive State: Unconfigured Good
- Secure JBOD Disk Slots: Secure JBOD Disk Slots
- M.2 RAID Configuration: on
- Virtual Drive Name: M2BootDisk
- Slot of the M.2 RAID cont...: MSTOR-RAID-1 (MST...
- MRAID/RAID Controller Configuration: off
- MRAID/RAID Single Drive RAID0 Configuration: off

Compute Configuration Policies

Boot Policy

To facilitate the automatic boot from the Red Hat CoreOS Discovery ISO image, CIMC Mapped DVD boot option is used. The following boot policy is used for both controller and workers nodes.

Note: It is critical to not enable UEFI Secure Boot. Secure Boot needs to be disabled for the proper functionality of Portworx Enterprise and the NVIDIA GPU Operator GPU driver initialization.



Policy Details

Add policy details.

All Platforms | UCS Server (Standalone) | UCS Server (FI-Attached)

Configured Boot Mode

● Unified Extensible Firmware Interface (UEFI) ○ Legacy

Enable Secure Boot

Local Disk (M2BootDisk) — Enabled

Device Name*: M2BootDisk

Slot: MSTOR-RAID

Bootloader Name: Bootloader Name

Bootloader Description: Bootloader Description

Bootloader Path: Bootloader Path

Virtual Media (CIMCMap-ISO) — Enabled

Device Name*: CIMCMap-ISO

Sub-Type: CIMC MAPPED DVD

Virtual Media (KVM-Mapped-ISO) — Enabled

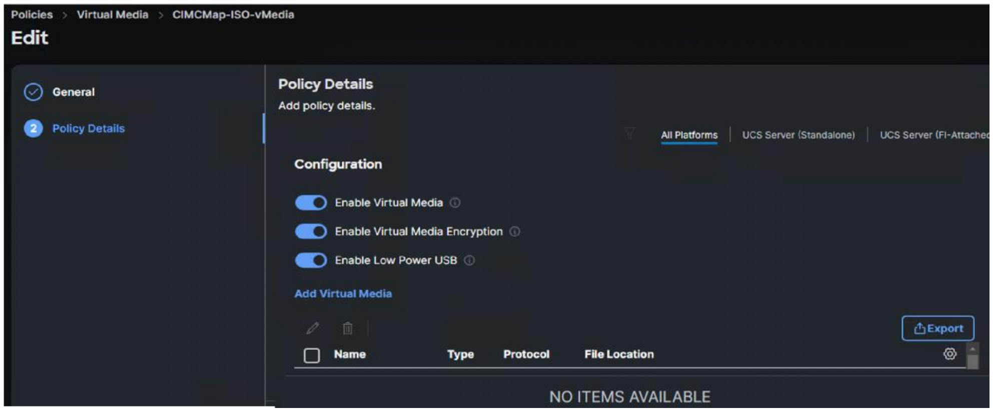
Device Name*: KVM-Mapped-ISO

Sub-Type: KVM MAPPED DVD

Local Disk boot option being at the top ensures that the nodes always boot from the M.2 disks once after CoreOS installed. The CIMC Mapped DVC option at the second is used to install the CoreOS using Discovery ISO which is mapped using a Virtual Media policy (CIMCMap-ISO). KVM Mapped DVD will be used if you want to manually mount any ISO to the KVM session of the server and install the OS. This option will be used when installing CoreOS during the OpenShift cluster expansion by adding additional worker node.

Virtual Media (vMedia) Policy

Virtual Media policy is used to mount the Red Hat CoreOS Discovery ISO to the server using CIMC Mapped DVD policy as previously explained. A file share service is required configured and must be accessed by OOB-Mgmt network. In this solution, the HTTP file share service is used to share the Discovery ISO over the network.



Policy Details

Add policy details.

All Platforms | UCS Server (Standalone) | UCS Server (FI-Attached

Configuration

Enable Virtual Media — enabled

Enable Virtual Media Encryption — enabled

Enable Low Power USB — enabled

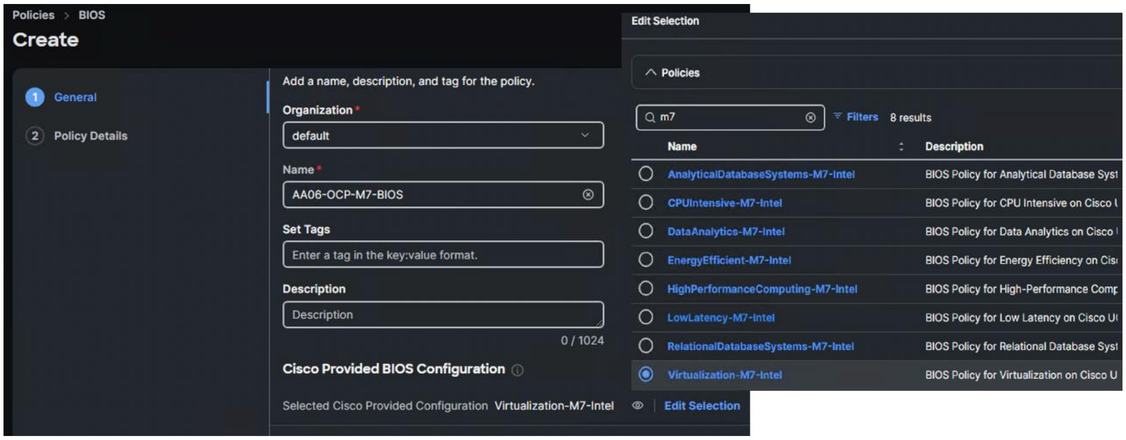
□ Name	Type	Protocol	File Location

NO ITEMS AVAILABLE

Note: Do not Add Virtual Media at this time, but the policy can be modified later and used to map an OpenShift Discovery ISO to a CIMC Mapped DVD policy.

Procedure 1. Bios Policy

For the **OpenShift** containerized and Virtualized solution, which is based on Intel M7 platform, system defined "**virtualization-M7-Intel**" policy is used in this solution.



Add a name, description, and tag for the policy.

Organization * default

Name * AA06-OCP-M7-BIOS

Set Tags: Enter a tag in the key:value format.

Description: Description 0 / 1024

Cisco Provided BIOS Configuration

Selected Cisco Provided Configuration Virtualization-M7-Intel

	Name	Description
○	AnalyticalDatabaseSystems-M7-Intel	BIOS Policy for Analytical Database Syst
○	CPUIntensive-M7-Intel	BIOS Policy for CPU Intensive on Cisco U
○	DataAnalytics-M7-Intel	BIOS Policy for Data Analytics on Cisco
○	EnergyEfficient-M7-Intel	BIOS Policy for Energy Efficiency on Cis
○	HighPerformanceComputing-M7-Intel	BIOS Policy for High-Performance Comp
○	LowLatency-M7-Intel	BIOS Policy for Low Latency on Cisco U
○	RelationalDatabaseSystems-M7-Intel	BIOS Policy for Relational Database Syst
●	Virtualization-M7-Intel	BIOS Policy for Virtualization on Cisco U

Step 1. Create BIOS policy and select pre-defined policy as shown above and click **Next**.

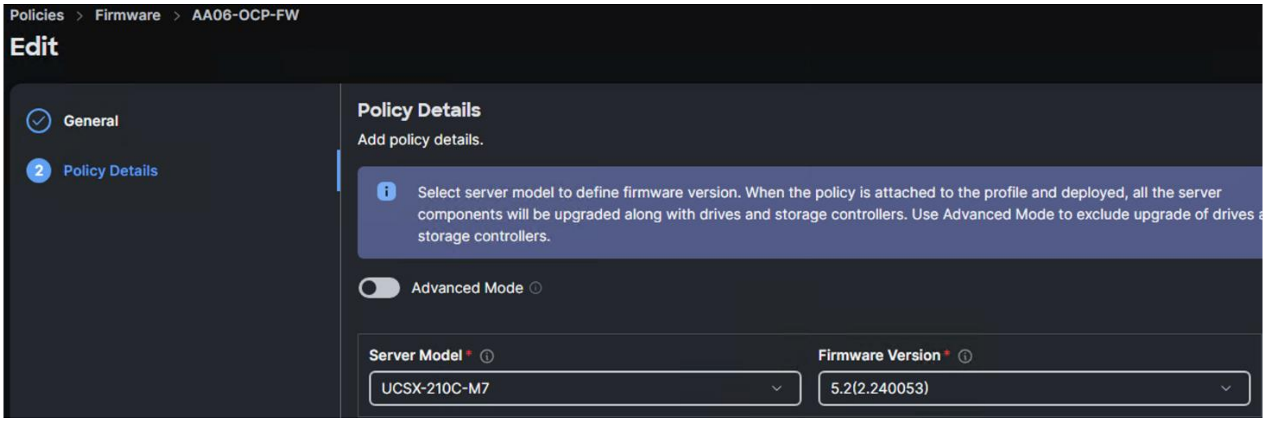
Step 2. Expand the Server Management and set **Consistent Device Name** (CDN) to enabled for Consistent Device Naming within the Operating System.

Step 3. The remaining bios tokens and their values mentioned here are based on the best practices guide from M7 platform. For more details, go to: Performance tuning best practices Guide for Cisco UCS M7 platform

Step 4. Click **Create** to complete the BIOS policy.

Procedure 2. Firmware Policy (optional)

Step 1. Create a Firmware policy (AA06-OCP-FW) and under the Policy Detail tab, set the Server Model as **UCSX-210C-M7** and set Firmware Version to the latest version. The following screenshot shows the firmware policy used in this solution.



Policy Details

Add policy details.

Select server model to define firmware version. When the policy is attached to the profile and deployed, all the server components will be upgraded along with drives and storage controllers. Use Advanced Mode to exclude upgrade of drives a storage controllers.

Advanced Mode

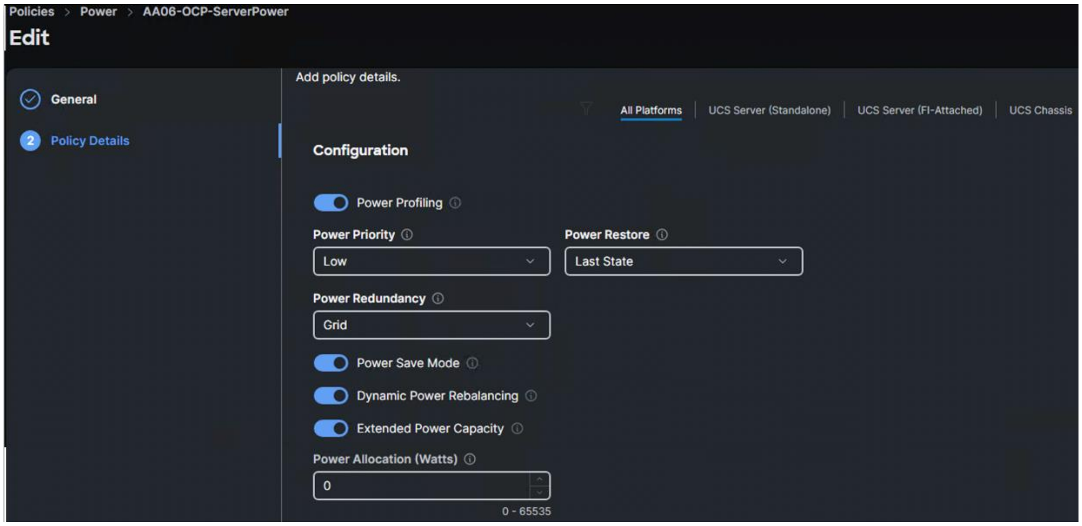
Server Model * UCSX-210C-M7

Firmware Version * 5.2(2.240053)

Procedure 3. Power Policy

Create a Policy with the following options:

Step 1. Select **All-Platform** (unless you want to create a dedicated power policy for FI-Attached servers) and select the following option and leave the rest of the settings at default. When you apply this policy to the server profile template, the system will take appropriate settings and apply to the server.


Policies > Power > AA06-OCP-ServerPower

Edit

General

2 Policy Details

Add policy details.

All Platforms | UCS Server (Standalone) | UCS Server (FI-Attached) | UCS Chassis

Configuration

Power Profiling: On

Power Priority: Low

Power Restore: Last State

Power Redundancy: Grid

Power Save Mode: On

Dynamic Power Rebalancing: On

Extended Power Capacity: On

Power Allocation (Watts): 0 (0 - 65535)

Management Configuration Policies

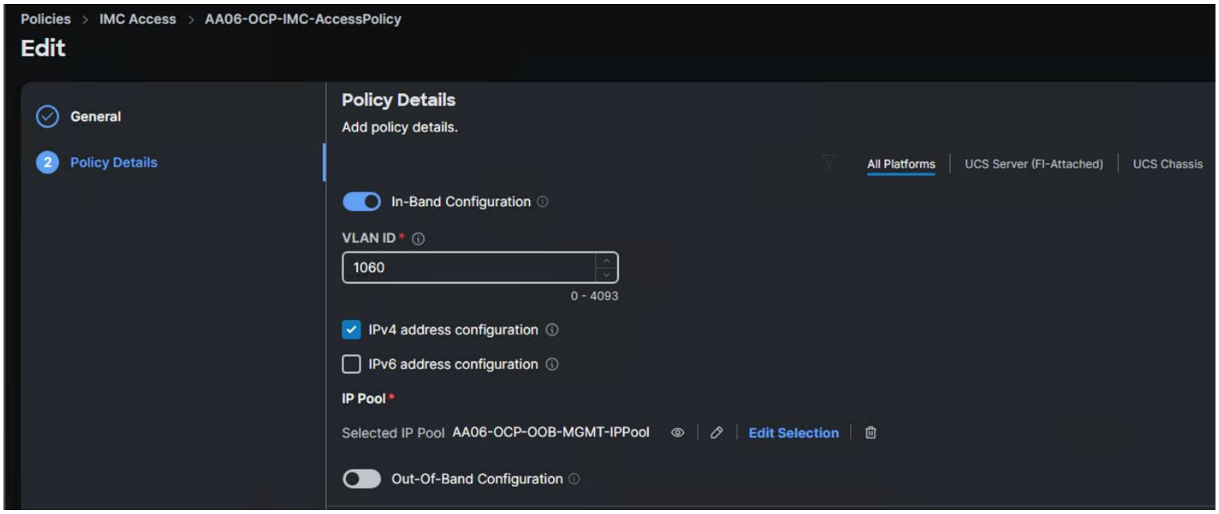
The following policies will be added to the management configuration:

- IMC Access to define the pool of IP addresses for compute node KVM access
- IPMI Over LAN to allow the servers to be managed by IPMI or redfish through the BMC or CIMC
- Local User to provide local administrator to access KVM
- Virtual KVM to allow the Tunneled KVM

Cisco IMC Access Policy

Create a CIMC Access Policy with settings as shown in the screenshot below.

Note: Since certain features are not yet enabled for Out-of-Band Configuration (accessed using the Fabric Interconnect mgmt0 ports), you need to access the OOB-MGMT VLAN (1060) through the Fabric Interconnect Uplinks and mapping it as the In-Band Configuration VLAN.


Policies > IMC Access > AA06-OCP-IMC-AccessPolicy

Edit

General

2 Policy Details

Policy Details

Add policy details.

All Platforms | UCS Server (FI-Attached) | UCS Chassis

In-Band Configuration: On

VLAN ID *: 1060 (0 - 4093)

☑ IPv4 address configuration

☐ IPv6 address configuration

IP Pool *

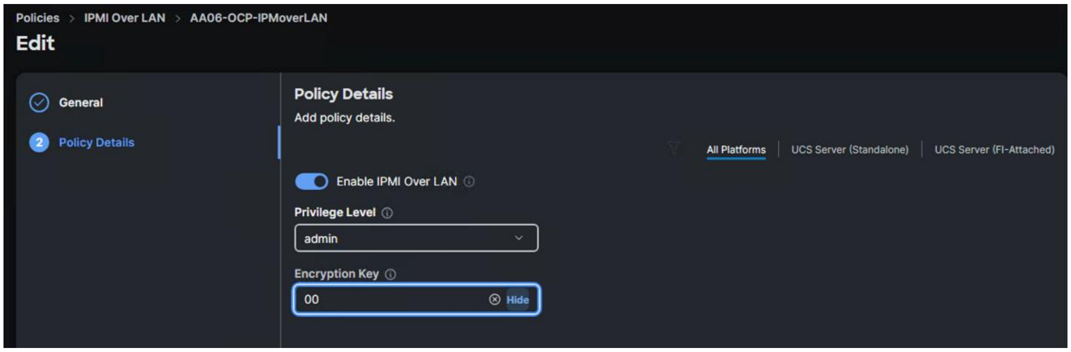
Selected IP Pool AA06-OCP-OOB-MGMT-IPPool | Edit Selection

Out-Of-Band Configuration: Off

IPMI over LAN and Local User Policies

The IPMI Over LAN Policy can be used to allow both IPMI and Redfish connectivity to Cisco UCS Servers. Red Hat OpenShift platform uses these two policies to power manage (power off, restart, and so on) the baremetal servers.

Create IPMI over LAN policy (**AA06-IPMIOvelLan**) as shown below.



Policies > IPMI Over LAN > AA06-OCP-IPMoverLAN

Edit

1. General
2. Policy Details

Policy Details

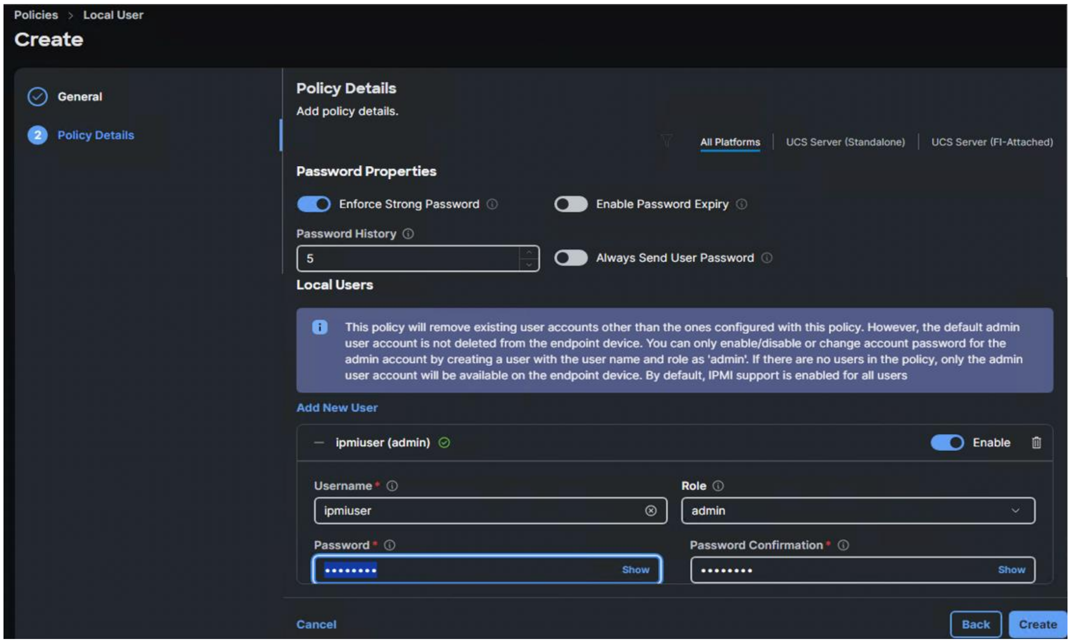
Add policy details.

All Platforms | UCS Server (Standalone) | UCS Server (FI-Attached)

Enable IPMI Over LAN: on

Privilege Level: admin

Encryption Key: 00



Policies > Local User

Create

1. General
2. Policy Details

Policy Details

Add policy details.

All Platforms | UCS Server (Standalone) | UCS Server (FI-Attached)

Password Properties

Enforce Strong Password: on

Enable Password Expiry: off

Password History: 5

Always Send User Password: off

Local Users

This policy will remove existing user accounts other than the ones configured with this policy. However, the default admin user account is not deleted from the endpoint device. You can only enable/disable or change account password for the admin account by creating a user with the user name and role as 'admin'. If there are no users in the policy, only the admin user account will be available on the endpoint device. By default, IPMI support is enabled for all users

Add New User

ipmiuser (admin) — Enable: on

Username: ipmiuser

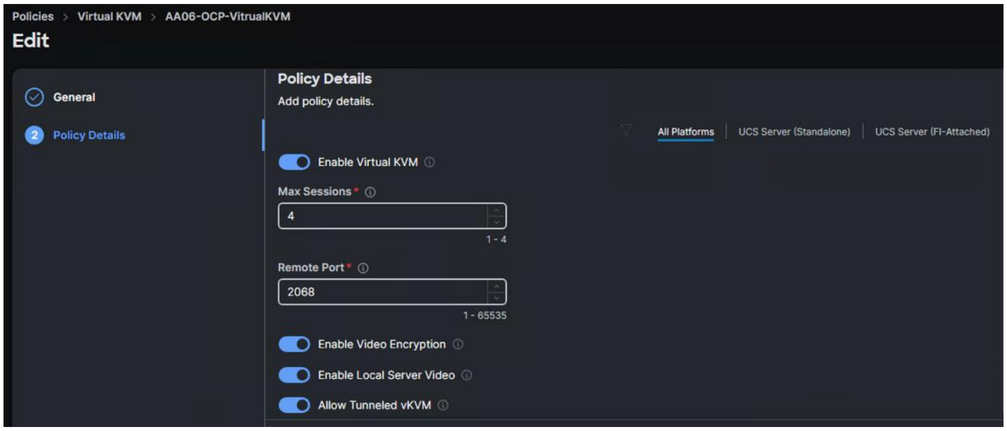
Role: admin

Password: ••••••••

Password Confirmation: ••••••••

Virtual KVM Policy

The following screenshot shows the virtual KVM policy (**AA06-OCP-VirtualKVM**) used in the solution.



Policies > Virtual KVM > AA06-OCP-VitrualKVM

Edit

1. General
2. Policy Details

Policy Details

Add policy details.

All Platforms | UCS Server (Standalone) | UCS Server (FI-Attached)

Enable Virtual KVM: on

Max Sessions: 4 (1 - 4)

Remote Port: 2068 (1 - 65535)

Enable Video Encryption: on

Enable Local Server Video: on

Allow Tunneled vKVM: on

Create Server Profile Templates

When you have the required pools, polices, vNIC templates created, Server profile templates can be created. Two separate Server Profile Templates are used for control and workers node.

The following table provides list of polices and pools used for creating Server Profile template (AA06-OCP-Master-M.2) for Control nodes:

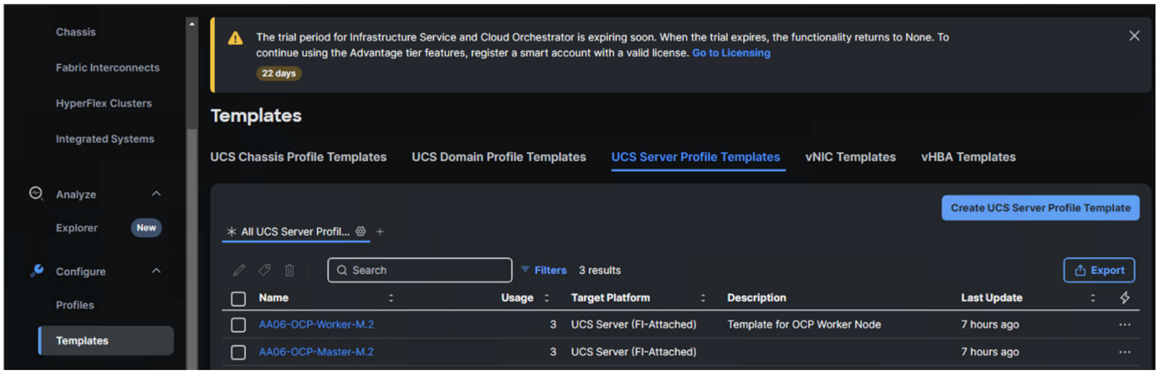
Page Name	Setting
General	Name: AA06-OCP-Master-M.2
Compute Configuration	UUID: AA06-OCP-UUIDPool BIOS: AA06-OCP-M7-BIOS Boot Order: AA06-OCP-BootOrder-M2 Firmware: AA06-OCP-FW Power: AA06-OCP-ServerPower Virtual Media: CIMCMap-ISO-vMedia
Management Configuration:	IMC Access: AA06-OCP-IMC-AccessPolicy IPMI Over LAN: AA06-OCP-IPMoverLAN Local User: AA06-OCP-IMCLocalUser Virtual KVM: AA06-OCP-VitrualKVM
Storage Configuration	Storage: AA06-OCP-Storage-M2R1
Network Configuration	LAN Connectivity: AA06-OCP-Master-LANCon

The following table provides list of polices and pools used for creating Server Profile template (AA06-OCP-Worker-M.2) for worker nodes:

Page Name	Setting
General	Name: AA06-OCP-Worker-M.2
Compute Configuration	UUID: AA06-OCP-UUIDPool BIOS: AA06-OCP-M7-BIOS Boot Order: AA06-OCP-BootOrder-M2 Firmware: AA06-OCP-FW Power: AA06-OCP-ServerPower Virtual Media: CIMCMap-ISO-vMedia
Management Configuration:	IMC Access: AA06-OCP-IMC-AccessPolicy IPMI Over LAN: AA06-OCP-IPMoverLAN Local User: AA06-OCP-IMCLocalUser

Page Name	Setting
	Virtual KVM: AA06-OCP-VitrualKVM
Storage Configuration	Storage: AA06-OCP-Storage-M2R1
Network Configuration	LAN Connectivity: AA06-OCP-Worker-LANConn

The following screenshot show the two server profile templates created for control and worker nodes:



The trial period for Infrastructure Service and Cloud Orchestrator is expiring soon. When the trial expires, the functionality returns to None. To continue using the Advantage tier features, register a smart account with a valid license. Go to Licensing

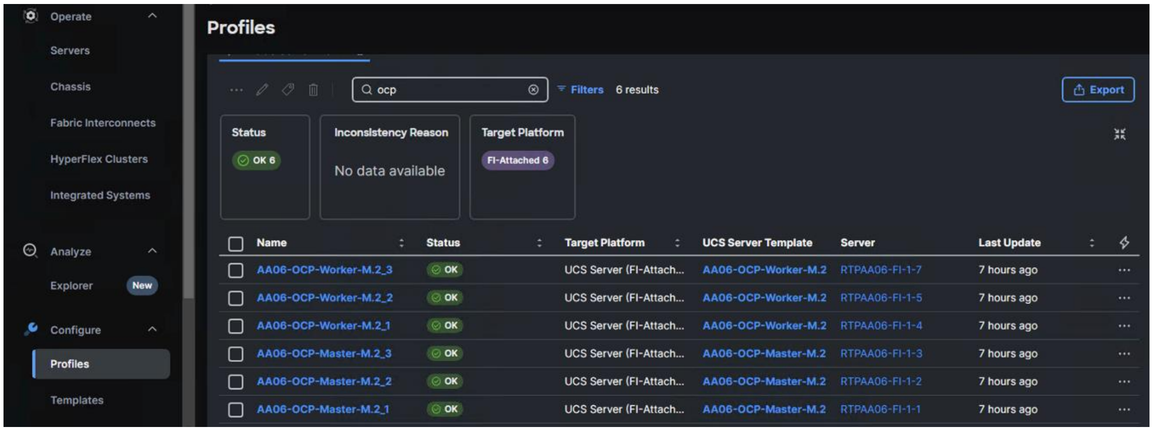
22 days

3 results

Name	Usage	Target Platform	Description	Last Update
AA06-OCP-Worker-M.2	3	UCS Server (FI-Attached)	Template for OCP Worker Node	7 hours ago
AA06-OCP-Master-M.2	3	UCS Server (FI-Attached)		7 hours ago

Create Server Profiles

Once Server Profile Templates are created, the server profiles can be derived from the template. The following screenshot shows total of six profiles are derived (three for control nodes and three for worker nodes).



6 results

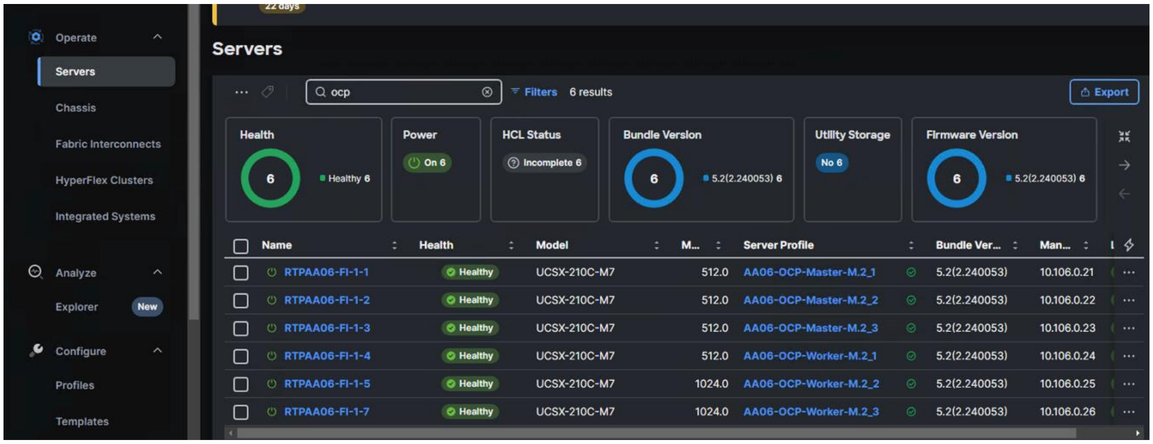
Status: OK 6

Inconsistency Reason: No data available

Target Platform: FI-Attached 6

Name	Status	Target Platform	UCS Server Template	Server	Last Update
AA06-OCP-Worker-M.2_3	OK	UCS Server (FI-Attach...	AA06-OCP-Worker-M.2	RTPAA06-FI-1-7	7 hours ago
AA06-OCP-Worker-M.2_2	OK	UCS Server (FI-Attach...	AA06-OCP-Worker-M.2	RTPAA06-FI-1-5	7 hours ago
AA06-OCP-Worker-M.2_1	OK	UCS Server (FI-Attach...	AA06-OCP-Worker-M.2	RTPAA06-FI-1-4	7 hours ago
AA06-OCP-Master-M.2_3	OK	UCS Server (FI-Attach...	AA06-OCP-Master-M.2	RTPAA06-FI-1-3	7 hours ago
AA06-OCP-Master-M.2_2	OK	UCS Server (FI-Attach...	AA06-OCP-Master-M.2	RTPAA06-FI-1-2	7 hours ago
AA06-OCP-Master-M.2_1	OK	UCS Server (FI-Attach...	AA06-OCP-Master-M.2	RTPAA06-FI-1-1	7 hours ago

When the Server profiles are created, associate these server profiles to the control and workers nodes as shown below.



Name	Health	Model	M...	Server Profile		Bundle Ver...	Man...
RTPAA06-FI-1-1	Healthy	UCSX-210C-M7	512.0	AA06-OCP-Master-M.2_1	✓	5.2(2.240053)	10.106.0.21
RTPAA06-FI-1-2	Healthy	UCSX-210C-M7	512.0	AA06-OCP-Master-M.2_2	✓	5.2(2.240053)	10.106.0.22
RTPAA06-FI-1-3	Healthy	UCSX-210C-M7	512.0	AA06-OCP-Master-M.2_3	✓	5.2(2.240053)	10.106.0.23
RTPAA06-FI-1-4	Healthy	UCSX-210C-M7	512.0	AA06-OCP-Worker-M.2_1	✓	5.2(2.240053)	10.106.0.24
RTPAA06-FI-1-5	Healthy	UCSX-210C-M7	1024.0	AA06-OCP-Worker-M.2_2	✓	5.2(2.240053)	10.106.0.25
RTPAA06-FI-1-7	Healthy	UCSX-210C-M7	1024.0	AA06-OCP-Worker-M.2_3	✓	5.2(2.240053)	10.106.0.26

Now the Cisco UCS X210c M7 blades are ready and OpenShift can be installed on these machines.

Pure Storage FlashArray Configuration

In this solution, Pure Storage FlashArray//XL170 is used as the storage provider for all the application pods and virtual machines provisioned on the OpenShift cluster using Portworx Enterprise. The Pure Storage FlashArray//XL170 array will be used as Cloud Storage Provider for Portworx which allows us to store data on-premises with FlashArray while benefiting from Portworx Enterprise cloud drive features.

This section describes high-level steps to configure Pure Storage FlashArray//X170 network interfaces required for storage connectivity over iSCSI. For this solution, Pure Storage FlashArray was loaded with Purity//FA Version 6.6.10.

Note: This document is not intended to explain every day-0 initial configuration steps to bring the array up and running. For detailed day-0 configuration steps, see: https://www.cisco.com/c/en/us/td/docs/unified_computing/ucs/UCS_CVDs/flashstack_ucs_xseries_e2e_5gen.html#FlashArrayConfiguration

The compute nodes are redundantly connected to the storage controllers through 4 x 100Gb connections (2 x 100Gb per storage controller module) from the redundant Cisco Nexus switches.

The Pure Storage FlashArray network settings were configured with three subnets across three VLANs. Storage Interfaces CT0.Eth0 and CT1.Eth0 were configured to access management for the storage on VLAN 1063. Storage Interfaces (CT0.Eth18, CT0.Eth19, CT1.Eth18, and CT1.Eth19) were configured to run iSCSI Storage network traffic on the VLAN 3010 and VLAN 3020.

The following tables provides the IP addressing configured on the interfaces used for storage access.

Table 10. iSCSI A Pure Storage FlashArray//XL170 Interface Configuration Settings

FlashArray Controller	iSCSI Port	IP Address	Subnet
FlashArray//X170 Controller 0	CT0.ETH18	192.168.51.4	255.255.255.0
FlashArray//X170 Controller 1	CT1.ETH18	192.168.51.5	255.255.255.0

Table 11. iSCSI B Pure Storage FlashArray//XL170 Interface Configuration Settings

FlashArray Controller	iSCSI Port	IP Address	Subnet
FlashArray//X170 Controller 0	CT0.ETH19	192.168.52.4	255.255.255.0
FlashArray//X170 Controller 1	CT1.ETH19	192.168.52.5	255.255.255.0

Procedure 1. Configure iSCSI Interfaces

Step 1. Log into Pure FlashArray//XL170 using its management IP addresses.

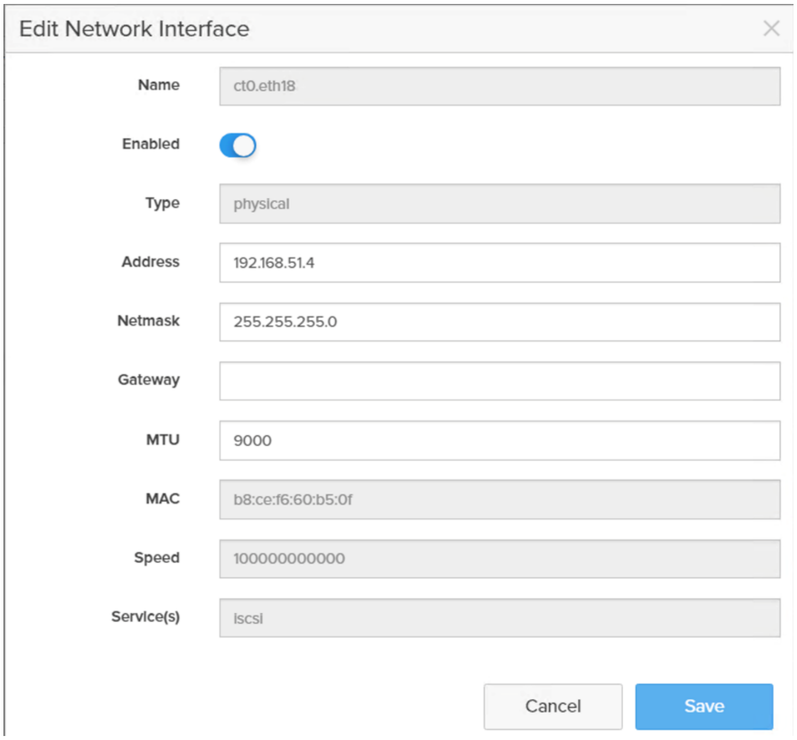
Step 2. Click **Settings > Network > Connectors > Ethernet**.

Step 3. Click **Edit** for Interface CT0.eth18.

Step 4. Click **Enable** and add the IP information from Table 10 and Table 11 and set the MTU to 9000.

Step 5. Click **Save**.

Step 6. Repeat steps 1 through 5 to configure the remaining interfaces CT0.eth19, CT1.eth18 and CT1.eth19.



Edit Network Interface

Name: ct0.eth18

Enabled: ●

Type: physical

Address: 192.168.51.4

Netmask: 255.255.255.0

Gateway:

MTU: 9000

MAC: b8:ce:f6:60:b5:0f

Speed: 100000000000

Service(s): iscsi

Procedure 2. Claim Pure Storage FlashArray//XL170 into Intersight

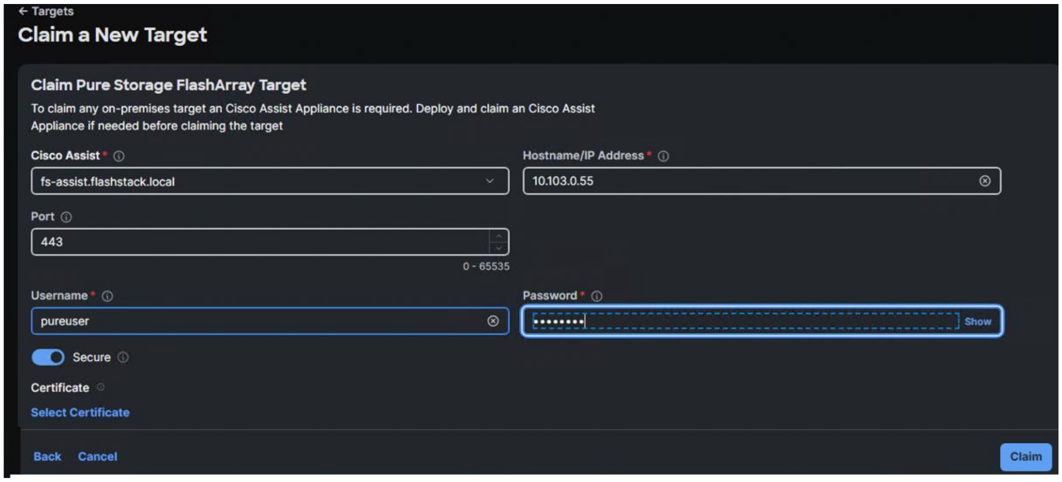
To claim the Pure Storage FlashArray//XL170 into Cisco Intersight using Intersight Assist, follow these steps.

Note: This procedure assumes that Cisco Intersight Is already hosted outside the OpenShift cluster and claimed into the Intersight.com.

Step 1. Log into Cisco Intersight with your login credentials. From the drop-down list located on the left top, select **System**.

Step 2. Under Admin, select **Target** and click **Claim a New Target**. Under Categories, select **Storage**, click **Pure Storage FlashArray** and then click **Start**.

Step 3. Select the Cisco Assist name which is already deployed and configured. Provide the Pure Storage FlashArray management IP address, username and password details and click **Claim**.



← Targets

Claim a New Target

Claim Pure Storage FlashArray Target

To claim any on-premises target an Cisco Assist Appliance is required. Deploy and claim an Cisco Assist Appliance if needed before claiming the target

Cisco Assist *: fs-assist.flashstack.local

Hostname/IP Address *: 10.103.0.55

Port: 443 (0 - 65535)

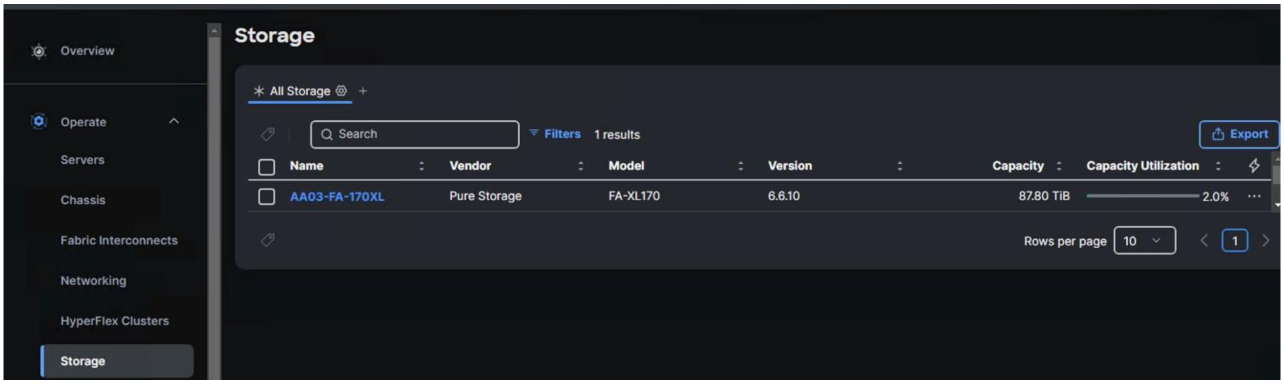
Username *: pureuser

Password *: ••••••••

Secure: ●

Certificate: Select Certificate

Step 4. When the storage is successfully claimed, from the top left drop-down list, select **Infrastructure Services**. Under Operate, click **Storage**. On the right you will find the newly claimed Pure Storage FlashArray and browse through it for viewing the inventory details.



Storage

Name	Vendor	Model	Version	Capacity	Capacity Utilization
AA03-FA-170XL	Pure Storage	FA-XL170	6.6.10	87.80 TiB	2.0%

OpenShift Container Platform Installation and Configuration

This chapter contains the following:

- OpenShift Container Platform – Installation Requirements
- Prerequisites
- Network Requirements

OpenShift 4.16 is deployed on the Cisco UCS infrastructure as M.2 booted bare metal servers. The Cisco UCS X210C M7 servers need to be equipped with an M.2 controller (SATA or NVMe) card and two identical M.2 drives. Three master nodes and three worker nodes are deployed in the validation environment and additional worker nodes can easily be added to increase the scalability of the solution. This document will guide you through the process of using the Assisted Installer to deploy OpenShift 4.16.

OpenShift Container Platform – Installation Requirements

The Red Hat OpenShift Assisted Installer provides support for installing OpenShift Container Platform on bare metal nodes. This guide provides a methodology to achieving a successful installation using the Assisted Installer.

Prerequisites

The FlashStack for OpenShift utilizes the Assisted Installer for OpenShift installation. Therefore, when provisioning and managing the FlashStack infrastructure, you must provide all the supporting cluster infrastructure and resources, including an installer VM or host, networking, storage, and individual cluster machines.

The following supporting cluster resources are required for the Assisted Installer installation:

- The control plane and compute machines that make up the cluster
- Cluster networking
- Storage for the cluster infrastructure and applications
- The Installer VM or Host

Network Requirements

The following infrastructure services need to be deployed to support the OpenShift cluster, during the validation of this solution we have provided VMs on your hypervisor of choice to run the required services. You can use existing DNS and DHCP services available in the data center.

There are various infrastructure services prerequisites for deploying OpenShift 4.16. These prerequisites are as follows:

- DNS and DHCP services – these services were configured on Microsoft Windows Server VMs in this validation
- NTP Distribution was done with Nexus switches
- Specific DNS entries for deploying OpenShift – added to the DNS server
- A Linux VM for initial automated installation and cluster management – a Rocky Linux / RHEL VM with appropriate packages

NTP

Each OpenShift Container Platform node in the cluster must have access to at least two NTP servers.

NICs

NICs configured on the Cisco UCS servers based on the design previously discussed.

DNS

Clients access the OpenShift Container Platform cluster nodes over the bare metal network. Configure a subdomain or subzone where the canonical name extension is the cluster name.

The following domain and OpenShift cluster names are used in this deployment guide:

- Base Domain: flashstack.local
- OpenShift Cluster Name: fs-ocp1

The DNS domain name for the OpenShift cluster should be the cluster name followed by the base domain, for example fs-ocp1. flashstack.local.

Table 12 lists the information for fully qualified domain names used during validation. The API and Nameserver addresses begin with canonical name extensions. The hostnames of the control plane and worker nodes are exemplary, so you can use any host naming convention you prefer.

Table 12. DNS FQDN Names Used

Usage	Hostname	IP Address
API	api.fs-ocp1.flashstack.local	10.106.1.31
Ingress LB (apps)	*.apps.fs-ocp1.flashstack.local	10.106.1.32
master1	master1.fs-ocp1.flashstack.local	10.106.1.33
master2	master2.fs-ocp1.flashstack.local	10.106.1.34
master3	master3.fs-ocp1.flashstack.local	10.106.1.35
worker1	worker1.fs-ocp1.flashstack.local	10.106.1.36
worker2	worker2.fs-ocp1.flashstack.local	10.106.1.37
worker3	worker3.fs-ocp1.flashstack.local	10.106.1.38

DHCP

For the bare metal network, a network administrator must reserve several IP addresses, including:

- One IP address for the API endpoint
- One IP address for the wildcard Ingress endpoint
- One IP address for each master node (DHCP server assigns to the node)
- One IP address for each worker node (DHCP server assigns to the node)

Note: Get the MAC addresses of the bare metal Interfaces from the UCS Server Profile for each node to be used in the DHCP configuration to assign reserved IP addresses (reservations) to the nodes. The KVM IP address also needs to be gathered for the master and worker nodes from the server profiles.

Procedure 1. Gather MAC Addresses of Node Bare Metal Interfaces

Step 1. Log into **Cisco Intersight**.

Step 2. Select **Infrastructure Service > Profiles > UCS Server Profile** (for example, AA06-OCP-Worker-M.2_3).

Step 3. In the center pane, select **Inventory > Network Adapters > Network Adapter** (for example, UCSX-ML-V5D200G).

Step 4. In the center pane, select **Interfaces**.

Step 5. Record the MAC address for NIC Interface eno5.

Step 6. Select the **General** tab and select **Identifiers** in the center pane.

Step 7. Record the Management IP assigned out of the AA06-OCP-OOB-MGMT-IPPool.

Table 13 lists the IP addresses used for the OpenShift cluster including bare metal network IPs and UCS KVM Management IPs for IMPI or Redfish access.

Table 13. Host BMC Information

Hostname	Management IP Address	UCS KVM Mgmt. IP Address	BareMetal MAC Address (eno5)
master1.fs-ocp1.flashstack.local	10.106.1.33	10.106.0.21	00-25-B5-A6-0A-00
master2.fs-ocp1.flashstack.local	10.106.1.34	10.106.0.22	00-25-B5-A6-0A-01
master3.fs-ocp1.flashstack.local	10.106.1.35	10.106.0.23	00-25-B5-A6-0A-02
worker1.fs-ocp1.flashstack.local	10.106.1.36	10.106.0.24	00-25-B5-A6-0A-03
worker2.fs-ocp1.flashstack.local	10.106.1.37	10.106.0.25	00-25-B5-A6-0A-09
worker3.fs-ocp1.flashstack.local	10.106.1.38	10.106.0.26	00-25-B5-A6-0A-0B

Step 8. From Table 13, enter the hostnames, IP addresses, and MAC addresses as reservations in your DHCP and DNS server(s) or configure the DHCP server to dynamically update DNS.

Step 9. You will also need to pipe VLAN interfaces for all 2 storage VLANs (3010 and 3020) and 2 management VLANs (1061 and 1062) into your DHCP server(s) and assign IPs in the storage networks on those interfaces. Then create a DHCP scope for each management and storage VLANs with appropriate subnets. Ensure that the IPs assigned by the scope do not overlap with already consumed IPs (like FlashArray//XL170 storage iSCSI interface IPs and OpenShift reserved IPs). Either enter the nodes in the DNS server or configure the DHCP server to forward entries to the DNS server. For the cluster nodes, create reservations to map the hostnames to the desired IP addresses.

Step 10. Setup either a VM (installer/bastion node) or spare server with the network interface connected to the Bare Metal VLAN and install either Red Hat Enterprise Linux (RHEL) 9.4 or Rocky Linux 9.4 “Server with GUI” and create an administrator user. Once the VM or host is up and running, update it and install and configure XRDP. Connect to this host with a Windows Remote Desktop client as the admin user.

Step 11. ssh into the installer node VM, open a terminal session and create an SSH key pair to use to communicate with the OpenShift hosts:

```
cd
ssh-keygen -t ed25519 -N '' -f ~/.ssh/id_ed25519
```

Step 12. Copy the public SSH key to the user directory:

```
cp ~/.ssh/id_ed25519.pub ~/
```

Step 13. Add the private key to the ssh-agent:

```
sshadd ~/.ssh/id_ed25519
```

Procedure 2. Install Red Hat OpenShift Container Platform using the Assisted Installer

Step 1. Launch Firefox and connect to https://console.redhat.com/openshift/cluster-list. Log into your Red Hat account.

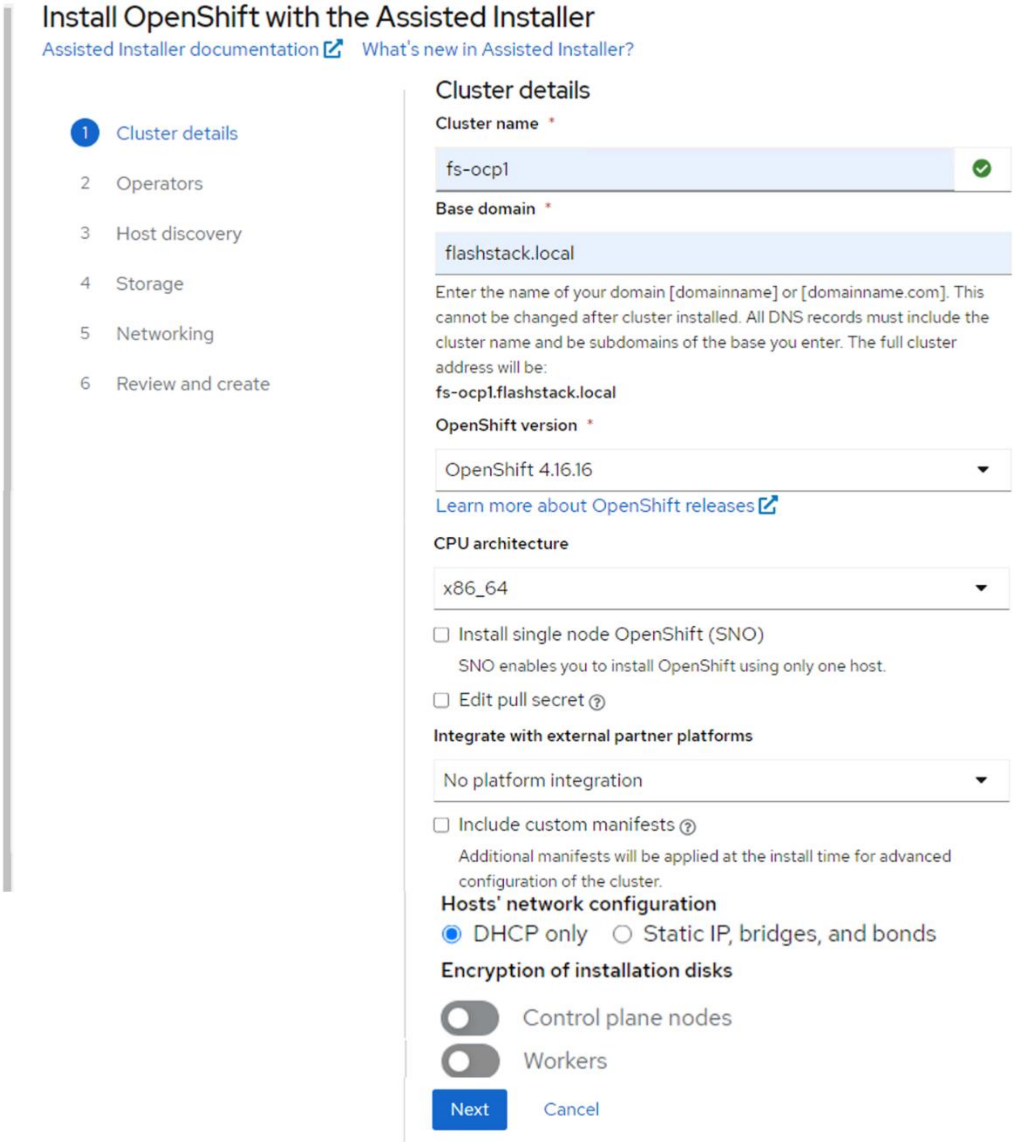
Step 2. Click **Create cluster** to create an OpenShift cluster.

Step 3. Select **Datacenter** and then select **Bare Metal (x86_64)**.

Step 4. Select **Interactive** to launch the Assisted Installer.

Step 5. Provide the cluster name and base domain.

Step 6. Select the latest OpenShift version, scroll down and click **Next**.



Install OpenShift with the Assisted Installer

Assisted Installer documentation | What's new in Assisted Installer?

1 Cluster details
2 Operators
3 Host discovery
4 Storage
5 Networking
6 Review and create

Cluster details

Cluster name * : fs-ocp1

Base domain * : flashstack.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:
fs-ocp1.flashstack.local

OpenShift version * : OpenShift 4.16.16

Learn more about OpenShift releases

CPU architecture : x86_64

□ Install single node OpenShift (SNO)
SNO enables you to install OpenShift using only one host.

□ Edit pull secret

Integrate with external partner platforms : No platform integration

□ 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

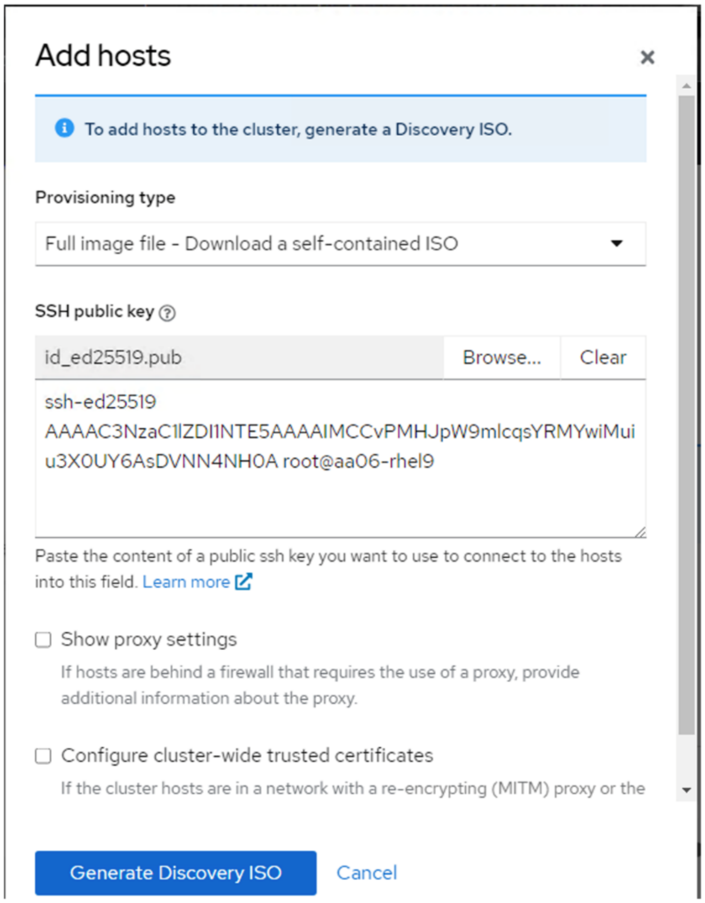
Encryption of installation disks
Control plane nodes (off)
Workers (off)

Step 7. Select the latest OpenShift version, scroll down and click **Next**.

Step 8. Select **Install OpenShift Virtualization** operator and click **Next**.

Step 9. Click **Add hosts**.

Step 10. Under provisioning type, from the drop-down list select the **Full Image file**. Under SSH public key, click **Browse** and browse to, select, and open the id_ed25519.pub file. The contents of the public key should now appear in the box. Click **Generate Discovery ISO** and click **Download Discovery ISO** to download the Discovery ISO.



Add hosts

To add hosts to the cluster, generate a Discovery ISO.

Provisioning type

Full image file - Download a self-contained ISO

SSH public key

id_ed25519.pub

```
ssh-ed25519
AAAAC3NzaC1lZDI1NTE5AAAAIMCCvPMHJpW9mlcqsYRMYwiMui
u3X0UY6AsDVNN4NH0A root@aa06-rhel9
```

Paste the content of a public ssh key you want to use to connect to the hosts into this field. Learn more

□ Show proxy settings

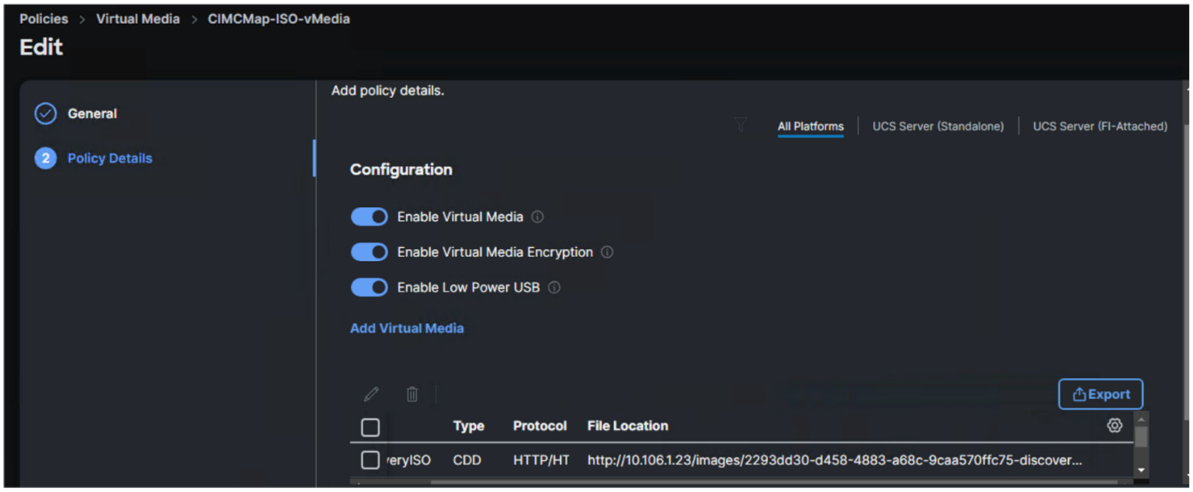
If hosts are behind a firewall that requires the use of a proxy, provide additional information about the proxy.

□ Configure cluster-wide trusted certificates

If the cluster hosts are in a network with a re-encrypting (MITM) proxy or the

Step 11. Copy the Discovery ISO to a http or https file share server, use a web browser to get a copy of the URL for the Discovery ISO.

Step 12. Log into **Cisco Intersight** and update the virtual Media policy with the Discovery ISO URL as shown below. This Discovery ISO image will be mapped to the server using CIMC Mapped DVD option defined in the Boot policy.



Policies > Virtual Media > CIMCMap-ISO-vMedia

Edit

Add policy details.

Configuration

● Enable Virtual Media

● Enable Virtual Media Encryption

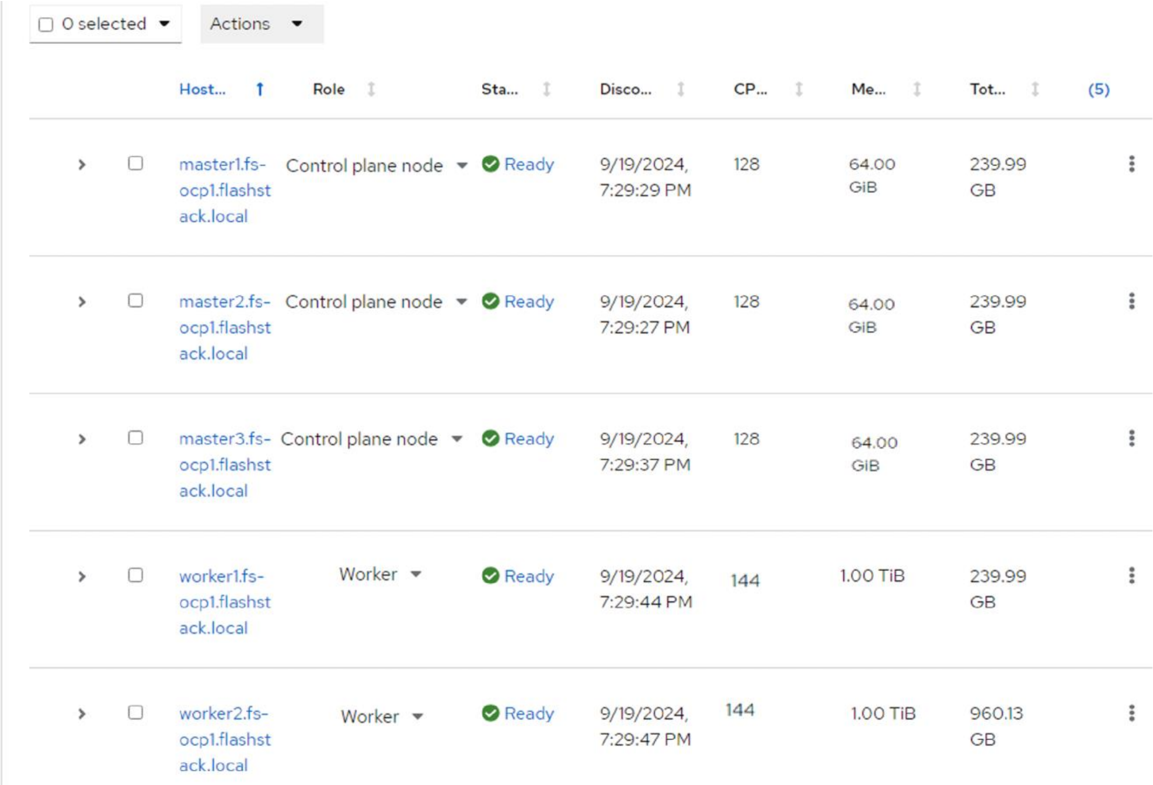
● Enable Low Power USB

□		Type	Protocol	File Location
□	eryISO	CDD	HTTP/HT	http://10.106.1.23/images/2293dd30-d458-4883-a68c-9caa570ffc75-discover...

Note: To demonstrate the OpenShift cluster expansion (adding additional worker node), only the first five nodes (3 master/control and 2 workers) will be used for the initial OpenShift cluster deployment. The sixth node is reserved for now and will be used for cluster expansion which will be discussed in the following sections.

Step 13. Reset first five UCSX-201c M7 server by selecting **Operate > Power > Reset System**.

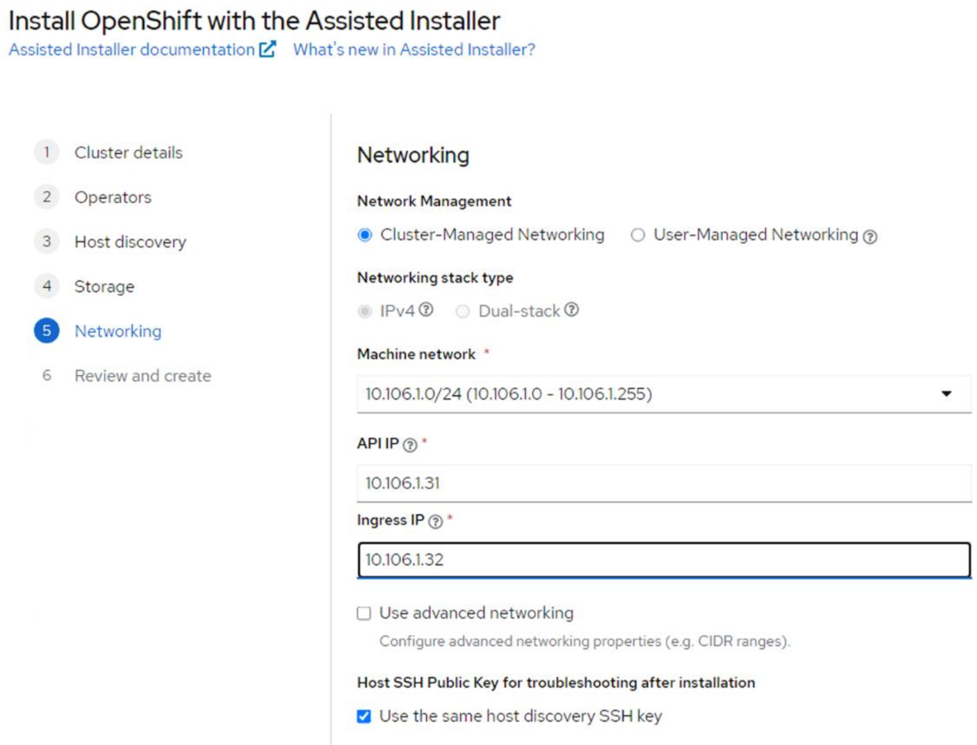
Step 14. When all five servers have booted "RHEL CoreOS (Live)" from the Discovery ISO, they will appear in the Assisted Installer. Use the drop-down lists under Role to assign the appropriate server roles. Scroll down and click **Next**.



		Host...	Role	Sta...	Disco...	CP...	Me...	Tot...	(5)
›	☐	master1.fs-ocp1.flashstack.local	Control plane node ▾	✓ Ready	9/19/2024, 7:29:29 PM	128	64.00 GiB	239.99 GB	⋮
›	☐	master2.fs-ocp1.flashstack.local	Control plane node ▾	✓ Ready	9/19/2024, 7:29:27 PM	128	64.00 GiB	239.99 GB	⋮
›	☐	master3.fs-ocp1.flashstack.local	Control plane node ▾	✓ Ready	9/19/2024, 7:29:37 PM	128	64.00 GiB	239.99 GB	⋮
›	☐	worker1.fs-ocp1.flashstack.local	Worker ▾	✓ Ready	9/19/2024, 7:29:44 PM	144	1.00 TiB	239.99 GB	⋮
›	☐	worker2.fs-ocp1.flashstack.local	Worker ▾	✓ Ready	9/19/2024, 7:29:47 PM	144	1.00 TiB	960.13 GB	⋮

Step 15. Expand each node and confirm the role of the M.2 disk is set to Installation disk. Click **Next**.

Step 16. Under Network Management, make sure Cluster-Managed Networking is selected. Under Machine network, from the drop-down list, select the subnet for the BareMetal VLAN. Enter the API IP for the api.cluster.basedomain entry in the DNS servers. For the Ingress IP, enter the IP for the *.apps.cluster.basedomain entry in the DNS servers.



Install OpenShift with the Assisted Installer

Assisted Installer documentation   What's new in Assisted Installer?

Networking

Network Management

● Cluster-Managed Networking ○ User-Managed Networking

Networking stack type

● IPv4 ○ Dual-stack

Machine network *

10.106.1.0/24 (10.106.1.0 - 10.106.1.255)

API IP *

10.106.1.31

Ingress IP *

10.106.1.32

□ Use advanced networking

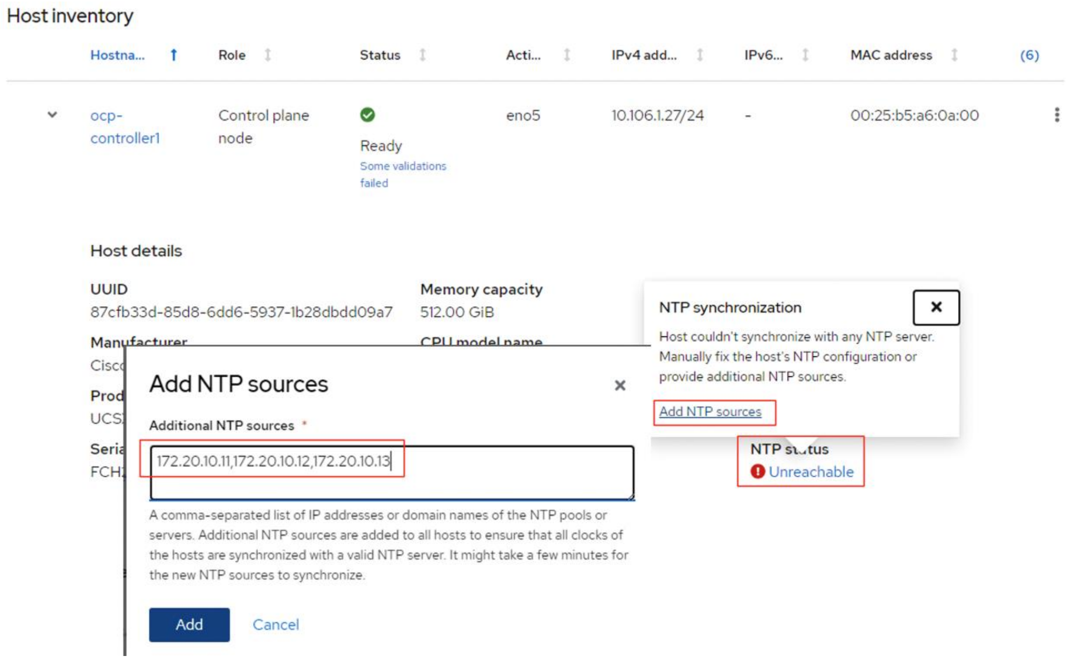
Configure advanced networking properties (e.g. CIDR ranges).

Host SSH Public Key for troubleshooting after installation

☑ Use the same host discovery SSH key

Step 17. Scroll down. All nodes should have a status of Ready.

Note: If you see insufficient warning message for the nodes due to missing ntp server information, expand one of the nodes, click **Add NTP Sources** and provide ntp servers IPs separated by a comma.



Host inventory

Hostna...	Role	Status	Acti...	IPv4 add...	IPv6...	MAC address	(6)
ocp-controller1	Control plane node	Ready Some validations failed	eno5	10.106.1.27/24	-	00:25:b5:a6:0a:00	

Host details

UUID
87cfb33d-85d8-6dd6-5937-1b28dbdd09a7

Memory capacity
512.00 GiB

Manufacturer
Cisco

CPU model name

Prod

UCS

Seria

FCH

NTP synchronization

Host couldn't synchronize with any NTP server. Manually fix the host's NTP configuration or provide additional NTP sources.

Add NTP sources

NTP status
Unreachable

Add NTP sources

Additional NTP sources *

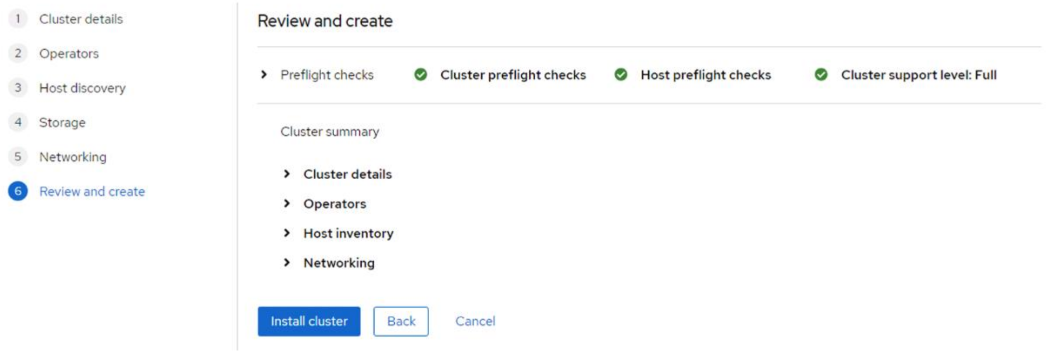
172.20.10.11,172.20.10.12,172.20.10.13

A comma-separated list of IP addresses or domain names of the NTP pools or servers. Additional NTP sources are added to all hosts to ensure that all clocks of the hosts are synchronized with a valid NTP server. It might take a few minutes for the new NTP sources to synchronize.

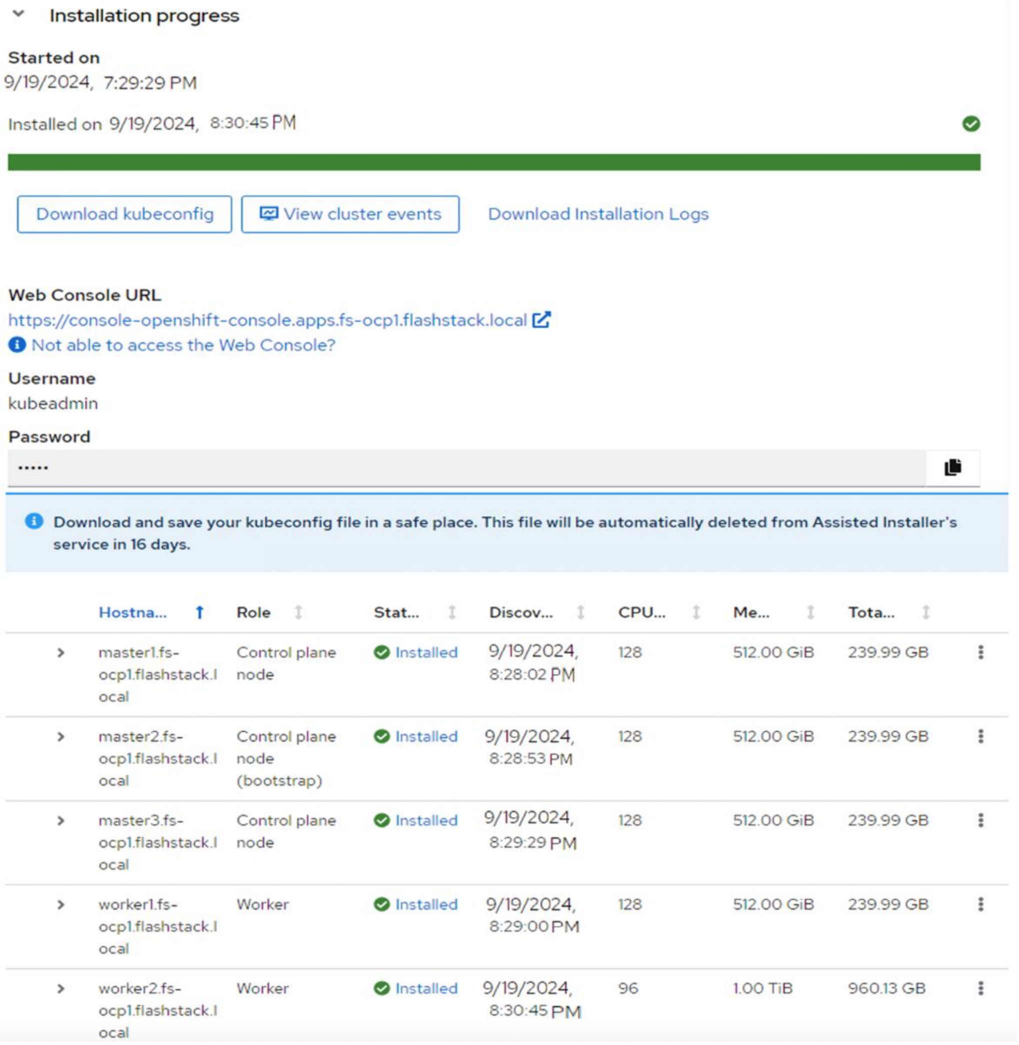
Note: You would see a warning message on each worker node around having multiple network devices on the L2 network. To resolve this, ssh into each worker and de-activate eno8,eno9 and eno10 interfaces using nmtui utility.

Step 18. When all the nodes are in ready status, click **Next**.

Step 19. Review the information and click **Install cluster** to begin the cluster installation.



Step 20. On the Installation progress page, expand the Host inventory. The installation will take 30-45 minutes. When installation is complete, all nodes will show a Status of Installed.



Started on
9/19/2024, 7:29:29 PM

Installed on 9/19/2024, 8:30:45 PM

Web Console URL
https://console-openshift-console.apps.fs-ocp1.flashstack.local
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 16 days.

Hostna...	Role	Stat...	Discov...	CPU...	Me...	Tota...
master1.fs-ocp1.flashstack.local	Control plane node	Installed	9/19/2024, 8:28:02 PM	128	512.00 GiB	239.99 GB
master2.fs-ocp1.flashstack.local	Control plane node (bootstrap)	Installed	9/19/2024, 8:28:53 PM	128	512.00 GiB	239.99 GB
master3.fs-ocp1.flashstack.local	Control plane node	Installed	9/19/2024, 8:29:29 PM	128	512.00 GiB	239.99 GB
worker1.fs-ocp1.flashstack.local	Worker	Installed	9/19/2024, 8:29:00 PM	128	512.00 GiB	239.99 GB
worker2.fs-ocp1.flashstack.local	Worker	Installed	9/19/2024, 8:30:45 PM	96	1.00 TiB	960.13 GB

Step 21. Select **Download kubeconfig** to download the kubeconfig file. In a terminal window, setup a cluster directory and save credentials:

```
cd
mkdir <clustername> # for example, ocp
cd <clustername>
mkdir auth
```

```
cd auth
mv ~/Downloads/kubeconfig ./
mkdir ~/.kube
cp kubeconfig ~/.kube/config
```

Step 22. In the Assisted Installer, click the icon to copy the kubeadmin password:

```
echo <paste password> > ./kubeadmin-password
```

Step 23. Click **Open console** to launch the OpenShift Console. Use kubeadmin and the kubeadmin password to login. Click the **?** mask located on the right most corner of the page. Links for various tools are provided in that page. Download oc for Linux for x86_64 and virtctl for Linux for x86_64 Common Line Tools.

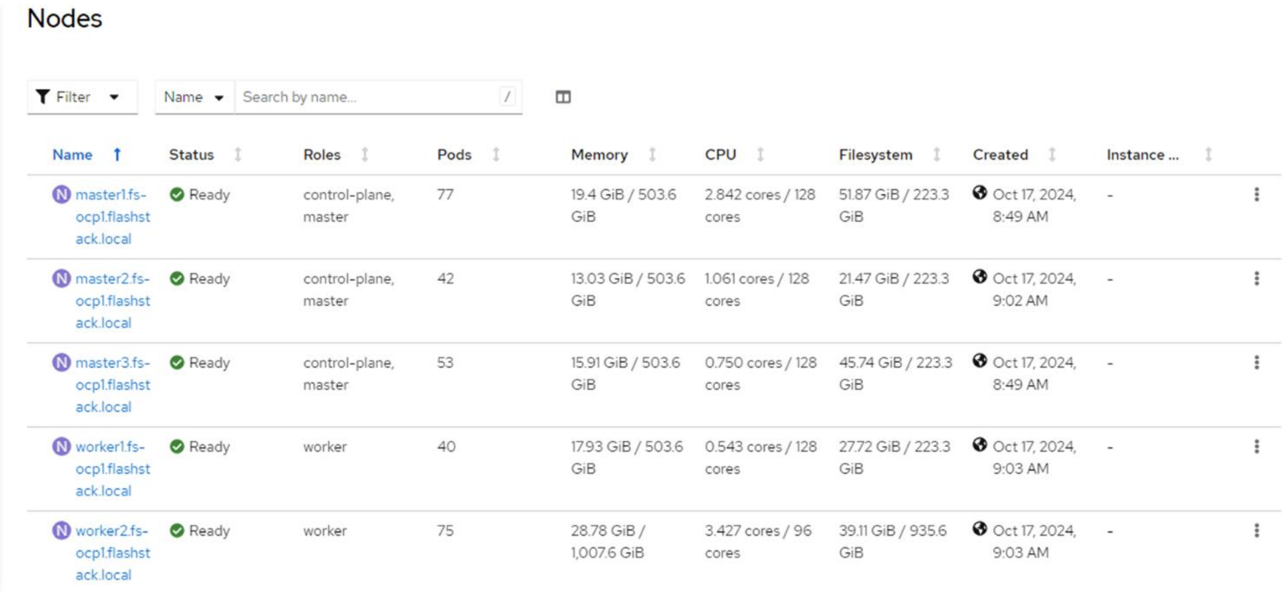
```
cd ..
mkdir client
cd client
ls ~/Downloads
mv ~/Downloads/oc.tar.gz ./
mv ~/Downloads/virtctl.tar.gz ./
tar xvf oc.tar
tar xvf virtctl.tar.gf
ls
sudo mv oc /usr/local/bin/
sudo mv virtcl /usr/local/bin/
sudo mv kubectl /usr/local/bin/
oc get nodes
```

Step 24. To enable oc tab completion for bash, run the following:

```
oc completion bash > oc_bash_completion
sudo mv oc_bash_completion /etc/bash_completion.d/
```

Step 25. In Cisco Intersight, edit the Virtual Media policy and remove the link to the Discovery ISO. Click **Save & Deploy** then click **Save & Proceed**. Do not select "Reboot Immediately to Activate." Click **Deploy**. The virtual media mount will be removed from the servers without rebooting them.

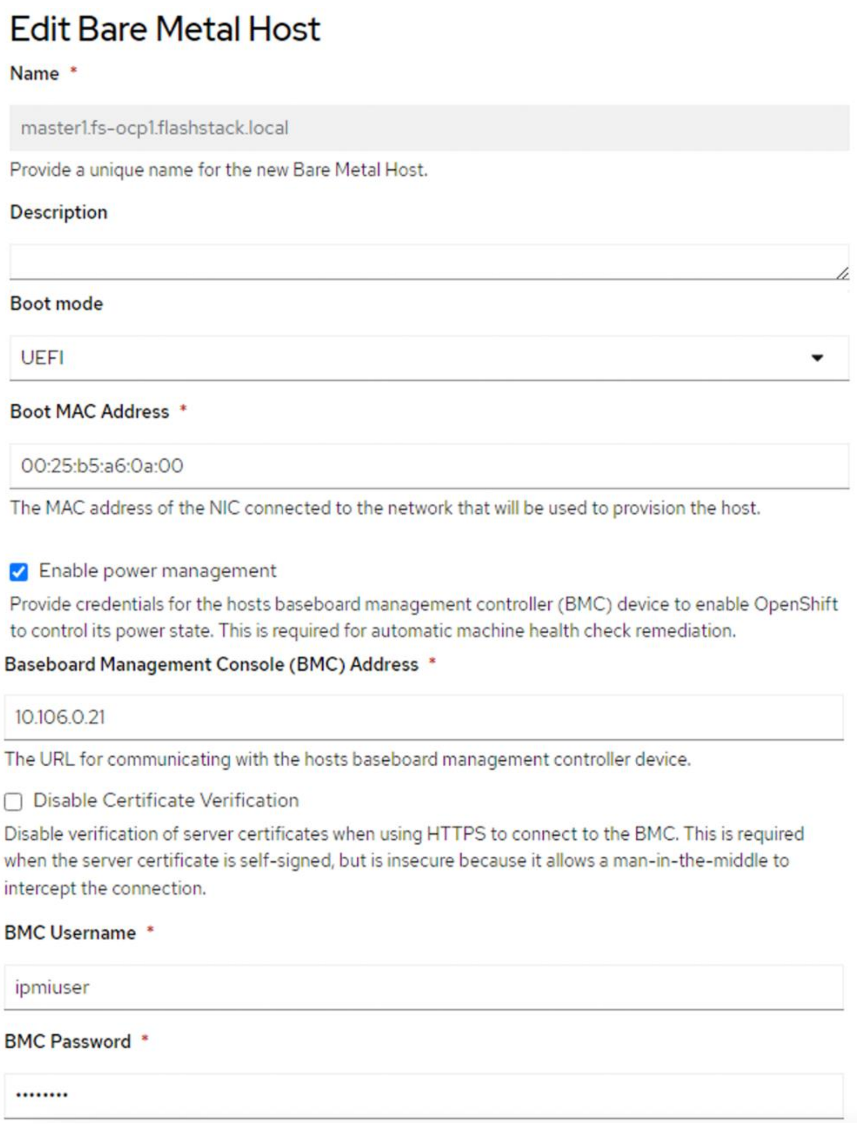
Step 26. In Firefox, in the Assisted Installer page, click **Open console** to launch the OpenShift Console. Use kubeadmin and the kubeadmin password to login. On the left, select **Compute > Nodes** to see the status of the OpenShift nodes.



Name	Status	Roles	Pods	Memory	CPU	Filesystem	Created	Instance ...
master1.fs-ocp1.flashstack.local	Ready	control-plane, master	77	19.4 GiB / 503.6 GiB	2.842 cores / 128 cores	51.87 GiB / 223.3 GiB	Oct 17, 2024, 8:49 AM	-
master2.fs-ocp1.flashstack.local	Ready	control-plane, master	42	13.03 GiB / 503.6 GiB	1.061 cores / 128 cores	21.47 GiB / 223.3 GiB	Oct 17, 2024, 9:02 AM	-
master3.fs-ocp1.flashstack.local	Ready	control-plane, master	53	15.91 GiB / 503.6 GiB	0.750 cores / 128 cores	45.74 GiB / 223.3 GiB	Oct 17, 2024, 8:49 AM	-
worker1.fs-ocp1.flashstack.local	Ready	worker	40	17.93 GiB / 503.6 GiB	0.543 cores / 128 cores	27.72 GiB / 223.3 GiB	Oct 17, 2024, 9:03 AM	-
worker2.fs-ocp1.flashstack.local	Ready	worker	75	28.78 GiB / 1,007.6 GiB	3.427 cores / 96 cores	39.11 GiB / 935.6 GiB	Oct 17, 2024, 9:03 AM	-

Step 27. In the Red Hat OpenShift console, select **Compute > Bare Metal Hosts**. For each Bare Metal Host, click the three dots to the right of the host and select **Edit Bare Metal Host**. Select **Enable power management**.

Step 28. From Table 13, fill in the BMC Address. Also, make sure the Boot MAC Address matches the MAC address in Table 13. For the BMC Username and BMC Password, use what was entered into the Cisco Intersight IPMI over LAN policy. Click **Save** to save the changes. Repeat this step for all Bare Metal Hosts.



Edit Bare Metal Host

Name *

master1.fs-ocp1.flashstack.local

Provide a unique name for the new Bare Metal Host.

Description

Boot mode

UEFI

Boot MAC Address *

00:25:b5:a6:0a:00

The MAC address of the NIC connected to the network that will be used to provision the host.

☑ Enable power management

Provide credentials for the hosts baseboard management controller (BMC) device to enable OpenShift to control its power state. This is required for automatic machine health check remediation.

Baseboard Management Console (BMC) Address *

10.106.0.21

The URL for communicating with the hosts baseboard management controller device.

☐ Disable Certificate Verification

Disable verification of server certificates when using HTTPS to connect to the BMC. This is required when the server certificate is self-signed, but is insecure because it allows a man-in-the-middle to intercept the connection.

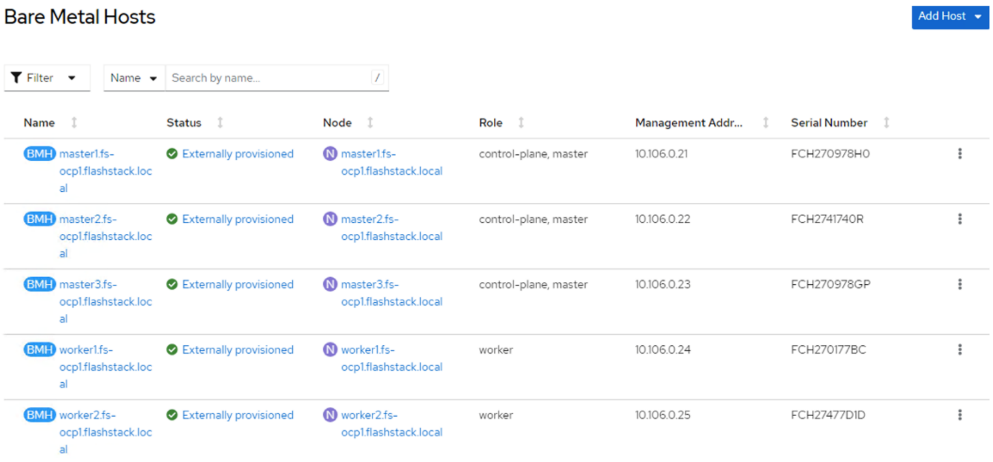
BMC Username *

ipmiuser

BMC Password *

••••••••

Step 29. Select **Compute > Bare Metal Hosts**. When all hosts have been configured, the Status displays “Externally provisioned,” and the Management Address are populated. You can now manage power on the OpenShift hosts from the OpenShift console.



Bare Metal Hosts

Name	Status	Node	Role	Management Addr...	Serial Number
BMH master1.fs-ocp1.flashstack.local	✓ Externally provisioned	N master1.fs-ocp1.flashstack.local	control-plane, master	10.106.0.21	FCH270978H0
BMH master2.fs-ocp1.flashstack.local	✓ Externally provisioned	N master2.fs-ocp1.flashstack.local	control-plane, master	10.106.0.22	FCH2741740R
BMH master3.fs-ocp1.flashstack.local	✓ Externally provisioned	N master3.fs-ocp1.flashstack.local	control-plane, master	10.106.0.23	FCH270978GP
BMH worker1.fs-ocp1.flashstack.local	✓ Externally provisioned	N worker1.fs-ocp1.flashstack.local	worker	10.106.0.24	FCH270177BC
BMH worker2.fs-ocp1.flashstack.local	✓ Externally provisioned	N worker2.fs-ocp1.flashstack.local	worker	10.106.0.25	FCH27477D1D

Note: For an IPMI connection to the server, use the BMC IP address. However, for Redfish to connect to the server, use this format for the BMS address; redfish://<BMC IP>/redfish/v1/Systems/<server Serial Number> and make sure to check **Disable Certificate Verification.** For Instance, for master1.fs-ocp1.flashstack.local Bare Metal node, the redfish BMC management Address will be: redfish://10.106.0.21/redfish/v1/Systems/FCH270978H0. When using the redfish to connect to the server, it is critical to select the **Disable Certificate Verification** checkbox.

Note: It is recommended to reserve enough resources (cpus and memory) for system components like kubelet and kube-proxy on the nodes. 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.

Step 30. To automatically determine and allocate the system-reserved resources on nodes, create a KubeletConfig custom resource (CR) to set the autoSizingReserved: true parameter as shown below and apply the machine configuration files:

```
cat dynamic-resource-alloc-workers.yaml
apiVersion: machineconfiguration.openshift.io/v1
kind: KubeletConfig
metadata:
  name: dynamic-node-master
spec:
  autoSizingReserved: true
  machineConfigPoolSelector:
    matchLabels:
      pools.operator.machineconfiguration.openshift.io/master: ""

cat dynamic-resource-alloc-master.yaml
apiVersion: machineconfiguration.openshift.io/v1
kind: KubeletConfig
metadata:
  name: dynamic-resource-allow-master
spec:
  autoSizingReserved: true
  machineConfigPoolSelector:
    matchLabels:
      pools.operator.machineconfiguration.openshift.io/master: ""


oc apply -f dynamic-resource-alloc-workers.yaml
oc apply -f dynamic-resource-alloc-master.yaml
```

To manually configure the resources for the system components on the nodes, go to: https://docs.openshift.com/container-platform/4.16/nodes/nodes/nodes-nodes-resources-configuring.html#nodes-nodes-resources-configuring-setting_nodes-nodes-resources-configuring

Procedure 3. Install NVIDIA GPU Operator (Optional)

If you have GPUs installed in your Cisco UCS servers, you need to install the Node Feature Discovery (NFD) Operator to detect NVIDIA GPUs and the NVIDIA GPU Operator to make these GPUs available to containers and virtual machines.

Step 1. In the OpenShift Container Platform web console, click **Operators > OperatorHub**.

Step 2. Type Node Feature in the Filter box and then click the **Node Feature Discovery Operator with Red Hat** in the upper right corner. Click **Install**.

Step 3. Do not change any settings and click **Install**.

Step 4. When the Install operator is ready for use, click **View Operator**.

Step 5. In the bar to the right of Details, click **NodeFeatureDiscovery**.

Step 6. Click **Create NodeFeatureDiscovery**.

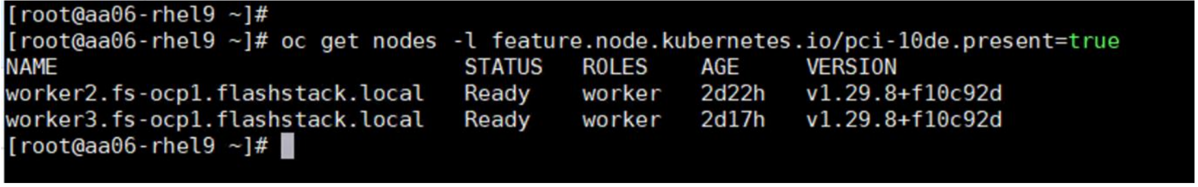
Step 7. Click **Create**.

Step 8. When the nfd-instance has a status of Available, Upgradeable, select **Compute > Nodes**.

Step 9. Select a node that has one or more GPUs and then select **Details**.

Step 10. The label **feature.node.kubernetes.io/pci-10de.present=true** should be present on the host:

This label appears on all nodes with GPUs:



```
[root@aa06-rhel9 ~]#
[root@aa06-rhel9 ~]# oc get nodes -l feature.node.kubernetes.io/pci-10de.present=true
NAME                              STATUS   ROLES    AGE     VERSION
worker2.fs-ocp1.flashstack.local   Ready    worker   2d22h   v1.29.8+f10c92d
worker3.fs-ocp1.flashstack.local   Ready    worker   2d17h   v1.29.8+f10c92d
[root@aa06-rhel9 ~]#
```

Step 11. Return to **Operators > OperatorHub**.

Step 12. Type NVIDIA in the Filter box and then click the **NVIDIA GPU Operator**. Click **Install**.

Step 13. Do not change any settings and click **Install**.

Step 14. When the Install operator is ready for use, click **View Operator**.

Step 15. In the bar to the right of Details, click **ClusterPolicy**.

Step 16. Click **Create ClusterPolicy**.

Step 17. Do not change any settings and scroll down and click **Create**. This will install the latest GPU driver.

Step 18. Wait for the gpu-cluster-policy Status to become Ready.

Step 19. Connect to a terminal window on the OpenShift Installer machine. Type the following commands. The output shown is for two servers that are equipped with GPUs:

```
oc project nvidia-gpu-operator
Already on project "nvidia-gpu-operator" on server "https://api.fs-ocp1.flashstack.local:6443".
oc get pods
NAME                                                READY   STATUS      RESTARTS        AGE
gpu-feature-discovery-cp9cg                         1/1     Running     0               5m23s
gpu-feature-discovery-gdt7j                         1/1     Running     0               5m14s
gpu-operator-7d8447447-9gnpq                        1/1     Running     0               2m49s
nvidia-container-toolkit-daemonset-4js4p            1/1     Running     0               5m23s
nvidia-container-toolkit-daemonset-wr6gv            1/1     Running     0               5m14s
nvidia-cuda-validator-828rz                         0/1     Completed   0               2m56s
nvidia-dcgm-44zbh                                   1/1     Running     0               5m23s
nvidia-dcgm-exporter-kq7jp                          1/1     Running     2 (3m ago)      5m23s
nvidia-dcgm-exporter-thjlc                          1/1     Running     2 (2m33s ago)   5m14s
nvidia-dcgm-h8mzq                                   1/1     Running     0               5m14s
nvidia-device-plugin-daemonset-pz87g                1/1     Running     0               5m14s
nvidia-device-plugin-daemonset-x9hrk                1/1     Running     0               5m23s
nvidia-driver-daemonset-416.94.202410020522-0-6hm42   2/2     Running     0               6m17s
nvidia-driver-daemonset-416.94.202410020522-0-nshpt   2/2     Running     0               6m17s
nvidia-node-status-exporter-hv8xp                   1/1     Running     0               6m16s
nvidia-node-status-exporter-msv56                   1/1     Running     0               6m16s
nvidia-operator-validator-66b4x                     1/1     Running     0               5m14s
nvidia-operator-validator-km9tb                     1/1     Running     0               5m23s
```

Step 20. Connect to one of the nvidia-driver-daemonset containers and view the GPU status:

```
oc exec -it nvidia-driver-daemonset-416.94.202410020522-0-6hm42 -- bash
[root@nvidia-driver-daemonset-416 drivers]# nvidia-smi
Mon Nov  4 15:36:49 2024
+-----------------------------------------------------------------------------------------+
| NVIDIA-SMI 550.127.05             Driver Version: 550.127.05     CUDA Version: 12.4     |
|-----------------------------------------+------------------------+----------------------+
| GPU  Name                 Persistence-M | Bus-Id          Disp.A | Volatile Uncorr. ECC |
| Fan  Temp   Perf          Pwr:Usage/Cap |           Memory-Usage | GPU-Util  Compute M. |
|                                         |                        |               MIG M. |
|=========================================+========================+======================|
|   0  NVIDIA L40S                    On  |   00000000:3D:00.0 Off |                    0 |
| N/A   28C    P8             35W /  350W |       1MiB /  46068MiB |      0%      Default |
|                                         |                        |                  N/A |
+-----------------------------------------+------------------------+----------------------+
|   1  NVIDIA L40S                    On  |   00000000:E1:00.0 Off |                    0 |
| N/A   28C    P8             36W /  350W |       1MiB /  46068MiB |      0%      Default |
|                                         |                        |                  N/A |
+-----------------------------------------+------------------------+----------------------+

+-----------------------------------------------------------------------------------------+
| Processes:                                                                              |
|  GPU   GI   CI        PID   Type   Process name                              GPU Memory |
|        ID   ID                                                               Usage      |
|=========================================================================================|
|  No running processes found                                                             |
+-----------------------------------------------------------------------------------------+
```

Procedure 4. Enable the GPU Monitoring Dashboard (Optional)

Step 1. Using https://docs.nvidia.com/datacenter/cloud-native/openshift/latest/enable-gpu-monitoring-dashboard.html, enable to GPU Monitoring Dashboard to monitor GPUs in the OpenShift Web-Console.

Expand OpenShift Cluster by Adding a Worker Node

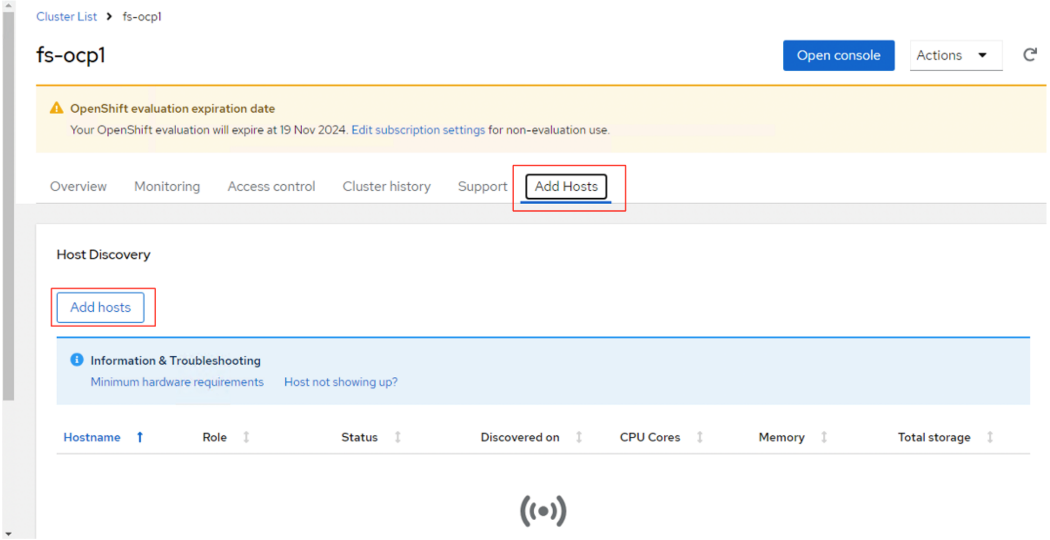
This chapter provides detailed steps to scale up the worker nodes of OpenShift cluster by adding a new worker node to the existing cluster. For this exercise, the sixth blade in the chassis will be used and be part of the cluster by the end of this exercise.

Note: This section assumes that a new server profile is already derived from the existing template and assigned to the new server successfully.

Procedure 1. OpenShift Cluster expansion

Step 1. Launch Firefox and Launch https://console.redhat.com/openshift/cluster-list. Log into your Red Hat account.

Step 2. Click your cluster name and go to **Add Hosts**.



OpenShift evaluation expiration date

Your OpenShift evaluation will expire at 19 Nov 2024. Edit subscription settings for non-evaluation use.

Information & Troubleshooting

Minimum hardware requirements   Host not showing up?

Hostname	Role	Status	Discovered on	CPU Cores	Memory	Total storage

Step 3. Click **Add hosts** under Host Discovery.

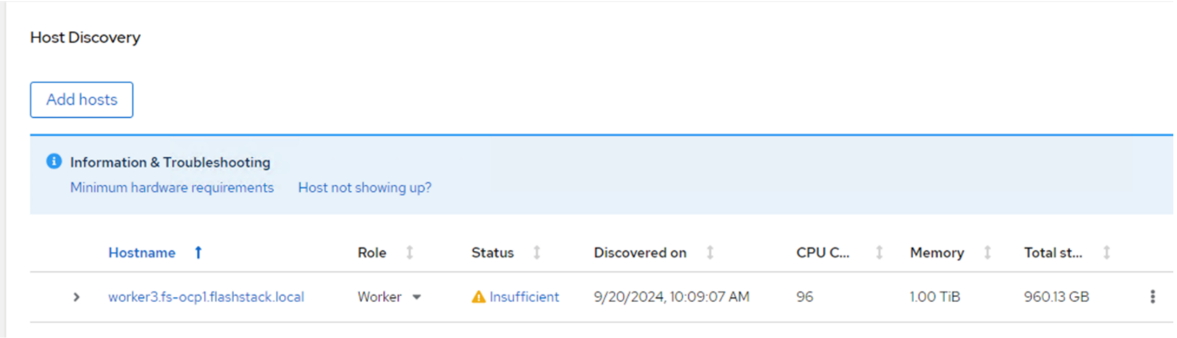
Step 4. On the Add hosts wizard, select **x86_64** for the CPU architecture and **DHCP Only** for the Host's network configuration. Click **Next.**

Step 5. For the provision type select **Full image** file from the drop-down list, for SSH public key browse or copy/paste the contents of id-ed25519.pub file. Click **Generate Discovery ISO** and when the file is generated and click **Download Discovery ISO** file.

Step 6. Copy the Discovery ISO to a http or https file share server, use web browser to get a copy of the URL for the new Discovery ISO.

Step 7. Log into the Cisco Intersight and update the virtual Media policy as explained in the previous section. This Discovery ISO image is a mapped server using the CIMC Mapped DVD option. Now **Reset** sixth UCSX-201c M7 server by selecting **Power > Reset System.**

Step 8. When the server has booted "RHEL CoreOS (live)" from the newly generated Discovery ISO, it will appear in the assisted installer under **Add hosts.**

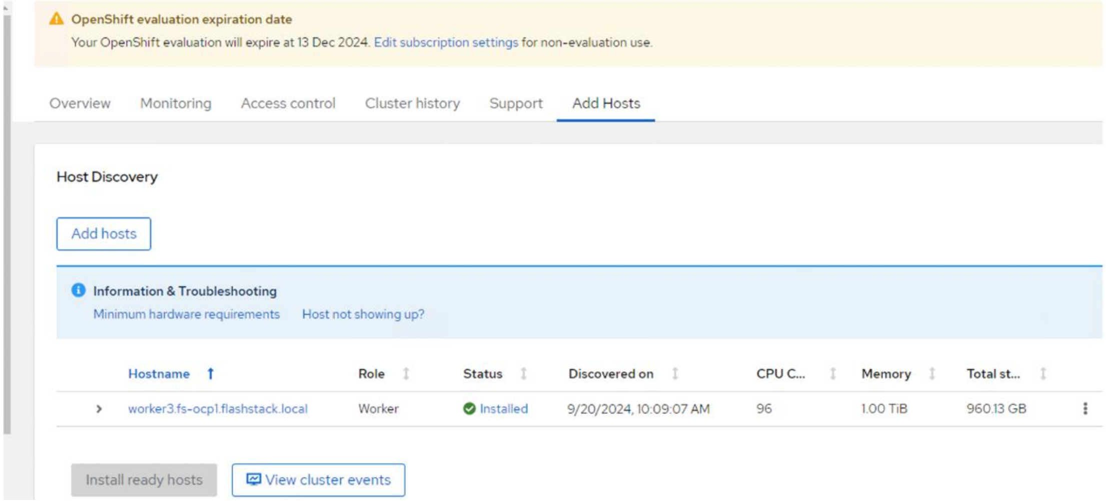


	Hostname	Role	Status	Discovered on	CPU C...	Memory	Total st...	
›	worker3.fs-ocp1.flashstack.local	Worker ▾	⚠ Insufficient	9/20/2024, 10:09:07 AM	96	1.00 TiB	960.13 GB	⋮

Note: If you see insufficient warning message for the node due to missing ntp server information, expand one of the nodes, click **Add NTP Sources** and provide ntp servers IPs separated by a comma.

Note: If a warning message appears stating you have multiple network devices on the L2 network, ssh into worker node and deactivate eno8,eno9, and eno10 interfaces using the **nmtui** utility.

Step 9. When the node status shows **Ready**, click **Install ready hosts**. After few minutes, the required components will be installed on the node and finally shows the status as **Installed**.

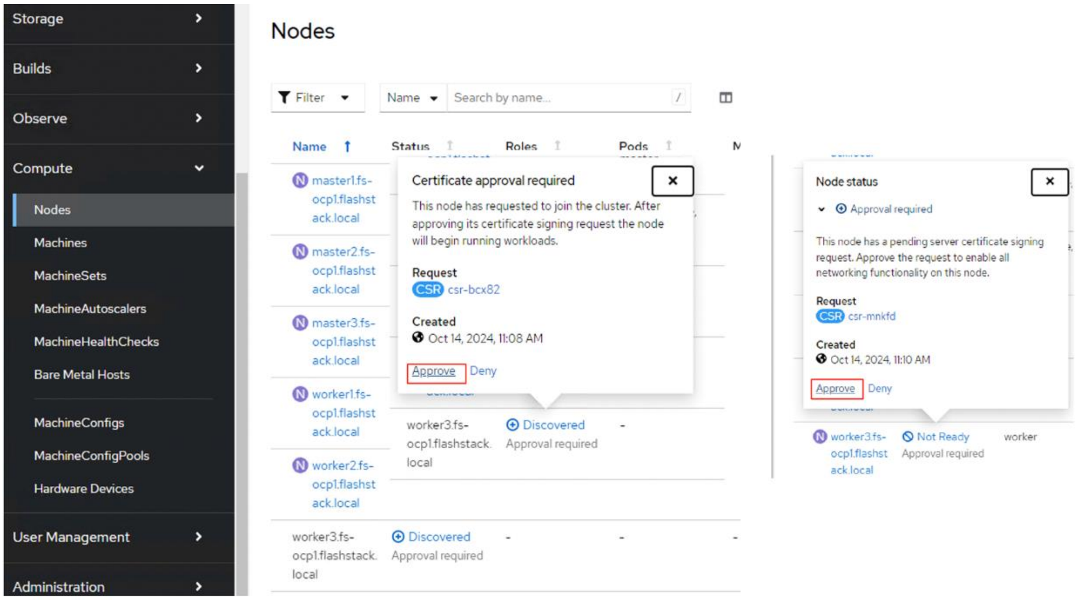

⚠ OpenShift evaluation expiration date

Your OpenShift evaluation will expire at 13 Dec 2024. Edit subscription settings for non-evaluation use.

	Hostname	Role	Status	Discovered on	CPU C...	Memory	Total st...	
›	worker3.fs-ocp1.flashstack.local	Worker	✓ Installed	9/20/2024, 10:09:07 AM	96	1.00 TiB	960.13 GB	⋮

Step 10. When the server successfully produces the CoreOS installed, log into Cisco Intersight, edit the vMedia policy and remove the virtual media mount. Go to **Profiles > Server Profiles** page, deploy the profile to the newly added worker profile without rebooting the host. The Inconsistent state on the remaining profiles should be cleared.

Step 11. Log into the cluster with kubeadmin user and go to **Compute** > **Nodes** > and select the newly added worker node and approve the Cluster join request of the worker node and request for server certificate signing.



Certificate approval required

This node has requested to join the cluster. After approving its certificate signing request the node will begin running workloads.

Request: CSR csr-bcx82

Created: Oct 14, 2024, 11:08 AM

Node status

Approval required

This node has a pending server certificate signing request. Approve the request to enable all networking functionality on this node.

Request: CSR csr-mnkfd

Created: Oct 14, 2024, 11:10 AM

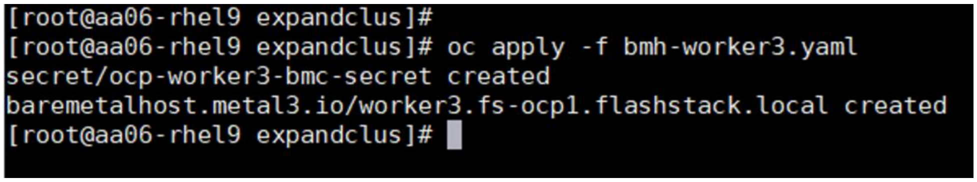
Step 12. Wait for few seconds and the node will be ready and the pods get scheduled on the newly added worker node.

Step 13. Create secret and BareMetalHost objects in openshift-machine-api namespace by executing the below manifest (bmh-worker3.yaml).

```
cat bmh-worker3.yaml
---
apiVersion: v1
kind: Secret
metadata:
  name: ocp-worker3-bmc-secret
  namespace: openshift-machine-api
type: Opaque
data:
  username: aXBtaXVzZXIK
  password: SDFnaFYwbHQK
---
apiVersion: metal3.io/v1alpha1
kind: BareMetalHost
metadata:
  name: worker3.fs-ocp1.flashstack.local
  namespace: openshift-machine-api
spec:
  online: True
  bootMACAddress: 00:25:B5:A6:0A:0B
  bmc:
    address: redfish://10.106.0.26/redfish/v1/Systems/FCH27477BZU
    credentialsName: ocp-worker3-bmc-secret
    disableCertificateVerification: True
  customDeploy:
    method: install_coreos
  externallyProvisioned: true
```

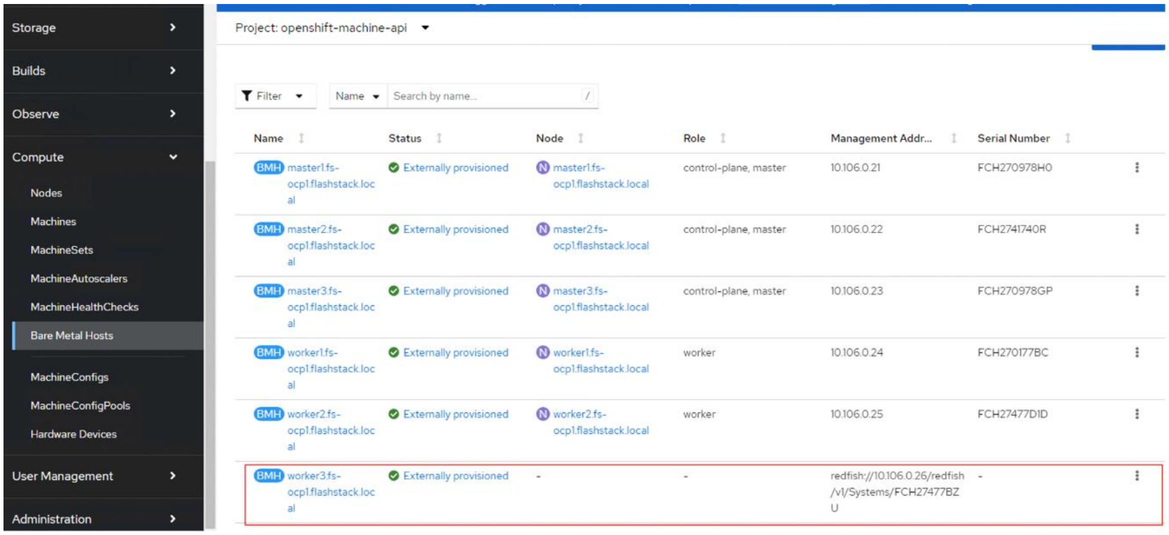
Note: The username and password shown in the above file are base64 encoded values.

Note: In this case, redfish connection is used for connecting to the server. 00:25:B5:A6:0A:0B is the MAC address of eno5 interface, 10.106.0.26 is the OOB management IP and FCH27477BZU is the serial number of the newly added worker node. These values are updated in the table 5. If you would like to use IPMI over LAN instead of redfish, just put the server's out of band management IP for the bmc address field.



```
[root@aa06-rhel9 expandclus]#
[root@aa06-rhel9 expandclus]# oc apply -f bmh-worker3.yaml
secret/ocp-worker3-bmc-secret created
baremetalhost.metal3.io/worker3.fs-ocp1.flashstack.local created
[root@aa06-rhel9 expandclus]#
```

A new entry will be created for the newly added worker node under Compute > Bare Metal Hosts.



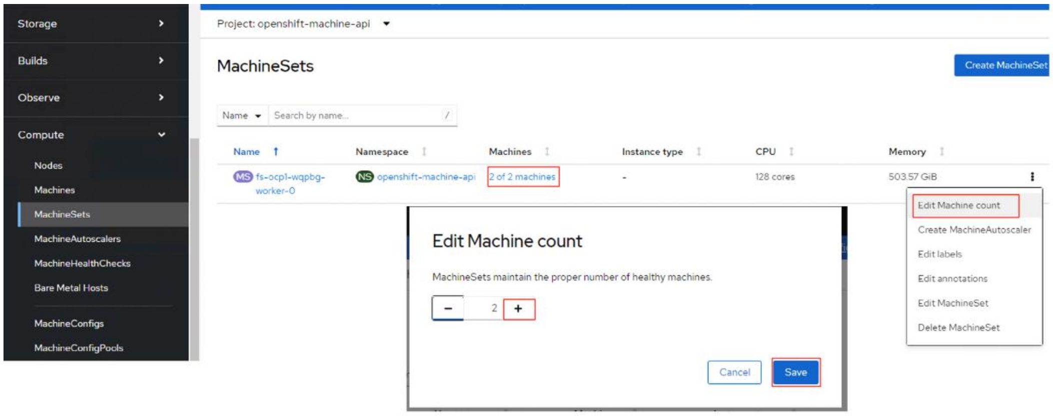
Project: openshift-machine-api

Name	Status	Node	Role	Management Addr...	Serial Number
BMH master1.fs-ocp1.flashstack.local	Externally provisioned	N master1.fs-ocp1.flashstack.local	control-plane, master	10.106.0.21	FCH270978H0
BMH master2.fs-ocp1.flashstack.local	Externally provisioned	N master2.fs-ocp1.flashstack.local	control-plane, master	10.106.0.22	FCH2741740R
BMH master3.fs-ocp1.flashstack.local	Externally provisioned	N master3.fs-ocp1.flashstack.local	control-plane, master	10.106.0.23	FCH270978GP
BMH worker1.fs-ocp1.flashstack.local	Externally provisioned	N worker1.fs-ocp1.flashstack.local	worker	10.106.0.24	FCH270177BC
BMH worker2.fs-ocp1.flashstack.local	Externally provisioned	N worker2.fs-ocp1.flashstack.local	worker	10.106.0.25	FCH27477D1D
BMH worker3.fs-ocp1.flashstack.local	Externally provisioned	-	-	redfish://10.106.0.26/redfish/v1/Systems/FCH27477BZU	-

Note: The node field is not yet populated for this bare metal host as it is not yet logically linked to any OpenShift Machine.

Note: Since there are only two machines (workers) in the cluster, the worker MachineSets count needs to be increased from **2** to **3**.

Step 14. To increase the worker's machineset count, go to **Compute > MachineSets.** Click the ellipses of worker-0 machineset and select **Edit Machine Count** and increase the count from **2** to **3.** Click **Save.**



Project: openshift-machine-api

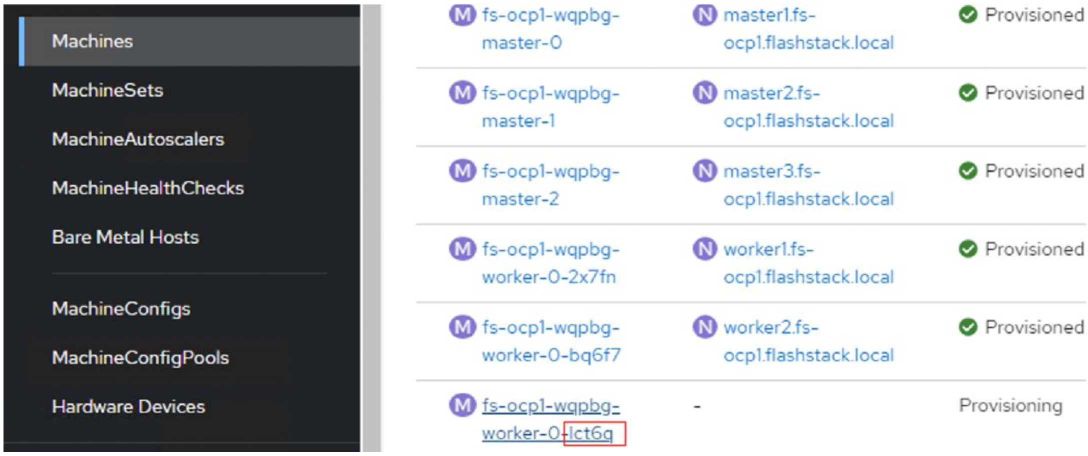
MachineSets

Name	Namespace	Machines	Instance type	CPU	Memory
MS fs-ocp1-wqpbg-worker-0	NS openshift-machine-api	2 of 2 machines	-	128 cores	503.57 GiB

Edit Machine count

MachineSets maintain the proper number of healthy machines.

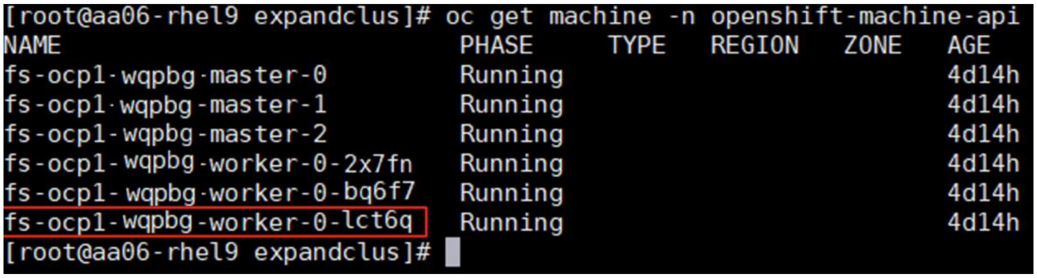
A new worker node will be provisioned to match to the worker machine count of 3. It will be under provisioning state until the node is logically mapped to the Bare Metal Host.



M fs-ocp1-wqpbg-master-0	N master1.fs-ocp1.flashstack.local	✓ Provisioned
M fs-ocp1-wqpbg-master-1	N master2.fs-ocp1.flashstack.local	✓ Provisioned
M fs-ocp1-wqpbg-master-2	N master3.fs-ocp1.flashstack.local	✓ Provisioned
M fs-ocp1-wqpbg-worker-0-2x7fn	N worker1.fs-ocp1.flashstack.local	✓ Provisioned
M fs-ocp1-wqpbg-worker-0-bq6f7	N worker2.fs-ocp1.flashstack.local	✓ Provisioned
M fs-ocp1-wqpbg-worker-0-lct6q	-	Provisioning

Procedure 2. Link the Machine and Bare Metal Host, Node and Bare Metal Host

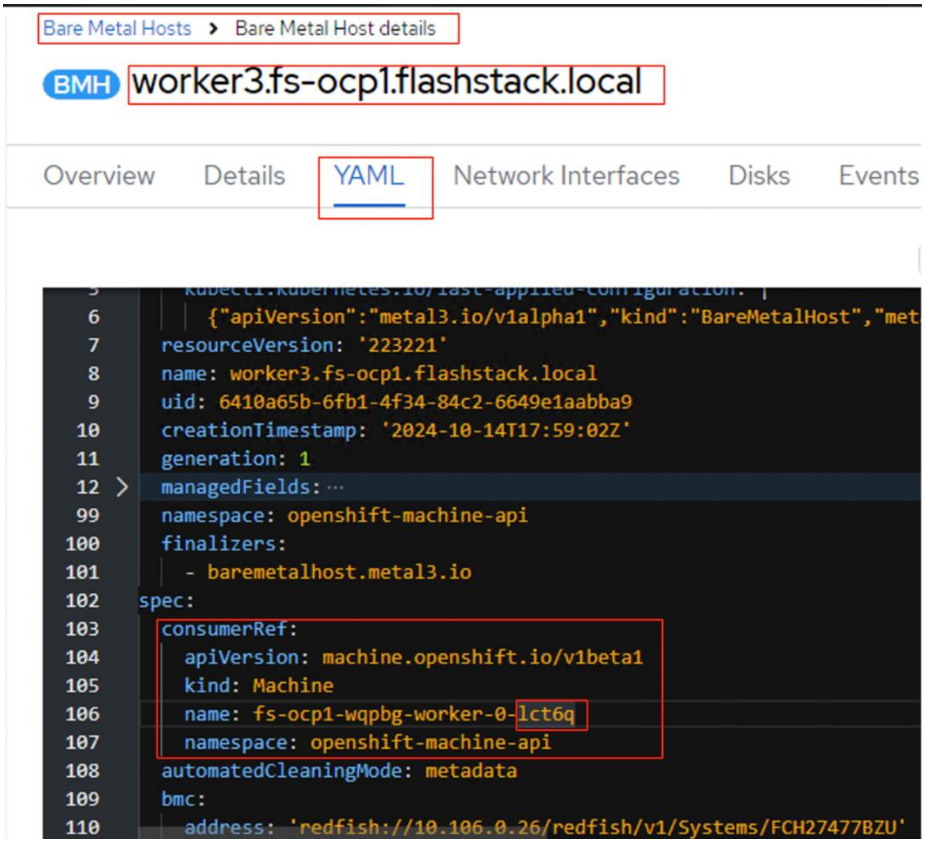
Step 1. To logically link Bare Metal Host to Machine, gather the name of the newly created machine from its manifest file or by executing **oc get machine -n openshift-machine-api**.



```
[root@aa06-rhel9 expandclus]# oc get machine -n openshift-machine-api
NAME                            PHASE     TYPE   REGION   ZONE   AGE
fs-ocp1-wqpbg-master-0          Running                          4d14h
fs-ocp1-wqpbg-master-1          Running                          4d14h
fs-ocp1-wqpbg-master-2          Running                          4d14h
fs-ocp1-wqpbg-worker-0-2x7fn    Running                          4d14h
fs-ocp1-wqpbg-worker-0-bq6f7    Running                          4d14h
fs-ocp1-wqpbg-worker-0-lct6q    Running                          4d14h
[root@aa06-rhel9 expandclus]#
```

Step 2. Update the machine name in the Bare Metal Host's manifest file under **spec. consumerRef** as shown below. Save the Yaml and reload.

```
consumerRef:
    apiVersion: machine.openshift.io/v1beta1
    kind: Machine
    name: fs-ocp1-wqpbg-worker-0-lct6q
    namespace: openshift-machine-api
```

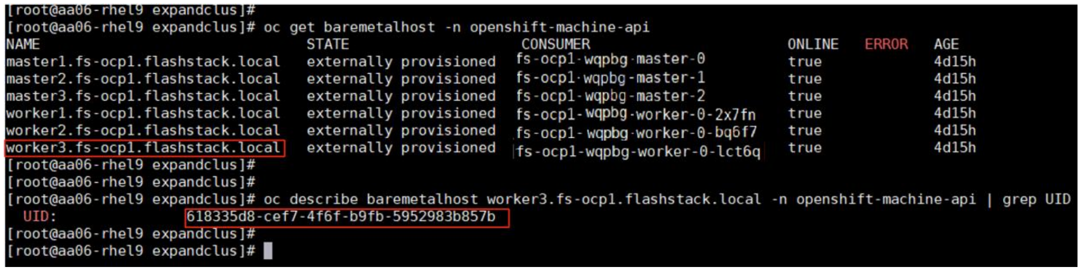

Bare Metal Hosts > Bare Metal Host details
BMH worker3.fs-ocp1.flashstack.local
Overview
Details
YAML
Network Interfaces
Disks
Events

```
    kubectl.kubernetes.io/last-applied-configuration: |
      {"apiVersion":"metal3.io/v1alpha1","kind":"BareMetalHost","met
  resourceVersion: '223221'
  name: worker3.fs-ocp1.flashstack.local
  uid: 6410a65b-6fb1-4f34-84c2-6649e1aabba9
  creationTimestamp: '2024-10-14T17:59:02Z'
  generation: 1
  managedFields: ···
  namespace: openshift-machine-api
  finalizers:
    - baremetalhost.metal3.io
spec:
  consumerRef:
    apiVersion: machine.openshift.io/v1beta1
    kind: Machine
    name: fs-ocp1-wqpbg-worker-0-lct6q
    namespace: openshift-machine-api
  automatedCleaningMode: metadata
  bmc:
    address: 'redfish://10.106.0.26/redfish/v1/Systems/FCH27477BZU'
```

After updating the machine name under Bare Metal Host yaml manifest, the newly created machine will turn to **Provisioned as node** state from **Provisioning** state.

Note: The Bare Metal Host ProviderID needs to be generated and updated in the newly added worker (worker3.fs-ocp1.flashstack.local).

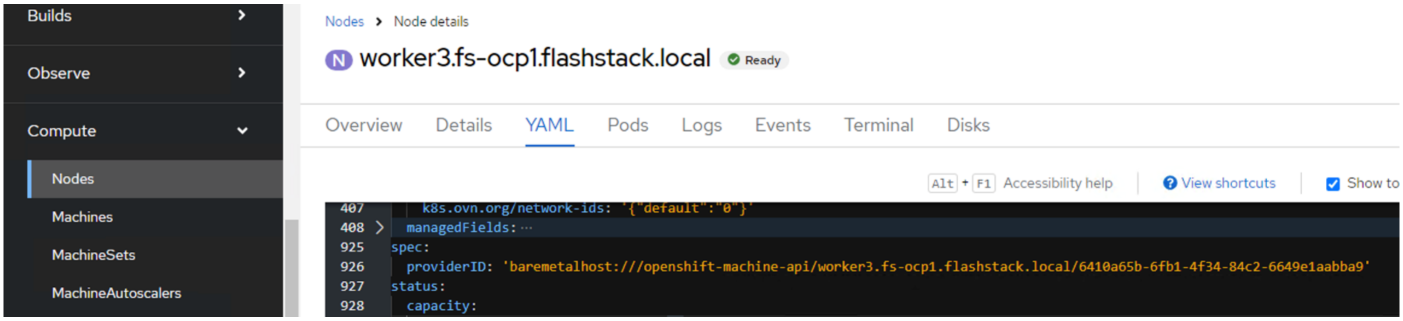
The ProviderID is a combination of the name and UUID of the Bare Metal Host and is shown below. These details can be gathered by running providerID: 'baremetalhost:///openshift-machine-api/<**Bare Metal Host Name**>/<**Bare Metal Host UID**>'



```
[root@aa06-rhel9 expandclus]#
[root@aa06-rhel9 expandclus]# oc get baremetalhost -n openshift-machine-api
NAME                              STATE                    CONSUMER                        ONLINE   ERROR   AGE
master1.fs-ocp1.flashstack.local  externally provisioned   fs-ocp1-wqpbg-master-0          true             4d15h
master2.fs-ocp1.flashstack.local  externally provisioned   fs-ocp1-wqpbg-master-1          true             4d15h
master3.fs-ocp1.flashstack.local  externally provisioned   fs-ocp1-wqpbg-master-2          true             4d15h
worker1.fs-ocp1.flashstack.local  externally provisioned   fs-ocp1-wqpbg-worker-0-2x7fn    true             4d15h
worker2.fs-ocp1.flashstack.local  externally provisioned   fs-ocp1-wqpbg-worker-0-bq6f7    true             4d15h
worker3.fs-ocp1.flashstack.local  externally provisioned   fs-ocp1-wqpbg-worker-0-lct6q    true             4d15h
[root@aa06-rhel9 expandclus]#
[root@aa06-rhel9 expandclus]#
[root@aa06-rhel9 expandclus]# oc describe baremetalhost worker3.fs-ocp1.flashstack.local -n openshift-machine-api | grep UID
  UID:                   618335d8-cef7-4f6f-b9fb-5952983b857b
[root@aa06-rhel9 expandclus]#
[root@aa06-rhel9 expandclus]# 
```

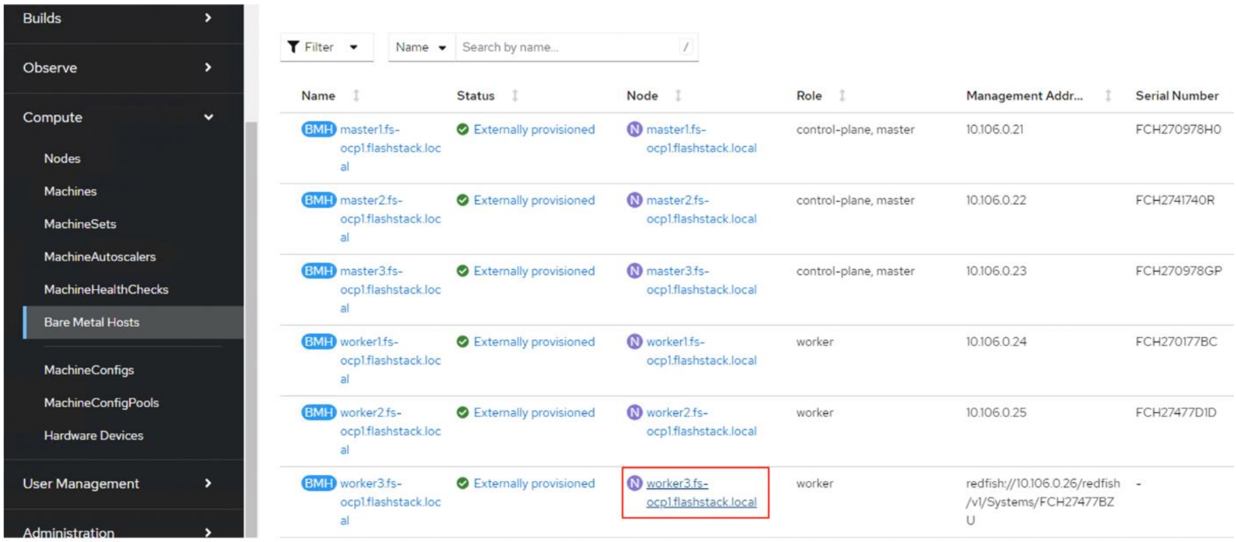
By using the provided information, the providerID for of the newly added Bare Metal Host is built as providerID: 'baremetalhost:///openshift-machine-api/worker3.fs-ocp1.flashstack.local/6410a65b-6fb1-4f34-84c2-6649e1aabba9'.

Step 3. Copy the providerID of the Bare Metal Host into the third node yaml manifest file under **spec.** as shown below.



```
    k8s.ovn.org/network-ids: '{"default":"0"}'
  managedFields: ···
spec:
  providerID: 'baremetalhost:///openshift-machine-api/worker3.fs-ocp1.flashstack.local/6410a65b-6fb1-4f34-84c2-6649e1aabba9'
status:
  capacity:
```

When the providerID of worker3 is updated, the node details are automatically populated for the newly added Bare Metal Host as shown below.



Name	Status	Node	Role	Management Addr...	Serial Number
BMH master1.fs-ocp1.flashstack.local	Externally provisioned	N master1.fs-ocp1.flashstack.local	control-plane, master	10.106.0.21	FCH270978H0
BMH master2.fs-ocp1.flashstack.local	Externally provisioned	N master2.fs-ocp1.flashstack.local	control-plane, master	10.106.0.22	FCH2741740R
BMH master3.fs-ocp1.flashstack.local	Externally provisioned	N master3.fs-ocp1.flashstack.local	control-plane, master	10.106.0.23	FCH270978GP
BMH worker1.fs-ocp1.flashstack.local	Externally provisioned	N worker1.fs-ocp1.flashstack.local	worker	10.106.0.24	FCH270177BC
BMH worker2.fs-ocp1.flashstack.local	Externally provisioned	N worker2.fs-ocp1.flashstack.local	worker	10.106.0.25	FCH27477D1D
BMH worker3.fs-ocp1.flashstack.local	Externally provisioned	N worker3.fs-ocp1.flashstack.local	worker	redfish://10.106.0.26/redfish/v1/Systems/FCH27477BZU	-

Install and Configure Portworx Enterprise on OpenShift with Pure Storage FlashArray

This chapter contains the following:

- Storage Architecture
- Prerequisites
- Configure Physical Environment
- Portworx Enterprise Console Plugin for OpenShift

Portworx by Pure Storage is fully integrated with Red Hat OpenShift. Hence you can install and manage Portworx Enterprise from OpenShift web console itself. Portworx Enterprise can be installed with Pure Storage FlashArray as a cloud storage provider. This allows you to store your data on-premises with Pure Storage FlashArray while benefiting from Portworx Enterprise cloud drive features, such as:

- Automatically provisioning block volumes
- Expanding a cluster by adding new drives or expanding existing ones
- Support for PX-Backup and Autopilot

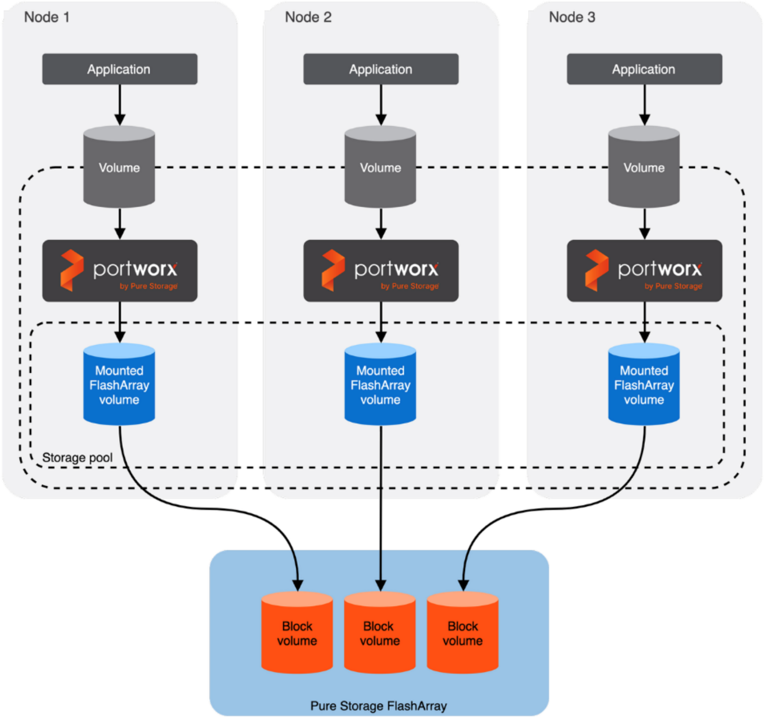
Portworx Enterprise will create and manage the underlying storage pool volumes on the registered arrays.

Note: Pure Storage recommends installing Portworx Enterprise with Pure Storage FlashArray Cloud Drives before using Pure Storage FlashArray Direct Access volumes.

Storage Architecture

This section provides the steps for installing Portworx Enterprise on OpenShift Container Platform running on Cisco UCSX-210C M7 bare metal servers. In this solution, Pure Storage FlashArray//XL170 is used as a backend storage connected over Ethernet to provide required Cloud Drives to be used by Portworx Enterprise. Figure 5 shows the high-level logical storage architecture of Portworx Enterprise deployment on Pure Storage FlashArray.

Figure 5. Portworx Enterprise deployment on OpenShift with Pure Storage FlashArray



This is the high-level summary of the Portworx Enterprise implementation of distributed storage on a typical Kubernetes based Cluster:

- Portworx Enterprise runs on each worker node as Daemonset pod and based on the configuration information provided in the StorageClass spec, Portworx Enterprise provisions one or more volumes on Pure Storage FlashArray for each worker node.
- All these Pure Storage FlashArray volumes are pooled together to form one or more Distributed Storage Pools.
- When a user creates a PVC, Portworx Enterprise provisions the volume from the storage pool.
- The PVCs consume space on the storage pool, and if space begins to run low, Portworx Enterprise can add or expand drive space from Pure Storage FlashArray.
- If a worker node goes down for less than 2 minutes, Portworx Enterprise will reattach Pure Storage FlashArray volumes when it recovers. If a node goes down for more than two minutes, a storageless node in the same zone or fault domain will take up the volumes and assume the identity of the downed storage node.

Prerequisites

These prerequisites must be met before installing the Portworx Enterprise on OpenShift with Pure Storage FlashArray:

- SecureBoot mode option must be disabled.
- The Pure Storage FlashArray should be time-synced with the same time service as the Kubernetes cluster.
- Pure Storage FlashArray must be running a minimum Purity//FA version of at least 4.8. Refer to the Supported models and versions topic for more information.

- Both multipath and iSCSI, if being used, should have their services enabled in systemd so that they start after reboots. These services are already enabled in systemd within the Red Hat CoreOS Linux.

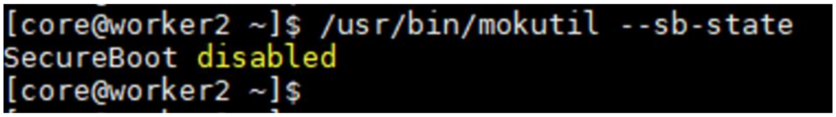
Configure Physical Environment

Before you install Portworx Enterprise, ensure that your physical network is configured appropriately and that you meet the prerequisites. You must provide Portworx Enterprise with your Pure Storage FlashArray configuration details during installation.

- Each Pure Storage FlashArray management IP address can be accessed by each node.
- Your cluster contains an up-and-running Pure Storage FlashArray with an existing dataplane connectivity layout (iSCSI, Fibre Channel).
- If you're using iSCSI, the storage node iSCSI initiators are on the same VLAN as the Pure Storage FlashArray iSCSI target ports.
- You have an API token for a user on your Pure Storage FlashArray with at least storage_admin permissions.

Procedure 1. Prepare for the Portworx Enterprise Deployment

Step 1. Secure the boot option already disabled at Intersight Boot Policy level. To confirm again at the OS level, ssh into any of the worker node from the installer VM and ensure that SecureBoot mode is disabled at the OS level.



```
[core@worker2 ~]$ /usr/bin/mokutil --sb-state
SecureBoot disabled
[core@worker2 ~]$
```

Step 2. Apply the following MachineConfig to the cluster configures each worker node with the following:

- Enable and start the multipathd.service with the specified multipath.conf configuration file.
- Enable and start the iscsid.service service
- Applies the Queue Settings with Udev rules.

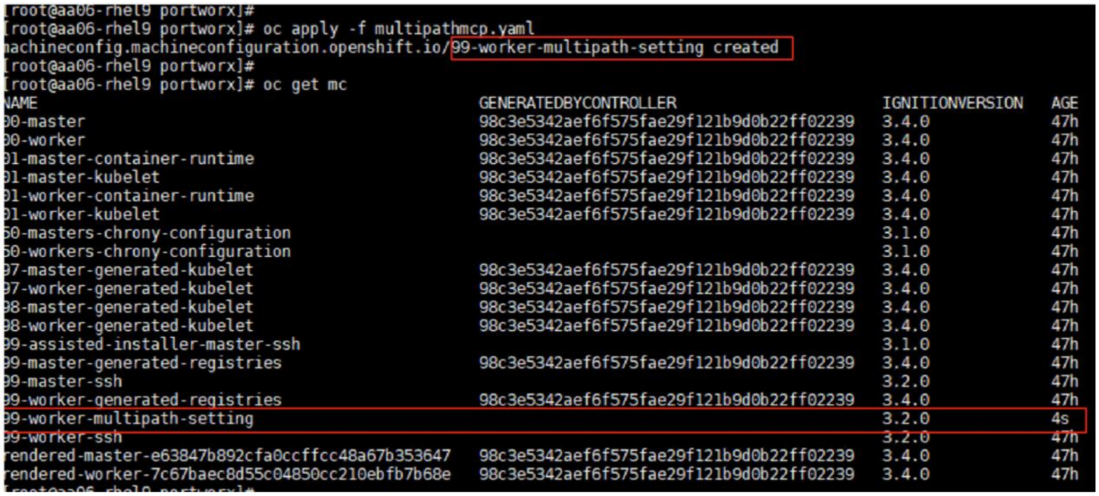
The settings of multipath and Udev rules are defined as shown below:

```
cat multipath.conf
blacklist {
      devnode "^pxd[0-9]*"
      devnode "^pxd*"
      device {
        vendor "VMware"
        product "Virtual disk"
      }
}
defaults {
 user_friendly_names no
 find_multipaths yes
 polling_interval  10
}
devices {
    device {
        vendor                   "PURE"
        product                  "FlashArray"
        path_selector            "service-time 0"
        hardware_handler         "1 alua"
        path_grouping_policy     group_by_prio
        prio                     alua
        failback                 immediate
        path_checker             tur
        fast_io_fail_tmo         10
        user_friendly_names      no
        no_path_retry            0
        features                 0
        dev_loss_tmo             600
    }
}

cat udevrules.txt
# Recommended settings for Pure Storage FlashArray.
# Use none scheduler for high-performance solid-state storage for SCSI devices
ACTION=="add|change", KERNEL=="sd*[!0-9]", SUBSYSTEM=="block", ENV{ID_VENDOR}=="PURE",
ATTR{queue/scheduler}="none"
ACTION=="add|change", KERNEL=="dm-[0-9]*", SUBSYSTEM=="block", ENV{DM_NAME}=="3624a937*",
ATTR{queue/scheduler}="none"

# Reduce CPU overhead due to entropy collection
ACTION=="add|change", KERNEL=="sd*[!0-9]", SUBSYSTEM=="block", ENV{ID_VENDOR}=="PURE",
ATTR{queue/add_random}="0"
ACTION=="add|change", KERNEL=="dm-[0-9]*", SUBSYSTEM=="block", ENV{DM_NAME}=="3624a937*",
ATTR{queue/add_random}="0"

# Spread CPU load by redirecting completions to originating CPU
ACTION=="add|change", KERNEL=="sd*[!0-9]", SUBSYSTEM=="block", ENV{ID_VENDOR}=="PURE",
ATTR{queue/rq_affinity}="2"
ACTION=="add|change", KERNEL=="dm-[0-9]*", SUBSYSTEM=="block", ENV{DM_NAME}=="3624a937*",
ATTR{queue/rq_affinity}="2"

# Set the HBA timeout to 60 seconds
ACTION=="add|change", KERNEL=="sd*[!0-9]", SUBSYSTEM=="block", ENV{ID_VENDOR}=="PURE",
ATTR{device/timeout}="60"
```

The following is the MachineConfig file that takes base64 encode results of above two files and copy them to the corresponding directory on each worker node. It also enables and starts iscsid and multipathd services:

```
cat multipathmcp.yaml
apiVersion: machineconfiguration.openshift.io/v1
kind: MachineConfig
metadata:
  creationTimestamp:
  labels:
    machineconfiguration.openshift.io/role: worker
  name: 99-worker-multipath-setting
spec:
  config:
    ignition:
      version: 3.2.0
    storage:
      files:
      - contents:
          source: data:text/plain;charset=utf-
8;base64,YmxhY2tsaXN0IHsKICAgICAgZGV2bm9kZSAiXnB4ZFswLTldKiIKICAgICAgZGV2bm9kZSAiXnB4ZCoiCiAgICAgIGRldmljZSB7
CiAgICAgICAgdmVuZG9yICJWTXdhcmUiCiAgICAgICAgcHJvZHVjdCAiVmlydHVhbCBkaXNrIgogICAgICB9Cn0KZGVmYXVsdHMgewogdXNlc
l9mcmllbmRseV9uYW1lcyBubwogZmluZF9tdWx0aXBhdGhzIHllcwogcG9sbGluZ19pbnRlcnZhbCAgMTAKfQpkZXZpY2VzIHsKICAgIGRldm
ljZSB7CiAgICAgICAgdmVuZG9yICAgICAgICAgICAgICAgICAgICJQVVJFIgogICAgICAgIHByb2R1Y3QgICAgICAgICAgICAgICAgICAiRmx
hc2hBcnJheSIKICAgICAgICBwYXRoX3NlbGVjdG9yICAgICAgICAgICAgInNlcnZpY2UtdGltZSAwIgogICAgICAgIGhhcmR3YXJlX2hhbmRs
ZXIgICAgICAgICAiMSBhbHVhIgogICAgICAgIHBhdGhfZ3JvdXBpbmdfcG9saWN5ICAgICBncm91cF9ieV9wcmlvCiAgICAgICAgcHJpbyAgI
CAgICAgICAgICAgICAgICAgIGFsdWEKICAgICAgICBmYWlsYmFjayAgICAgICAgICAgICAgICAgaW1tZWRpYXRlCiAgICAgICAgcGF0aF9jaG
Vja2VyICAgICAgICAgICAgIHR1cgogICAgICAgIGZhc3RfaW9fZmFpbF90bW8gICAgICAxMAogICAgICAgIHVzZXJfZnJpZW5kbHlfbmF
tZXMgICAgICBubwogICAgICAgIG5vX3BhdGhfcmV0cnkgICAgICAgICAgICAwCiAgICAgICAgZmVhdHVyZXMgICAgICAgICAgICAgICAgIDAK
ICAgICAgICBkZXZfbG9zc190bW8gICAgICAgICAgICAgNjAwCiAgICB9Cn0K
        filesystem: root
        mode: 0644
        overwrite: true
        path: /etc/multipath.conf
      - contents:
          source: data:text/plain;charset=utf-
8;base64,IyBSZWNvbW1lbmRlZCBzZXR0aW5ncyBmb3IgUHVyZSBTdG9yYWdlIEZsYXNoQXJyYXkuCiMgVXNlIG5vbmUgc2NoZWR1bGVyIGZv
ciBoaWdoLXBlcmZvcm1hbmNlIHNvbGlkLXN0YXRlIHN0b3JhZ2UgZm9yIFNDU0kgZGV2aWNlcwpBQ1RJT049PSJhZGR8Y2hhbmdlIiwgS0VST
kVMPT0ic2QqWyEwLTldIiwgU1VCU1lTVEVNPT0iYmxvY2siLCBFTlZ7SURfVkVORE9SfT09IlBVUkUiLCBBVFRSe3F1ZXVlL3NjaGVkdWxlcn
09Im5vbmUiCkFDVElPTj09ImFkZHxjaGFuZ2UiLCBLRVJORUw9PSJkbS1bMC05XSoiLCBTVUJTWVNURU09PSJibG9jayIsIEVOVntETV9OQU1
FfT09IjM2MjRhOTM3KiIsIEFUVFJ7cXVldWUvc2NoZWR1bGVyfT0ibm9uZSIKCiMgUmVkdWNlIENQVSBvdmVyaGVhZCBkdWUgdG8gZW50cm9w
eSBjb2xsZWN0aW9uCkFDVElPTj09ImFkZHxjaGFuZ2UiLCBLRVJORUw9PSJzZCpbITAtOV0iLCBTVUJTWVNURU09PSJibG9jayIsIEVOVntJR
F9WRU5ET1J9PT0iUFVSRSIsIEFUVFJ7cXVldWUvYWRkX3JhbmRvbX09IjAiCkFDVElPTj09ImFkZHxjaGFuZ2UiLCBLRVJORUw9PSJkbS1bMC
```

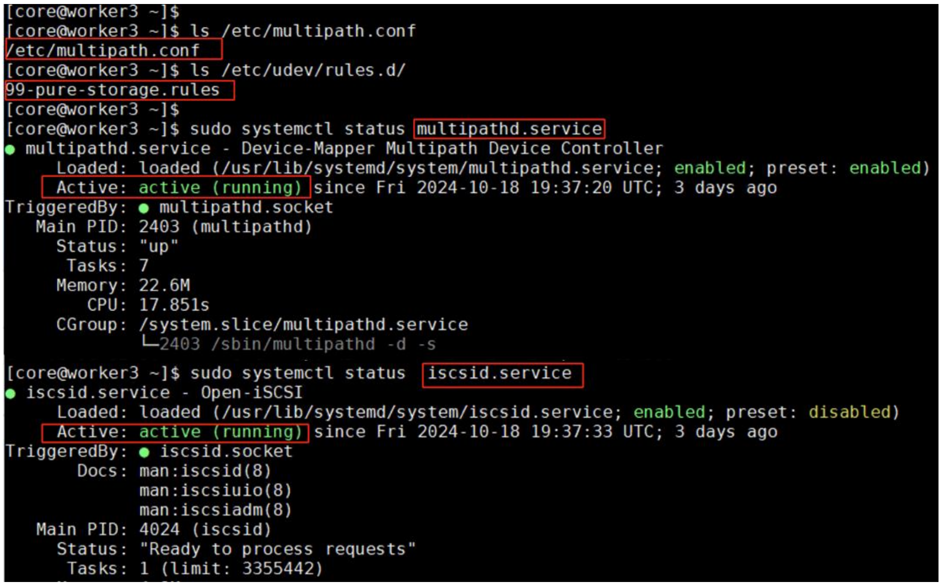
```
05XSoiLCBTVUJTWVNURU09PSJibG9jayIsIEVOVntETV9OQU1FfT09IjM2MjRhOTM3KiIsIEFUVFJ7cXVldWUvYWRkX3JhbmRvbX09IjAiCgo
jIFNwcmVhZCBDUFUgbG9hZCBieSByZWRpcmVjdGluZyBjb21wbGV0aW9ucyB0byBvcmlnaW5hdGluZyBDUFUKQUNUSU9OPT0iYWRkfGNoYW5n
ZSIsIEtFUk5FTD09InNkKlshMC05XSIsIFNVQlNZU1RFTT09ImJsb2NrIiwgRU5We0lEX1ZFTkRPUn09PSJQVVJFIiwgQVRUUntxdWV1ZS9yc
V9hZmZpbml0eX09IjIiCkFDVElPTj09ImFkZHxjaGFuZ2UiLCBLRVJORUw9PSJkbS1bMC05XSoiLCBTVUJTWVNURU09PSJibG9jayIsIEVOVn
tETV9OQU1FfT09IjM2MjRhOTM3KiIsIEFUVFJ7cXVldWUvcnFfYWZmaW5pdHl9PSIyIgoKIyBTZXQgdGhlIEhCQSB0aW1lb3V0IHRvIDYwIHN
lY29uZHMKQUNUSU9OPT0iYWRkfGNoYW5nZSIsIEtFUk5FTD09InNkKlshMC05XSIsIFNVQlNZU1RFTT09ImJsb2NrIiwgRU5We0lEX1ZFTkRP
Un09PSJQVVJFIiwgQVRUUntkZXZpY2UvdGltZW91dH09IjYwIgo=
        filesystem: root
        mode: 0644
        overwrite: true
        path: /etc/udev/rules.d/99-pure-storage.rules
    systemd:
      units:
      - enabled: true
        name: iscsid.service
      - enabled: true
        name: multipathd.service
```



```
[root@aa06-rhel9 portworx]#
[root@aa06-rhel9 portworx]# oc apply -f multipathmcp.yaml
achineconfig.machineconfiguration.openshift.io/99-worker-multipath-setting created
[root@aa06-rhel9 portworx]#
[root@aa06-rhel9 portworx]# oc get mc
NAME                                               GENERATEDBYCONTROLLER                      IGNITIONVERSION   AGE
00-master                                          98c3e5342aef6f575fae29f121b9d0b22ff02239   3.4.0             47h
00-worker                                          98c3e5342aef6f575fae29f121b9d0b22ff02239   3.4.0             47h
01-master-container-runtime                        98c3e5342aef6f575fae29f121b9d0b22ff02239   3.4.0             47h
01-master-kubelet                                  98c3e5342aef6f575fae29f121b9d0b22ff02239   3.4.0             47h
01-worker-container-runtime                        98c3e5342aef6f575fae29f121b9d0b22ff02239   3.4.0             47h
01-worker-kubelet                                  98c3e5342aef6f575fae29f121b9d0b22ff02239   3.4.0             47h
50-masters-chrony-configuration                                                               3.1.0             47h
50-workers-chrony-configuration                                                               3.1.0             47h
97-master-generated-kubelet                        98c3e5342aef6f575fae29f121b9d0b22ff02239   3.4.0             47h
97-worker-generated-kubelet                        98c3e5342aef6f575fae29f121b9d0b22ff02239   3.4.0             47h
98-master-generated-kubelet                        98c3e5342aef6f575fae29f121b9d0b22ff02239   3.4.0             47h
98-worker-generated-kubelet                        98c3e5342aef6f575fae29f121b9d0b22ff02239   3.4.0             47h
99-assisted-installer-master-ssh                                                              3.1.0             47h
99-master-generated-registries                     98c3e5342aef6f575fae29f121b9d0b22ff02239   3.4.0             47h
99-master-ssh                                                                                 3.2.0             47h
99-worker-generated-registries                     98c3e5342aef6f575fae29f121b9d0b22ff02239   3.4.0             47h
99-worker-multipath-setting                                                                   3.2.0             4s
99-worker-ssh                                                                                 3.2.0             47h
rendered-master-e63847b892cfa0ccffcc48a67b353647   98c3e5342aef6f575fae29f121b9d0b22ff02239   3.4.0             47h
rendered-worker-7c67baec8d55c04850cc210ebfb7b68e   98c3e5342aef6f575fae29f121b9d0b22ff02239   3.4.0             47h
[root@aa06-rhel9 portworx]#
```

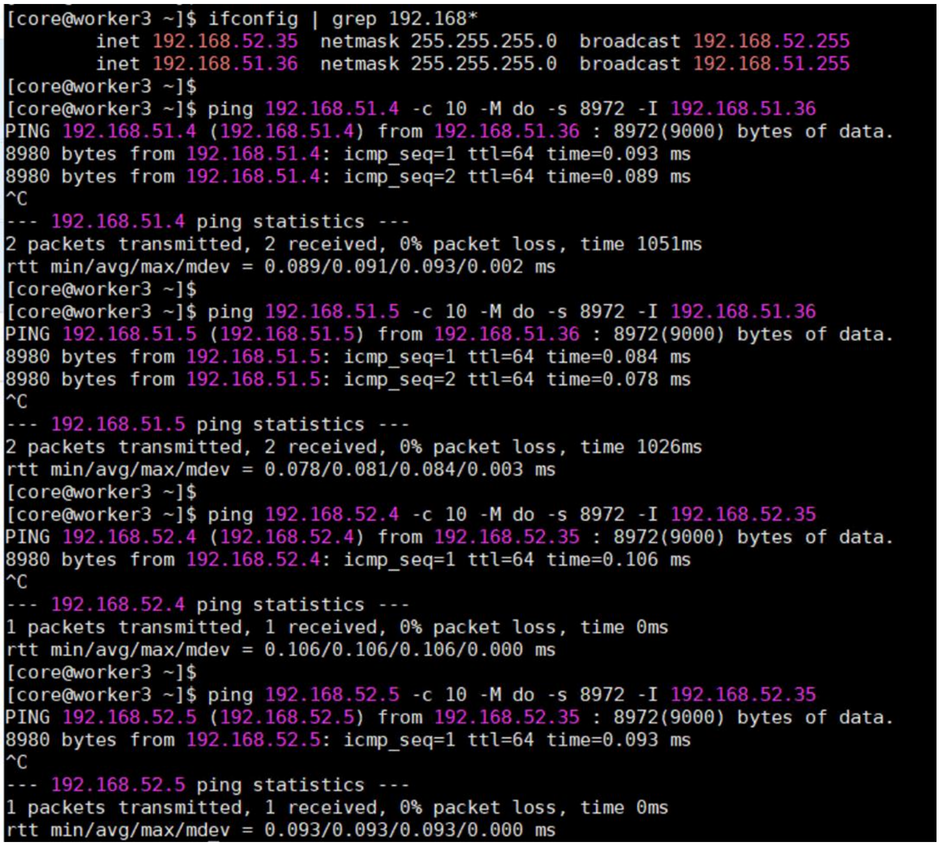
Step 3. This machine config is applied to each worker node one by one. To see the status of this process, go to **Administration -> Cluster Settings** from the cluster console.

Step 4. After the MachineConfig is being applied on all the worker nodes, ssh into one of the worker nodes and verify.



```
[core@worker3 ~]$
[core@worker3 ~]$ ls /etc/multipath.conf
/etc/multipath.conf
[core@worker3 ~]$ ls /etc/udev/rules.d/
99-pure-storage.rules
[core@worker3 ~]$
[core@worker3 ~]$ sudo systemctl status multipathd.service
● multipathd.service - Device-Mapper Multipath Device Controller
     Loaded: loaded (/usr/lib/systemd/system/multipathd.service; enabled; preset: enabled)
     Active: active (running) since Fri 2024-10-18 19:37:20 UTC; 3 days ago
TriggeredBy: ● multipathd.socket
   Main PID: 2403 (multipathd)
     Status: "up"
      Tasks: 7
     Memory: 22.6M
        CPU: 17.851s
     CGroup: /system.slice/multipathd.service
             └─2403 /sbin/multipathd -d -s

[core@worker3 ~]$ sudo systemctl status  iscsid.service
● iscsid.service - Open-iSCSI
     Loaded: loaded (/usr/lib/systemd/system/iscsid.service; enabled; preset: disabled)
     Active: active (running) since Fri 2024-10-18 19:37:33 UTC; 3 days ago
TriggeredBy: ● iscsid.socket
       Docs: man:iscsid(8)
             man:iscsiuio(8)
             man:iscsiadm(8)
   Main PID: 4024 (iscsid)
     Status: "Ready to process requests"
      Tasks: 1 (limit: 3355442)
```

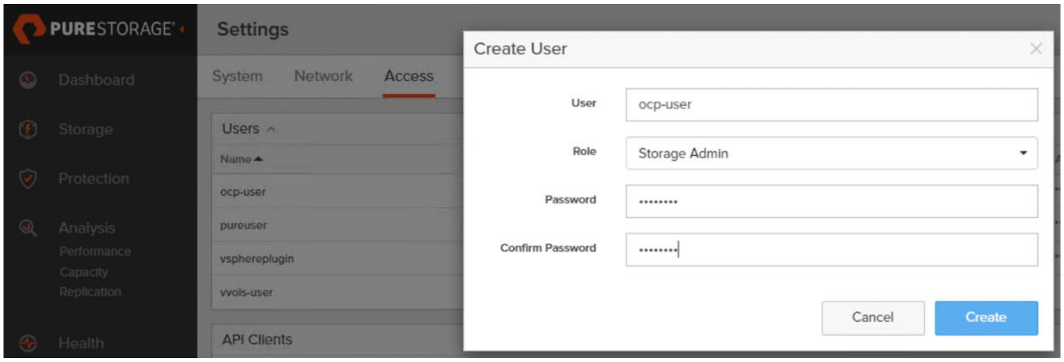
Step 5. From each worker node, ensure the Pure Storage FlashArray storage IPs are reachable with large packet size (8972) and without fragmenting the packets:



```
[core@worker3 ~]$ ifconfig | grep 192.168*
        inet 192.168.52.35  netmask 255.255.255.0  broadcast 192.168.52.255
        inet 192.168.51.36  netmask 255.255.255.0  broadcast 192.168.51.255
[core@worker3 ~]$
[core@worker3 ~]$ ping 192.168.51.4 -c 10 -M do -s 8972 -I 192.168.51.36
PING 192.168.51.4 (192.168.51.4) from 192.168.51.36 : 8972(9000) bytes of data.
8980 bytes from 192.168.51.4: icmp_seq=1 ttl=64 time=0.093 ms
8980 bytes from 192.168.51.4: icmp_seq=2 ttl=64 time=0.089 ms
^C
--- 192.168.51.4 ping statistics ---
2 packets transmitted, 2 received, 0% packet loss, time 1051ms
rtt min/avg/max/mdev = 0.089/0.091/0.093/0.002 ms
[core@worker3 ~]$
[core@worker3 ~]$ ping 192.168.51.5 -c 10 -M do -s 8972 -I 192.168.51.36
PING 192.168.51.5 (192.168.51.5) from 192.168.51.36 : 8972(9000) bytes of data.
8980 bytes from 192.168.51.5: icmp_seq=1 ttl=64 time=0.084 ms
8980 bytes from 192.168.51.5: icmp_seq=2 ttl=64 time=0.078 ms
^C
--- 192.168.51.5 ping statistics ---
2 packets transmitted, 2 received, 0% packet loss, time 1026ms
rtt min/avg/max/mdev = 0.078/0.081/0.084/0.003 ms
[core@worker3 ~]$
[core@worker3 ~]$ ping 192.168.52.4 -c 10 -M do -s 8972 -I 192.168.52.35
PING 192.168.52.4 (192.168.52.4) from 192.168.52.35 : 8972(9000) bytes of data.
8980 bytes from 192.168.52.4: icmp_seq=1 ttl=64 time=0.106 ms
^C
--- 192.168.52.4 ping statistics ---
1 packets transmitted, 1 received, 0% packet loss, time 0ms
rtt min/avg/max/mdev = 0.106/0.106/0.106/0.000 ms
[core@worker3 ~]$
[core@worker3 ~]$ ping 192.168.52.5 -c 10 -M do -s 8972 -I 192.168.52.35
PING 192.168.52.5 (192.168.52.5) from 192.168.52.35 : 8972(9000) bytes of data.
8980 bytes from 192.168.52.5: icmp_seq=1 ttl=64 time=0.093 ms
^C
--- 192.168.52.5 ping statistics ---
1 packets transmitted, 1 received, 0% packet loss, time 0ms
rtt min/avg/max/mdev = 0.093/0.093/0.093/0.000 ms
```

Step 6. Create Kubernetes secret object constituting the Pure Storage FlashArray API endpoints and API Tokens that Portworx Enterprise needs to communicate with and manage the Pure Storage FlashArray storage device.

Step 7. Log into Pure Storage FlashArray and go to **Settings > Users and Polices**. Create a dedicated user (for instance ocp-user) with storage Admin role for Portworx Enterprise authentication.



Create User

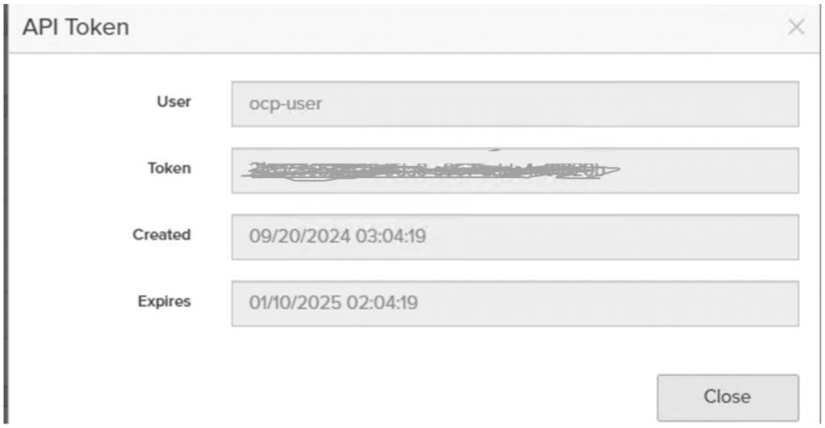
User: ocp-user

Role: Storage Admin

Password: ••••••••

Confirm Password: ••••••••

Step 8. Click the ellipses of the user previously create and select **Create API Token.** On the **Create API Token** wizard, set number of weeks (for instance 24) for the API key to expire and click **Create**. A new API Token for the ocp-user will be created and shown to you. Copy the API key and preserve it as it will be used later to create Kubernetes Secret.



API Token

User: ocp-user

Token: [redacted]

Created: 09/20/2024 03:04:19

Expires: 01/10/2025 02:04:19

Step 9. Create Kubernetes secret with the above API key using the following manifest:

```
## Create a json file constituting FlashArray Management IP addresses and API key created in the above step.

cat pure.json
{
  "FlashArrays": [
    {
      "MgmtEndPoint": "10.103.0.55",
      "APIToken": "< your API KEY >"
    }
  ]
}

## secret name must match with below name
kubectl create secret generic px-pure-secret --namespace px --from-file=pure.json
```

Note: If multiple arrays configured with Availability Zones labels (AZs) are available, then you can use these AZ topology labels and enter those into pure.json to distinguish the arrays. For more details, go to: https://docs.portworx.com/portworx-enterprise/operations/operate-kubernetes/cluster-topology/csi-topology

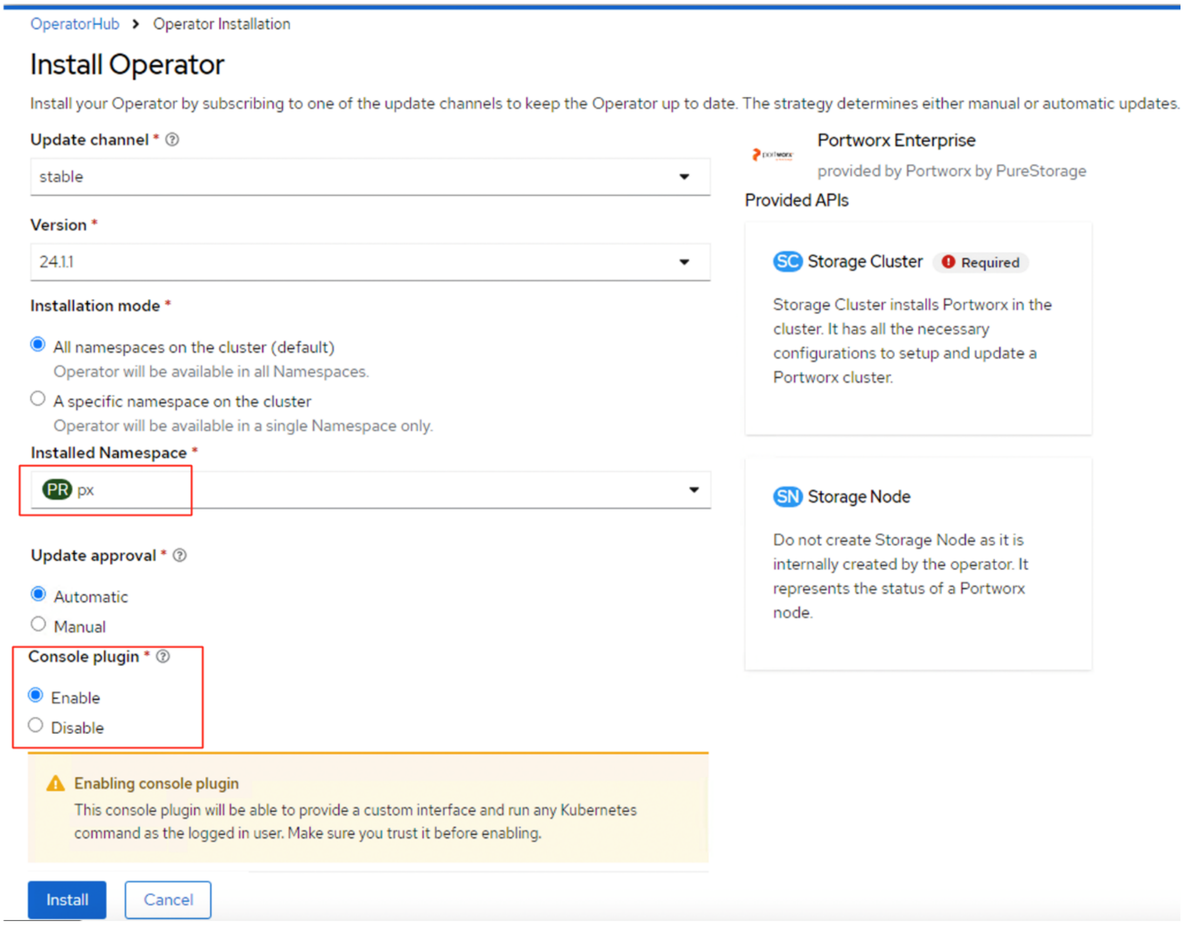
Procedure 2. Deploy Portworx Enterprise

Step 1. Log into the OpenShift cluster console using kubeadmin account and navigate to **Operators > OperatorHub**.

Step 2. On the right-side pane, enter Portworx Enterprise to filter the available operators in the Operator Hub. Select **Portworx Enterprise** and click **Install.**

Step 3. On the Operator Installation screen, under **Installed Namespace** drop down list, select **Create Project** and create a new project (for instance **px**) and select the newly created project to install the Portworx Operator.

Step 4. Install the **Portworx plugin** for OpenShift by clicking **Enable** under the Console Plugin**.** 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 *: stable

Version *: 24.1.1

Installation mode *

● All namespaces on the cluster (default)
Operator will be available in all Namespaces.

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

Installed Namespace *: PR px

Update approval *

● Automatic

○ Manual

Console plugin *

● Enable

○ Disable

Enabling console plugin
This console plugin will be able to provide a custom interface and run any Kubernetes command as the logged in user. Make sure you trust it before enabling.

Portworx Enterprise
provided by Portworx by PureStorage

Provided APIs

SC Storage Cluster — Required
Storage Cluster installs Portworx in the cluster. It has all the necessary configurations to setup and update a Portworx cluster.

SN Storage Node
Do not create Storage Node as it is internally created by the operator. It represents the status of a Portworx node.

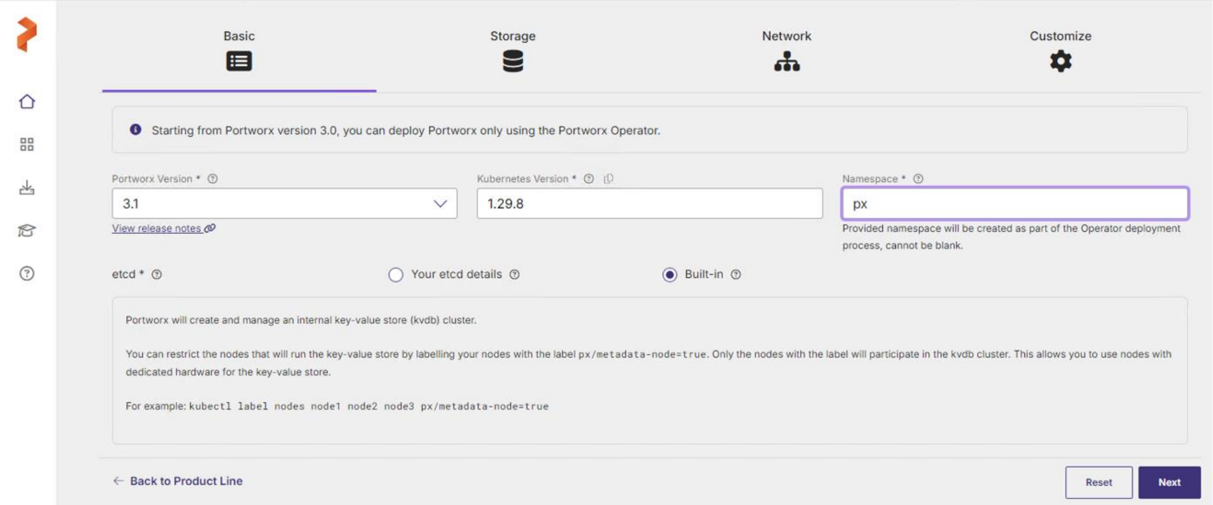
Note: Portworx Console Plugin for OpenShift will be activated and shown only after installing the StorageCluster. Follow the below steps to create Portworx StorageCluster.

Step 5. When the Portworx operator is successfully installed, the StorageCluster needs to be created. The StorageCluster Specifications (manifest file) can be created by logging into https://central.portworx.com/. Log into the portal using your credentials and click **Get Started**.

Step 6. Select **Portworx Enterprise** and click **Continue**.

Step 7. Under the Generate Spec page, select the latest Portworx version (version 3.1 was the latest when this solution was validated). Select **Pure FlashArray** as platform. Select **OpenShift 4+** for the Distribution drop-down list, provide **px** for the Namespace field. Click the **Customize** option located at the bottom of the page.

Step 8. Get the Kubernetes version by running **kubectl version | grep -i 'Server Version'**. Click **Next** .



Starting from Portworx version 3.0, you can deploy Portworx only using the Portworx Operator.

Portworx Version *: 3.1

View release notes

Kubernetes Version *: 1.29.8

Namespace *: px

Provided namespace will be created as part of the Operator deployment process, cannot be blank.

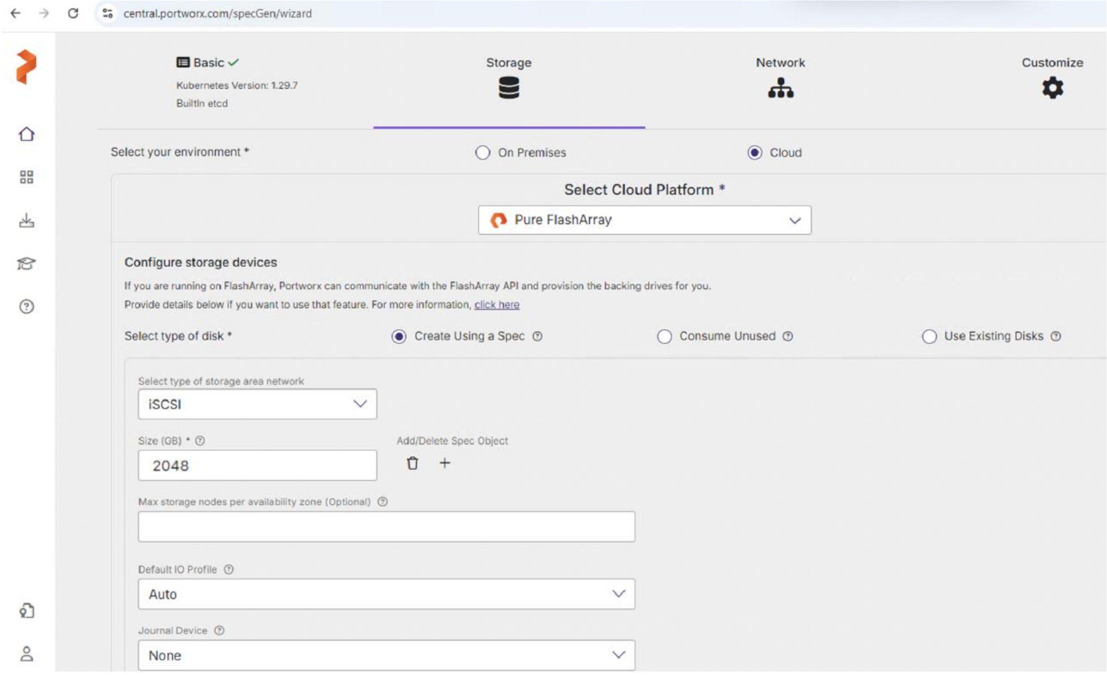
etcd * ○ Your etcd details ● Built-in

Portworx will create and manage an internal key-value store (kvdb) cluster.

You can restrict the nodes that will run the key-value store by labelling your nodes with the label `px/metadata-node=true`. Only the nodes with the label will participate in the kvdb cluster. This allows you to use nodes with dedicated hardware for the key-value store.

For example: `kubectl label nodes node1 node2 node3 px/metadata-node=true`

Step 9. From the Storage tab, select **iSCSI** for Storage Area Network. Provide the size of the Cloud drive and click plus (**+**) to add additional disks. Click **Next**.



Kubernetes Version: 1.29.7

Builtin etcd

Select your environment * ○ On Premises ● Cloud

Select Cloud Platform *: Pure FlashArray

Configure storage devices

If you are running on FlashArray, Portworx can communicate with the FlashArray API and provision the backing drives for you.

Provide details below if you want to use that feature. For more information, click here

Select type of disk * ● Create Using a Spec ○ Consume Unused ○ Use Existing Disks

Select type of storage area network: iSCSI

Size (GB) *: 2048

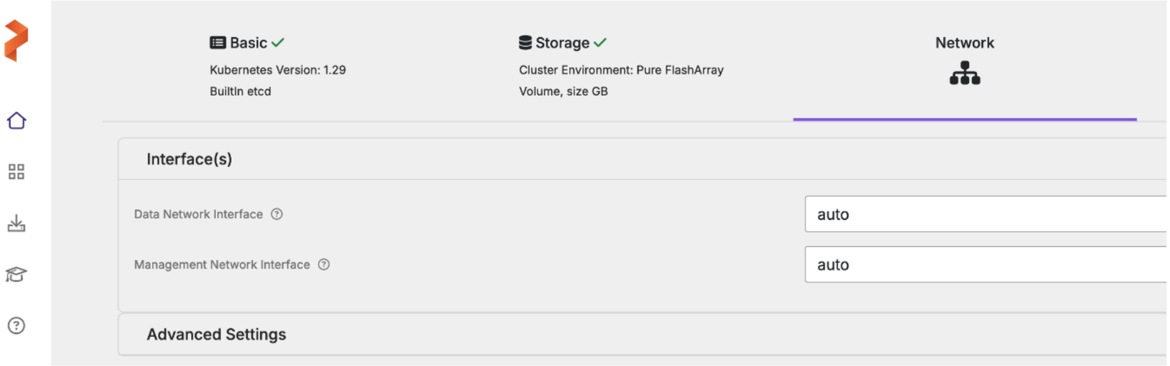
Add/Delete Spec Object

Max storage nodes per availability zone (Optional):

Default IO Profile: Auto

Journal Device: None

Step 10. On the Network tab, set **Auto** for both Data and Management Network Interfaces. Click **Next**.



Kubernetes Version: 1.29

BuiltIn etcd

Cluster Environment: Pure FlashArray

Volume, size GB

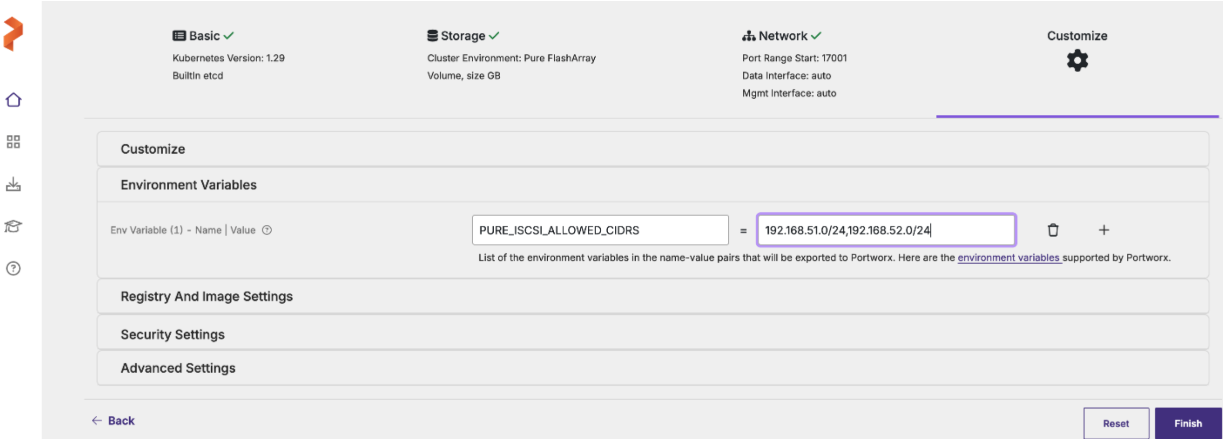
Interface(s)

Data Network Interface: auto

Management Network Interface: auto

Advanced Settings

Step 11. On the Customize tab, click **Auto** for both Data and Management Network Interfaces. Click **Next**.



Basic ✓
Kubernetes Version: 1.29
Builtin etcd

Storage ✓
Cluster Environment: Pure FlashArray
Volume, size GB

Network ✓
Port Range Start: 17001
Data Interface: auto
Mgmt Interface: auto

Customize

Customize

Environment Variables

Env Variable (1) - Name | Value: PURE_ISCSI_ALLOWED_CIDRS = 192.168.51.0/24,192.168.52.0/24

List of the environment variables in the name-value pairs that will be exported to Portworx. Here are the environment variables supported by Portworx.

Registry And Image Settings

Security Settings

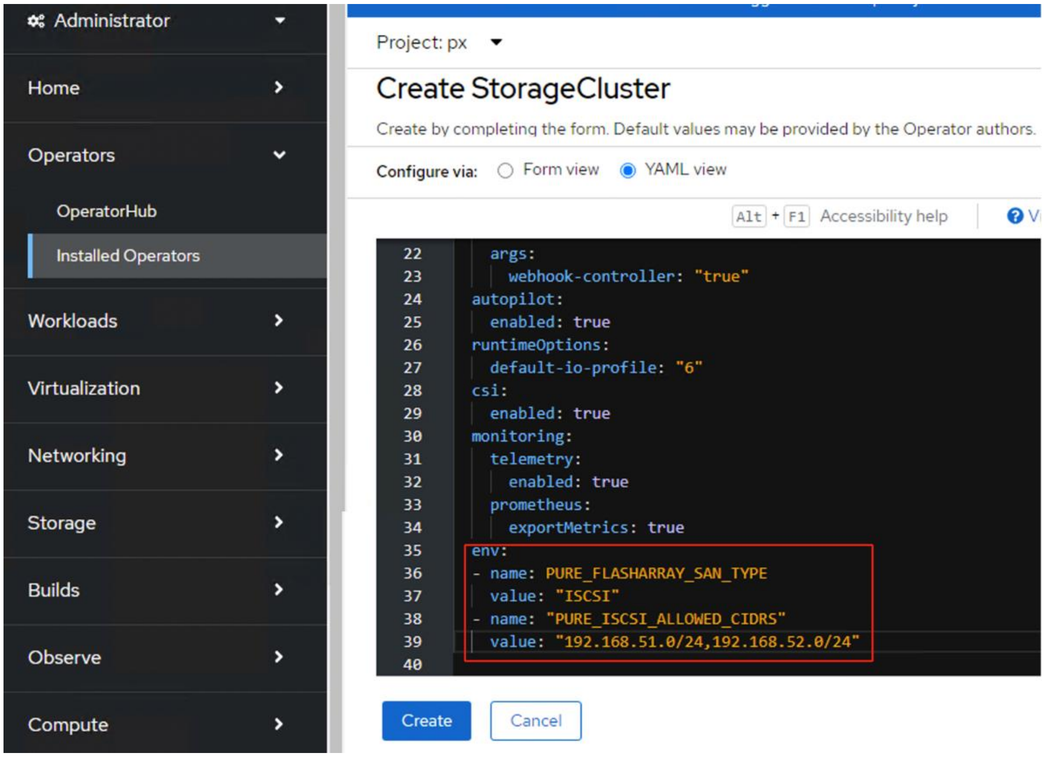
Advanced Settings

Note: Ensure to enter both the iSCSI network subnets here. This enables the iSCSI volumes on the workers nodes to leverage all the available data paths to access target volumes.

Step 12. Click **Advanced Settings**, enter the name of the portworx cluster (for instance, **ocp-pxclus**). Click **Finish**. Click **Download.yaml** to download the StorageCluster specification file.

Step 13. From the OpenShift console, go to **Operators > Installed Operators > Portworx Enterprise**. Click the **StorageCluster** tab and click **Create Storage Cluster** to create StorageCluster. this opens the YAML view of Storage Cluster.

Step 14. Copy the contents of the spec file previously downloaded and paste it in the yaml body. Verify that both the iSCSI subnet networks are listed under **env**: as shown below. Click **Create** to create the StorageCluster.



Project: px

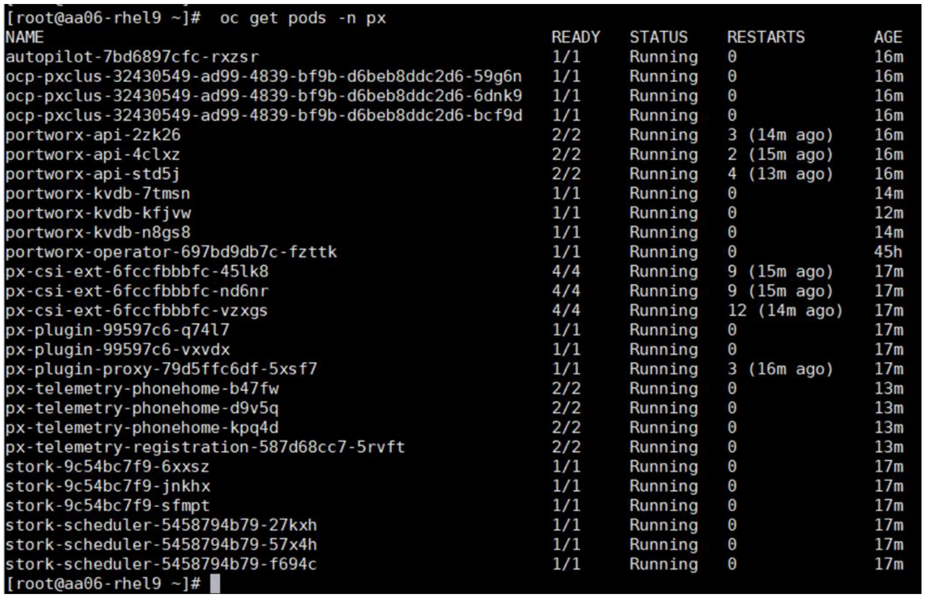
Create StorageCluster

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

Configure via: ○ Form view ● YAML view

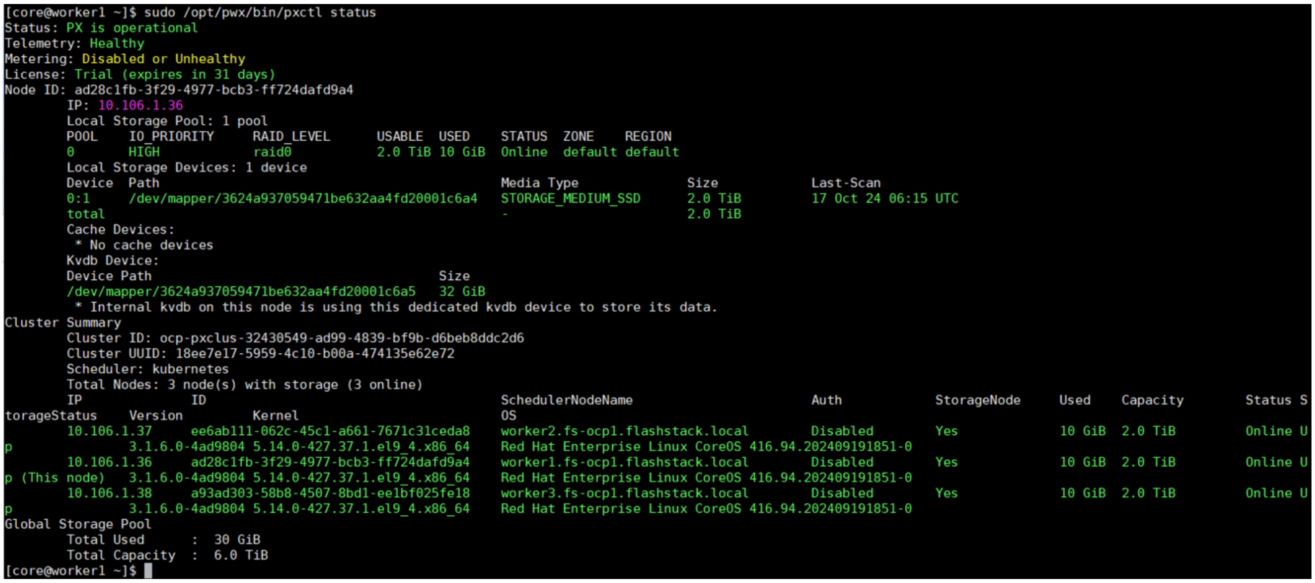
```
    args:
      webhook-controller: "true"
  autopilot:
    enabled: true
  runtimeOptions:
    default-io-profile: "6"
  csi:
    enabled: true
  monitoring:
    telemetry:
      enabled: true
    prometheus:
      exportMetrics: true
  env:
  - name: PURE_FLASHARRAY_SAN_TYPE
    value: "ISCSI"
  - name: "PURE_ISCSI_ALLOWED_CIDRS"
    value: "192.168.51.0/24,192.168.52.0/24"
```

Wait until all the Portworx related pods come online.



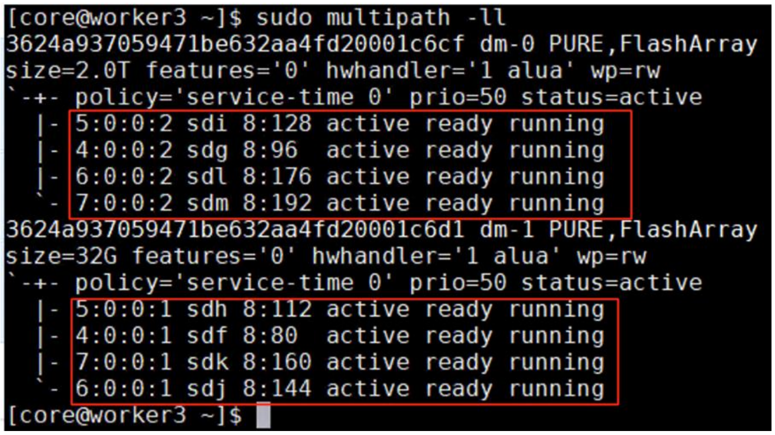
```
[root@aa06-rhel9 ~]#  oc get pods -n px
NAME                                                    READY   STATUS    RESTARTS       AGE
autopilot-7bd6897cfc-rxzsr                              1/1     Running   0              16m
ocp-pxclus-32430549-ad99-4839-bf9b-d6beb8ddc2d6-59g6n   1/1     Running   0              16m
ocp-pxclus-32430549-ad99-4839-bf9b-d6beb8ddc2d6-6dnk9   1/1     Running   0              16m
ocp-pxclus-32430549-ad99-4839-bf9b-d6beb8ddc2d6-bcf9d   1/1     Running   0              16m
portworx-api-2zk26                                      2/2     Running   3 (14m ago)    16m
portworx-api-4clxz                                      2/2     Running   2 (15m ago)    16m
portworx-api-std5j                                      2/2     Running   4 (13m ago)    16m
portworx-kvdb-7tmsn                                     1/1     Running   0              14m
portworx-kvdb-kfjvw                                     1/1     Running   0              12m
portworx-kvdb-n8gs8                                     1/1     Running   0              14m
portworx-operator-697bd9db7c-fzttk                      1/1     Running   0              45h
px-csi-ext-6fccfbbbfc-45lk8                             4/4     Running   9 (15m ago)    17m
px-csi-ext-6fccfbbbfc-nd6nr                             4/4     Running   9 (15m ago)    17m
px-csi-ext-6fccfbbbfc-vzxgs                             4/4     Running   12 (14m ago)   17m
px-plugin-99597c6-q74l7                                 1/1     Running   0              17m
px-plugin-99597c6-vxvdx                                 1/1     Running   0              17m
px-plugin-proxy-79d5ffc6df-5xsf7                        1/1     Running   3 (16m ago)    17m
px-telemetry-phonehome-b47fw                            2/2     Running   0              13m
px-telemetry-phonehome-d9v5q                            2/2     Running   0              13m
px-telemetry-phonehome-kpq4d                            2/2     Running   0              13m
px-telemetry-registration-587d68cc7-5rvft               2/2     Running   0              13m
stork-9c54bc7f9-6xxsz                                   1/1     Running   0              17m
stork-9c54bc7f9-jnkhx                                   1/1     Running   0              17m
stork-9c54bc7f9-sfmpt                                   1/1     Running   0              17m
stork-scheduler-5458794b79-27kxh                        1/1     Running   0              17m
stork-scheduler-5458794b79-57x4h                        1/1     Running   0              17m
stork-scheduler-5458794b79-f694c                        1/1     Running   0              17m
[root@aa06-rhel9 ~]#
```

Step 15. Verify the cluster status by executing the command on any worker node: sudo /opt/pwx/bin/pxctl status.



```
[core@worker1 ~]$ sudo /opt/pwx/bin/pxctl status
Status: PX is operational
Telemetry: Healthy
Metering: Disabled or Unhealthy
License: Trial (expires in 31 days)
Node ID: ad28c1fb-3f29-4977-bcb3-ff724dafd9a4
        IP: 10.106.1.36
        Local Storage Pool: 1 pool
        POOL    IO_PRIORITY     RAID_LEVEL      USABLE  USED    STATUS  ZONE    REGION
        0       HIGH            raid0           2.0 TiB 10 GiB  Online  default default
        Local Storage Devices: 1 device
        Device  Path                                            Media Type              Size            Last-Scan
        0:1     /dev/mapper/3624a937059471be632aa4fd20001c6a4   STORAGE_MEDIUM_SSD      2.0 TiB         17 Oct 24 06:15 UTC
        total                                                   -                       2.0 TiB
        Cache Devices:
         * No cache devices
        Kvdb Device:
        Device Path                                     Size
        /dev/mapper/3624a937059471be632aa4fd20001c6a5   32 GiB
         * Internal kvdb on this node is using this dedicated kvdb device to store its data.
Cluster Summary
        Cluster ID: ocp-pxclus-32430549-ad99-4839-bf9b-d6beb8ddc2d6
        Cluster UUID: 18ee7e17-5959-4c10-b00a-474135e62e72
        Scheduler: kubernetes
        Total Nodes: 3 node(s) with storage (3 online)
        IP              ID                                      SchedulerNodeName                       Auth            StorageNode     Used    Capacity        Status S
torageStatus    Version         Kernel                          OS
        10.106.1.37     ee6ab111-062c-45c1-a661-7671c31ceda8    worker2.fs-ocp1.flashstack.local        Disabled        Yes             10 GiB  2.0 TiB         Online U
p               3.1.6.0-4ad9804 5.14.0-427.37.1.el9_4.x86_64    Red Hat Enterprise Linux CoreOS 416.94.202409191851-0
        10.106.1.36     ad28c1fb-3f29-4977-bcb3-ff724dafd9a4    worker1.fs-ocp1.flashstack.local        Disabled        Yes             10 GiB  2.0 TiB         Online U
p (This node)   3.1.6.0-4ad9804 5.14.0-427.37.1.el9_4.x86_64    Red Hat Enterprise Linux CoreOS 416.94.202409191851-0
        10.106.1.38     a93ad303-58b8-4507-8bd1-ee1bf025fe18    worker3.fs-ocp1.flashstack.local        Disabled        Yes             10 GiB  2.0 TiB         Online U
p               3.1.6.0-4ad9804 5.14.0-427.37.1.el9_4.x86_64    Red Hat Enterprise Linux CoreOS 416.94.202409191851-0
Global Storage Pool
        Total Used      :  30 GiB
        Total Capacity  :  6.0 TiB
[core@worker1 ~]$
```

Step 16. Run the sudo multipath -ll command on one of the workers nodes to verify all four paths from worker node to storage target are being used. As you see below there are four active running paths for each volume.



```
[core@worker3 ~]$ sudo multipath -ll
3624a937059471be632aa4fd20001c6cf dm-0 PURE,FlashArray
size=2.0T features='0' hwhandler='1 alua' wp=rw
`-+- policy='service-time 0' prio=50 status=active
  |- 5:0:0:2 sdi 8:128 active ready running
  |- 4:0:0:2 sdg 8:96  active ready running
  |- 6:0:0:2 sdl 8:176 active ready running
  `- 7:0:0:2 sdm 8:192 active ready running
3624a937059471be632aa4fd20001c6d1 dm-1 PURE,FlashArray
size=32G features='0' hwhandler='1 alua' wp=rw
`-+- policy='service-time 0' prio=50 status=active
  |- 5:0:0:1 sdh 8:112 active ready running
  |- 4:0:0:1 sdf 8:80  active ready running
  |- 7:0:0:1 sdk 8:160 active ready running
  `- 6:0:0:1 sdj 8:144 active ready running
[core@worker3 ~]$
```

Now the Portworx StorageCluster is ready to use.

Procedure 3. Dynamic Volume Provisioning and Data Protection

Portworx by Pure Storage allows platform administrators to create custom Kubernetes StorageClasses to offer different classes of service to their developers. Administrators can customize things like the replication factors, snapshot schedules, file system types, io-profiles, etc. in their storage class definitions. Developers can then choose to use a different storage class for their test/dev workloads and a different storage class for their production applications and so on.

This procedure details how to create customized Storage Classes (SCs) for dynamic provisioning of volumes (PVs) using PersistentVolumeClaims (PVCs).

For instance, the following manifest files are used to create two SCs. One for provisioning shared volume with ReadWriteMany (RWX) attribute to share a volume among multiple pods at the same time for read-write access. Other SC is for provisioning the volumes for OpenShift Virtual Machines. For provisioning volumes for typical application pods, predefined SCs can be leveraged.

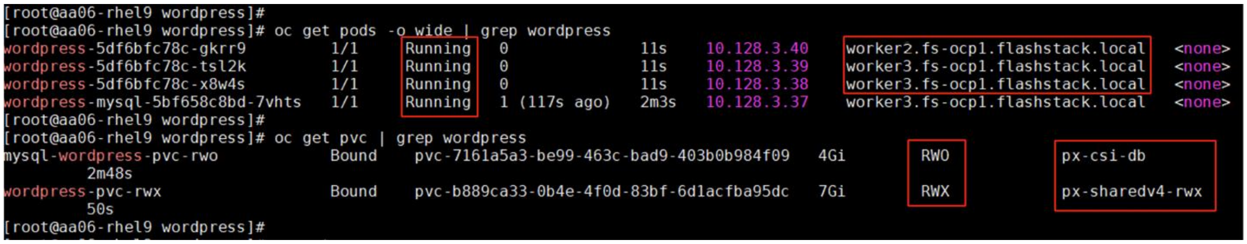
```
## This SC is used to provision sharedv4 Service volumes (exposed as CLusterIP service) with two replicas.
## cat sharedv4-sc-svc.yaml
apiVersion: storage.k8s.io/v1
kind: StorageClass
metadata:
  name: px-sharedv4-svc
provisioner: pxd.portworx.com
parameters:
  repl: "2"
  sharedv4: "true"
  sharedv4_svc_type: "ClusterIP"
reclaimPolicy: Retain
allowVolumeExpansion: true

## This SC is used to provision sharedv4 Service volumes (with specific NFS settings) with two replicas.
## cat px-rwx-kubevirt.yaml
apiVersion: storage.k8s.io/v1
kind: StorageClass
metadata:
  name: px-sharedv4-kubevirt
provisioner: pxd.portworx.com
parameters:
  repl: "2"
  sharedv4: "true"
  sharedv4_mount_options: vers=3.0,nolock
  sharedv4_svc_type: "ClusterIP"
  volumeBindingMode: immediate
reclaimPolicy: Retain
allowVolumeExpansion: true
```

Procedure 4. Deploy sample WordPress Application with Portworx Sharedv4 Service volumes

Step 1. Use the following manifests are to create a sharedv4 service volumes to be consumed my multiple WordPress application accessing the sharev4 service volume (consumed by multiple WordPress pods) and ReadWriteOnce volume (Consumed by one MySQL pod).

```
## Deploying MySql database manifests
## create password.txt and update it with mysql root password.
kubectl create secret generic mysql-pass --from-file=./password.txt
## mysql PVC: cat mysql-pvc.yaml
apiVersion: v1
kind: PersistentVolumeClaim
metadata:
  name: mysql-wordpress-pvc-rwo
  annotations:
spec:
  storageClassName: px-csi-db
  accessModes:
    - ReadWriteOnce
  resources:
    requests:
      storage: 4Gi
## mysql deployment manifest: cat mysql-dep.yaml
apiVersion: apps/v1
kind: Deployment
metadata:
  name: wordpress-mysql
  labels:
    app: wordpress
spec:
  selector:
    matchLabels:
      app: wordpress
  strategy:
    type: Recreate
  template:
    metadata:
      labels:
        app: wordpress
        tier: mysql
    spec:
      # Use the Stork scheduler to enable more efficient placement of the pods
      schedulerName: stork
      containers:
      - image: mysql:5.6
        imagePullPolicy:
        name: mysql
        env:
          # $ kubectl create secret generic mysql-pass --from-file=password.txt
          # make sure password.txt does not have a trailing newline
        - name: MYSQL_ROOT_PASSWORD
          valueFrom:
            secretKeyRef:
              name: mysql-pass
              key: password.txt
        ports:
        - containerPort: 3306
          name: mysql
        volumeMounts:
        - name: mysql-persistent-storage
          mountPath: /var/lib/mysql
      volumes:
      - name: mysql-persistent-storage
        persistentVolumeClaim:
          claimName: mysql-wordpress-pvc-rwo

##worpess PVC created with ReadWriteMany access mode: cat wp-pvc.yaml
apiVersion: v1
kind: PersistentVolumeClaim
metadata:
  name: wordpress-pvc-rwx
  labels:
    app: wordpress
spec:
  storageClassName: px-sharedv4-rwx
  accessModes:
    - ReadWriteMany
  resources:
    requests:
      storage: 7Gi

## Deployment manifest for wordpress application: cat wp-dep.yaml
apiVersion: apps/v1
kind: Deployment
metadata:
  name: wordpress
  labels:
    app: wordpress
spec:
  selector:
    matchLabels:
      app: wordpress
  replicas: 3
  strategy:
    type: Recreate
  template:
    metadata:
      labels:
        app: wordpress
        tier: frontend
    spec:
      # Use the Stork scheduler to enable more efficient placement of the pods
      schedulerName: stork
      containers:
      - image: wordpress:4.8-apache
        name: wordpress
        imagePullPolicy:
        env:
        - name: WORDPRESS_DB_HOST
          value: wordpress-mysql
        - name: WORDPRESS_DB_PASSWORD
          valueFrom:
            secretKeyRef:
              name: mysql-pass
              key: password.txt
        ports:
        - containerPort: 80
          name: wordpress
        volumeMounts:
        - name: wordpress-persistent-storage
          mountPath: /var/www/html
      volumes:
      - name: wordpress-persistent-storage
        persistentVolumeClaim:
          claimName: wordpress-pvc-rwx
```

The following screenshot shows that 3 WordPress pods are created accessing the same sharedv4 volume with ReadWriteMany access mode and one mysql pod created with single volume with ReadWriteOnce access mode.



```
[root@aa06-rhel9 wordpress]#
[root@aa06-rhel9 wordpress]# oc get pods -o wide | grep wordpress
wordpress-5df6bfc78c-gkrr9         1/1     Running   0              11s     10.128.3.40    worker2.fs-ocp1.flashstack.local   <none>
wordpress-5df6bfc78c-tsl2k         1/1     Running   0              11s     10.128.3.39    worker3.fs-ocp1.flashstack.local   <none>
wordpress-5df6bfc78c-x8w4s         1/1     Running   0              11s     10.128.3.38    worker3.fs-ocp1.flashstack.local   <none>
wordpress-mysql-5bf658c8bd-7vhts   1/1     Running   1 (117s ago)   2m3s    10.128.3.37    worker3.fs-ocp1.flashstack.local   <none>
[root@aa06-rhel9 wordpress]#
[root@aa06-rhel9 wordpress]# oc get pvc | grep wordpress
mysql-wordpress-pvc-rwo            Bound    pvc-7161a5a3-be99-463c-bad9-403b0b984f09   4Gi        RWO            px-csi-db
          2m48s
wordpress-pvc-rwx                  Bound    pvc-b889ca33-0b4e-4f0d-83bf-6d1acfba95dc   7Gi        RWX            px-sharedv4-rwx
          50s
[root@aa06-rhel9 wordpress]#
```

Sharedv4 Volume with ReadWriteMany (RWX) Access: Three WordPress pods are accessing the same sharedv4 volume with the ReadWriteMany access mode, meaning multiple pods can concurrently read from and write to this single volume. This is especially useful for applications like WordPress that may need to scale out to handle load but still rely on a shared data volume.

ReadWriteOnce (RWO) Access for MySQL: There is also a single MySQL pod accessing a volume with the ReadWriteOnce access mode, which allows only one pod to mount the volume at a time. This setup is common for databases, where concurrent access could lead to data inconsistencies.

This setup highlights Portworx's ability to support various access modes for different use cases, allowing flexible storage configurations that suit both shared applications (like WordPress) and single-instance databases (like MySQL). By providing the ReadWriteMany access mode with sharedv4, Portworx enables efficient scaling and resource usage, allowing applications to use shared storage across multiple pods and hosts

Snapshots and Clones: Portworx Enterprise offers data protection of volumes using volume snapshots and restore them for point in time recovery of the data. Any Storage Class that implements portworx csi driver pxd.portworx.com supports volume Snapshots.

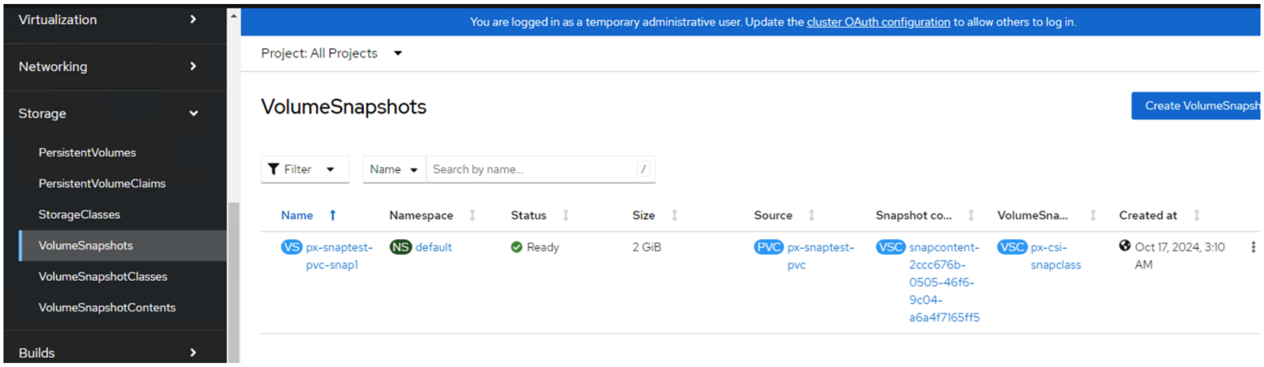
Step 2. Run the following to create VolumeSnapshotClass, thus creating a Snapshot of a PVC and then restore the snapshot as a new PVC. The following manifest is used to create VolumeSnapshotClass for taking local snap shots of the PVCs:

```
## For Openshift platform, px-csi-account service account needs to be added to the privileged security
context.
oc adm policy add-scc-to-user privileged system:serviceaccount:kube-system:px-csi-account

## Now creat VolumeSnapshotClass using below manifest
## cat VolumeStroageClass.yaml
apiVersion: snapshot.storage.k8s.io/v1
kind: VolumeSnapshotClass
metadata:
  name: px-csi-snapclass
  annotations:
    snapshot.storage.kubernetes.io/is-default-class: "true"
driver: pxd.portworx.com
deletionPolicy: Delete
parameters:
  csi.openstorage.org/snapshot-type: local
```

Step 3. Use the following sample manifest to create a sample pvc:

```
## cat px-snaptest-pvc.yaml
kind: PersistentVolumeClaim
apiVersion: v1
metadata:
  name: px-snaptest-pvc
spec:
  storageClassName: px-csi-db. ## Any Storage Class can be used which implements Portworx CSI driver.
  accessModes:
    - ReadWriteOnce
  resources:
    requests:
      storage: 2Gi
## Assume this pvc is attached to a pod and the pod has written some data into the pvc.
## Now create Snapshot of the above volume. It can be created using UI also.
## cat create-snapshot-snaptest-pvc.yaml
apiVersion: snapshot.storage.k8s.io/v1
kind: VolumeSnapshot
metadata:
  name: px-snaptest-pvc-snap1
spec:
  volumeSnapshotClassName: px-csi-snapclass
  source:
    persistentVolumeClaimName: px-snaptest-pvc. ## the name of the pvc
```

The following screenshot shows px-snaptest-pvc-snap1 is the snapshot of the PVC px-snaptest-pvc:



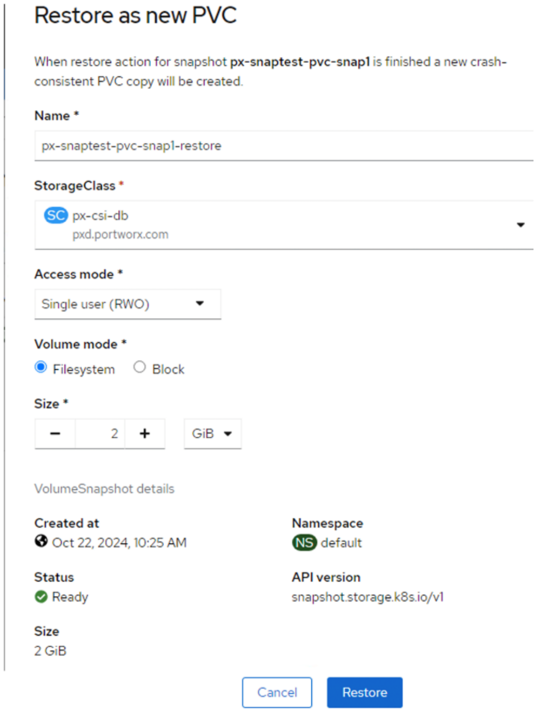
You are logged in as a temporary administrative user. Update the cluster OAuth configuration to allow others to log in.

VolumeSnapshots

Name	Namespace	Status	Size	Source	Snapshot co...	VolumeSna...	Created at
VS px-snaptest-pvc-snap1	NS default	✓ Ready	2 GiB	PVC px-snaptest-pvc	VSC snapcontent-2ccc676b-0505-46f6-9c04-a6a4f7165ff5	VSC px-csi-snapclass	Oct 17, 2024, 3:10 AM

Step 4. You can now restore this as snapshot as a new PVC and then can be mounted to any other pod.

Step 5. Click the ellipses of the snapshot and select **Restore as new PVC**. In the **Restore as new PVC** window**,** click **Restore**.



Restore as new PVC

When restore action for snapshot **px-snaptest-pvc-snap1** is finished a new crash-consistent PVC copy will be created.

Name *
px-snaptest-pvc-snap1-restore

StorageClass *
SC px-csi-db
pxd.portworx.com

Access mode *
Single user (RWO)

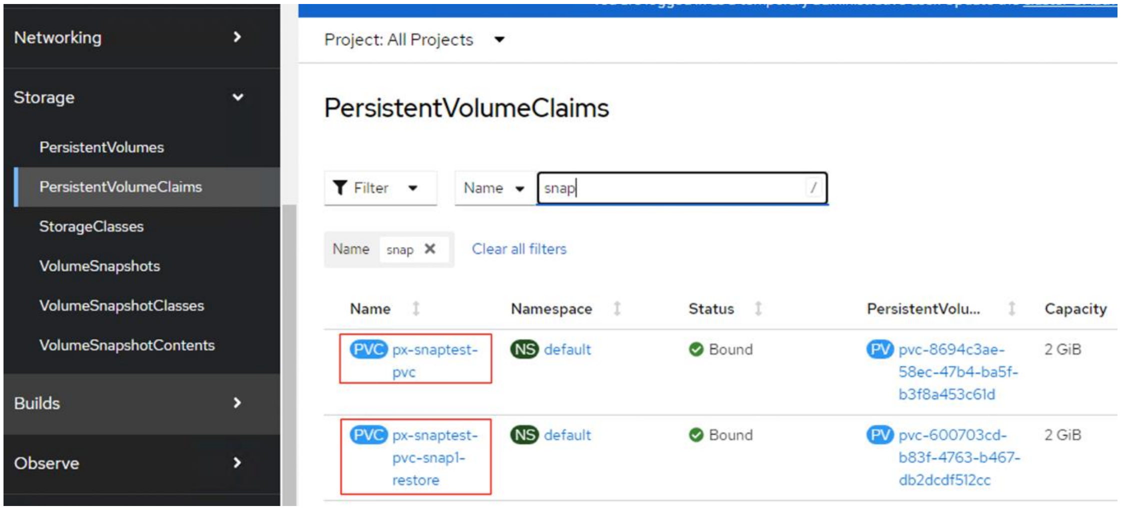
Volume mode *
● Filesystem ○ Block

Size *
− 2 + GiB

VolumeSnapshot details

Created at Oct 22, 2024, 10:25 AM	**Namespace** NS default
Status ✓ Ready	**API version** snapshot.storage.k8s.io/v1
Size 2 GiB	

Step 6. You can view the original and restored PVC under PVC list.

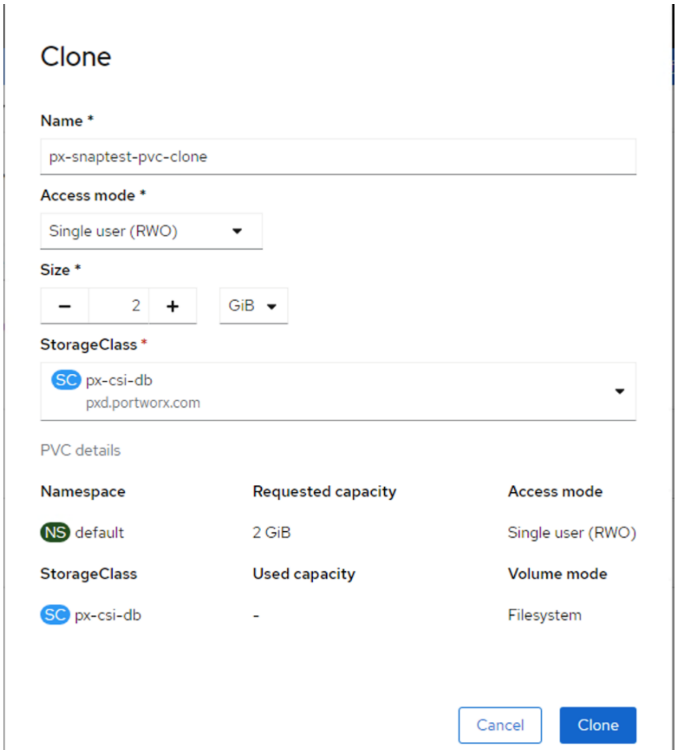


PersistentVolumeClaims

Name: snap

Name	Namespace	Status	PersistentVolu...	Capacity
PVC px-snaptest-pvc	NS default	✓ Bound	PV pvc-8694c3ae-58ec-47b4-ba5f-b3f8a453c61d	2 GiB
PVC px-snaptest-pvc-snap1-restore	NS default	✓ Bound	PV pvc-600703cd-b83f-4763-b467-db2dcdf512cc	2 GiB

Step 7. To clone of a PVC (px-snaptest-pvc), click three of the PVC and select **Clone PVC**. Click **Clone**.



Clone

Name *

px-snaptest-pvc-clone

Access mode *

Single user (RWO)

Size *

2 GiB

StorageClass *

SC px-csi-db
pxd.portworx.com

PVC details

Namespace	Requested capacity	Access mode
NS default	2 GiB	Single user (RWO)
StorageClass	**Used capacity**	**Volume mode**
SC px-csi-db	-	Filesystem

Portworx Enterprise Console Plugin for OpenShift

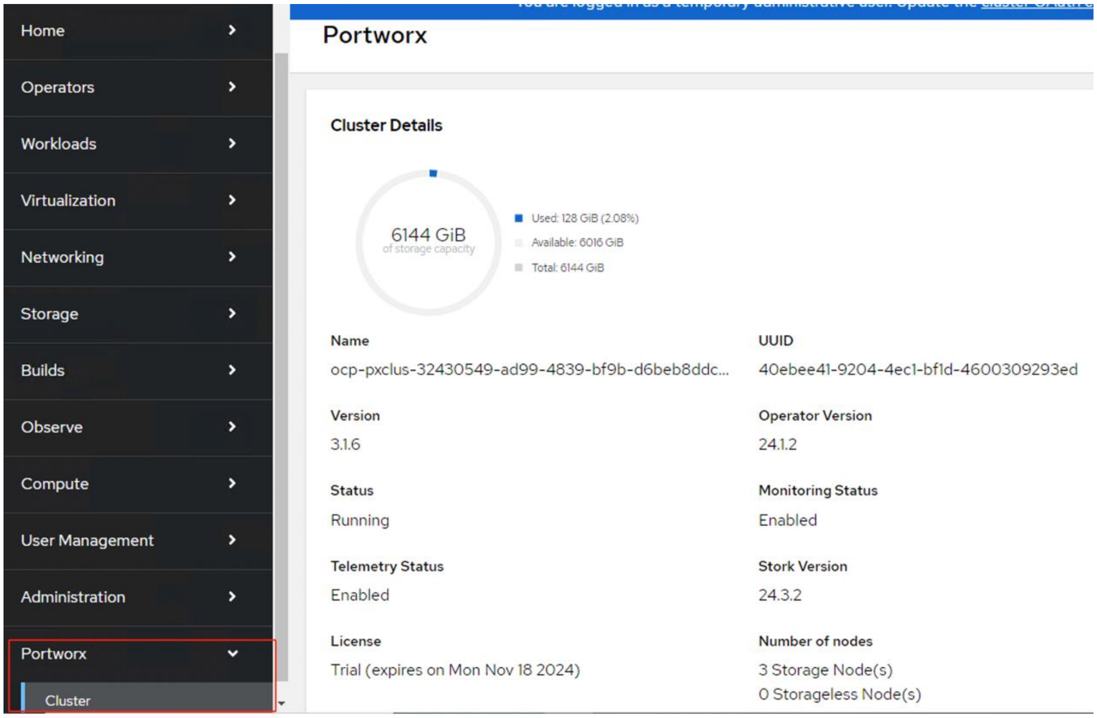
Portworx by Pure Storage has built an OpenShift Dynamic console plugin that enables single-pane-of-glass management of storage resources running on Red Hat OpenShift clusters. This allows platform administrators to use the OpenShift web console to manage not just their applications and their OpenShift cluster, but also their Portworx Enterprise installation and their stateful applications running on OpenShift.

This plugin can be installed with a single click during the Installation of Portworx Operator as explained in the previous sections. Once the plugin is enabled, the Portworx Operator will automatically install the plugin pods in the same OpenShift project as the Portworx storage cluster. When the pods are up and running, administrators will see a message in the OpenShift web console to refresh their browser window for the Portworx tabs to show up in the UI.

With this plugin, Portworx has built three different UI pages, including a Portworx Cluster Dashboard that shows up in the left navigation menu, a Portworx tab under Storage > Storage Class section, and another Portworx tab under Storage > Persistent Volume Claims.

Portworx Cluster Dashboard

Platform administrators can use the Portworx Cluster Dashboard to monitor the status of their Portworx Storage Cluster and their persistent volumes and storage nodes. Here are a few operations that are now streamlined by the OpenShift Dynamic plugin from Portworx.



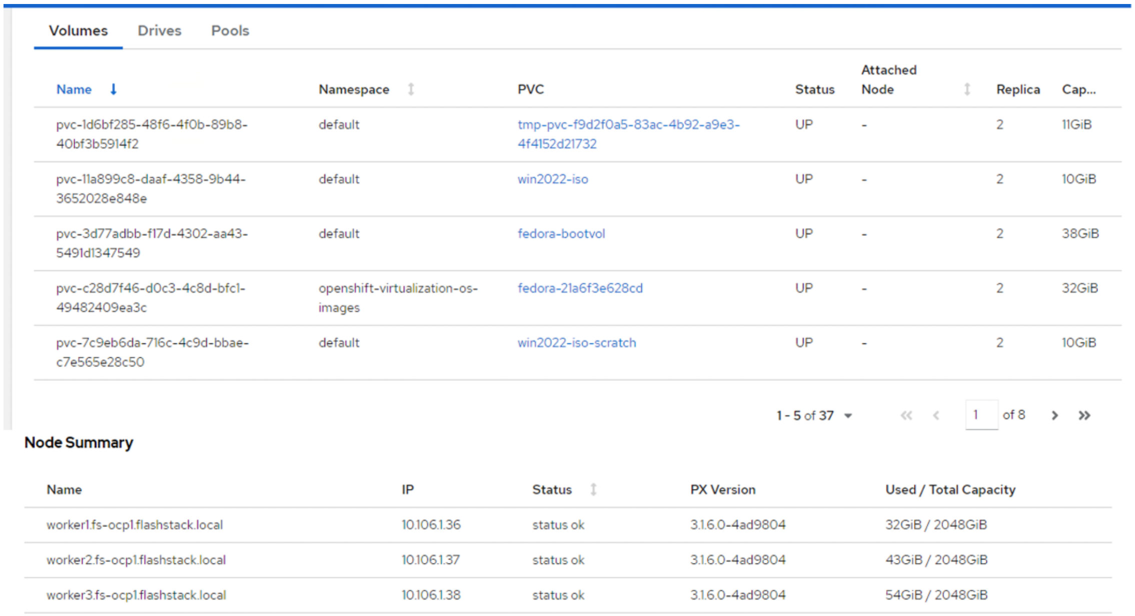
Portworx

Cluster Details

6144 GiB
of storage capacity

- Used: 128 GiB (2.08%)
- Available: 6016 GiB
- Total: 6144 GiB

Name ocp-pxclus-32430549-ad99-4839-bf9b-d6beb8ddc...	UUID 40ebee41-9204-4ec1-bf1d-4600309293ed
Version 3.1.6	Operator Version 24.1.2
Status Running	Monitoring Status Enabled
Telemetry Status Enabled	Stork Version 24.3.2
License Trial (expires on Mon Nov 18 2024)	Number of nodes 3 Storage Node(s) 0 Storageless Node(s)



Name	Namespace	PVC	Status	Attached Node	Replica	Cap...
pvc-1d6bf285-48f6-4f0b-89b8-40bf3b5914f2	default	tmp-pvc-f9d2f0a5-83ac-4b92-a9e3-4f4152d21732	UP	-	2	11GiB
pvc-11a899c8-daaf-4358-9b44-3652028e848e	default	win2022-iso	UP	-	2	10GiB
pvc-3d77adbb-f17d-4302-aa43-5491d1347549	default	fedora-bootvol	UP	-	2	38GiB
pvc-c28d7f46-d0c3-4c8d-bfc1-49482409ea3c	openshift-virtualization-os-images	fedora-21a6f3e628cd	UP	-	2	32GiB
pvc-7c9eb6da-716c-4c9d-bbae-c7e565e28c50	default	win2022-iso-scratch	UP	-	2	10GiB

1 - 5 of 37 1 of 8

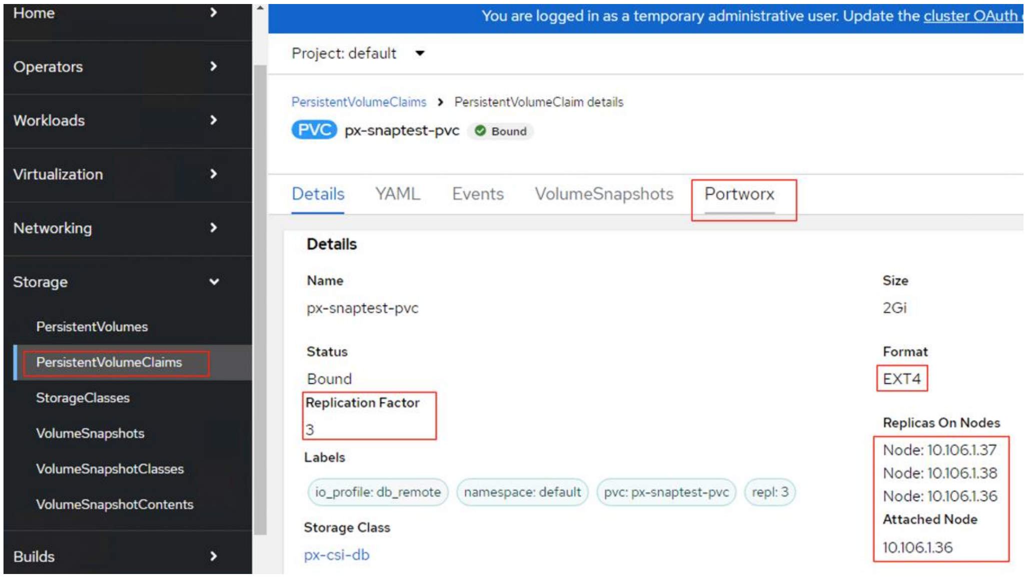
Node Summary

Name	IP	Status	PX Version	Used / Total Capacity
worker1.fs-ocp1.flashstack.local	10.106.1.36	status ok	3.1.6.0-4ad9804	32GiB / 2048GiB
worker2.fs-ocp1.flashstack.local	10.106.1.37	status ok	3.1.6.0-4ad9804	43GiB / 2048GiB
worker3.fs-ocp1.flashstack.local	10.106.1.38	status ok	3.1.6.0-4ad9804	54GiB / 2048GiB

To obtain detailed inventory information of the Portworx Cluster, click the Drives and Pools tabs.

Portworx PVC Dashboard

This dashboard shows some of the important attributes of a PVC like Replication Factor, node details of replicas, attached node and so on. You may have to use multiple pxctl inspect volume CLI commands to get these details. Instead, all this information can be found in Console Plugin.



px-snaptest-pvc

Details

Name: px-snaptest-pvc

Size: 2Gi

Status: Bound

Format: EXT4

Replication Factor: 3

Replicas On Nodes:
Node: 10.106.1.37
Node: 10.106.1.38
Node: 10.106.1.36

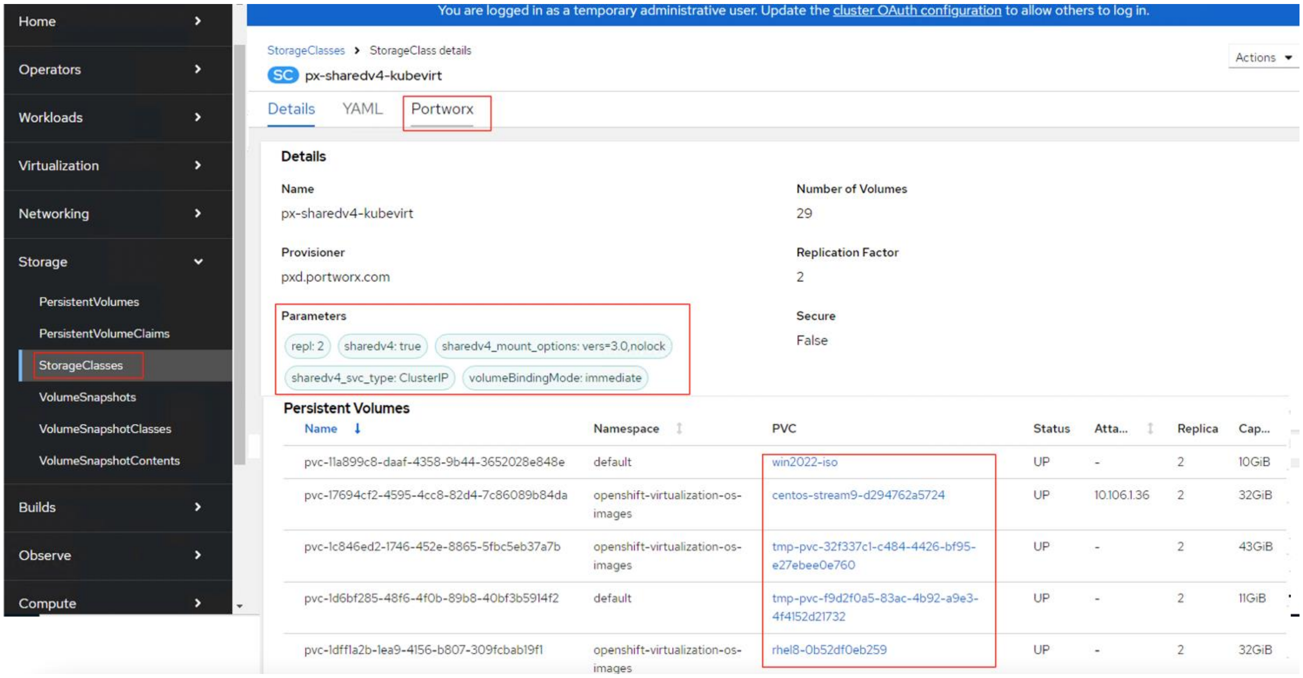
Attached Node: 10.106.1.36

Labels: io_profile: db_remote, namespace: default, pvc: px-snaptest-pvc, repl: 3

Storage Class: px-csi-db

Portworx StorageClass Dashboard

From the Portworx storage cluster tab, administrators can get details about the custom parameters set for each storage class, the number of persistent volumes dynamically provisioned using the storage class, and a table that lists all the persistent volumes deployed using that storage class. The OpenShift dynamic plugin eliminates the need for administrators to use multiple “kubectl get” and “kubectl describe” commands to find all these details—instead, they can just use a simple UI to monitor their storage classes.



px-sharedv4-kubevirt

Details

Name: px-sharedv4-kubevirt

Number of Volumes: 29

Provisioner: pxd.portworx.com

Replication Factor: 2

Parameters: repl: 2, sharedv4: true, sharedv4_mount_options: vers=3.0,nolock, sharedv4_svc_type: ClusterIP, volumeBindingMode: immediate

Secure: False

Persistent Volumes

Name	Namespace	PVC	Status	Atta...	Replica	Cap...
pvc-11a899c8-daaf-4358-9b44-3652028e848e	default	win2022-iso	UP	-	2	10GiB
pvc-17694cf2-4595-4cc8-82d4-7c86089b84da	openshift-virtualization-os-images	centos-stream9-d294762a5724	UP	10.106.1.36	2	32GiB
pvc-1c846ed2-1746-452e-8865-5fbc5eb37a7b	openshift-virtualization-os-images	tmp-pvc-32f337c1-c484-4426-bf95-e27ebee0e760	UP	-	2	43GiB
pvc-1d6bf285-48f6-4f0b-89b8-40bf3b5914f2	default	tmp-pvc-f9d2f0a5-83ac-4b92-a9e3-4f4152d21732	UP	-	2	11GiB
pvc-1dff1a2b-1ea9-4156-b807-309fcbab19f1	openshift-virtualization-os-images	rhel8-0b52df0eb259	UP	-	2	32GiB

OpenShift Virtualization

This chapter contains the following:

- OpenShift Virtualization Operator
- Post Installation Configuration
- Live Migration of VMs within the OpenShift Cluster

Red Hat® OpenShift® Virtualization, an included feature of Red Hat OpenShift, provides a modern platform for organizations to run and deploy their new and existing virtual machine (VM) workloads. The solution allows for easy migration and management of traditional virtual machines onto a trusted, consistent, and comprehensive hybrid cloud application platform.

OpenShift Virtualization is an operator included with any OpenShift subscription. It enables infrastructure architects to create and add virtualized applications to their projects from OperatorHub in the same way they would for a containerized application.

Existing virtual machines can be migrated from other platforms onto the OpenShift application platform through the use of free, intuitive migration tools. The resulting VMs will run alongside containers on the same Red Hat OpenShift nodes.

The following sections and procedures provide detailed steps to create custom network policies for creating management and iSCSI networks for virtual machines to use, steps to deploy Red Hat virtual machines using pre-defined templates, steps to create custom Windows Server template and create windows virtual machine from this custom template.

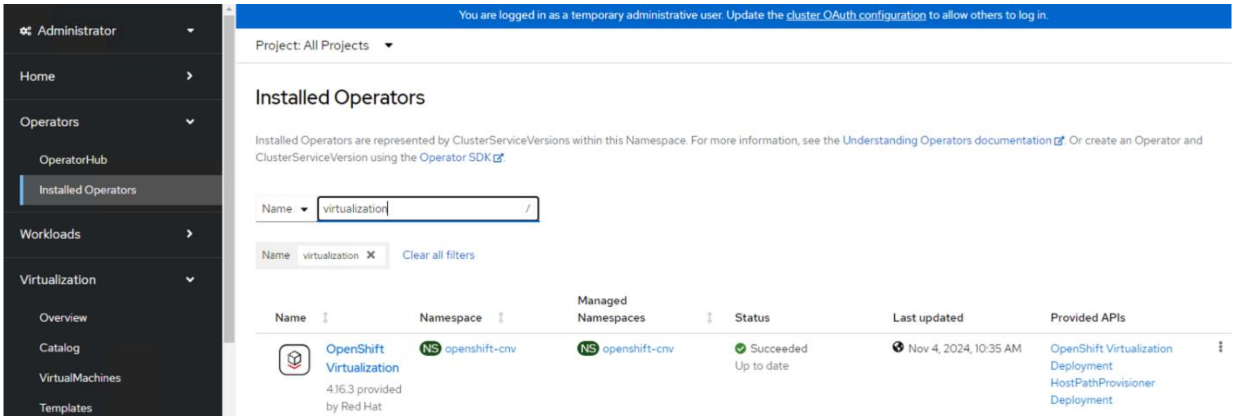
OpenShift Virtualization Operator

Procedure 1. Deploy OpenShift Virtualization Operator

Step 1. If the Red Hat OpenShift Virtualization is not deployed, go to **Operators > OperatorHub**. Type **virtualization** in All Items checkbox. From the available list, select the **OpenShift Virtualization** tile with Red Hat source label. Click **Install** to install OpenShift Virtualization with default settings.

Note: For OpenShift Virtualization operator, ensure that the Operator recommended namespace option is selected. This installs the Operator in the mandatory openshift-cnv namespace, which is automatically created if it does not exist.

Step 2. When the operator is installed successfully, go to **Operators > Installed Operators** and type **virtualization** under the Name checkbox and verify that operator is installed successfully.



You are logged in as a temporary administrative user. Update the cluster OAuth configuration to allow others to log in.

Project: All Projects

Installed Operators

Installed Operators are represented by ClusterServiceVersions within this Namespace. For more information, see the Understanding Operators documentation. Or create an Operator and ClusterServiceVersion using the Operator SDK.

Name: virtualization

Name	Namespace	Managed Namespaces	Status	Last updated	Provided APIs
OpenShift Virtualization 4.16.3 provided by Red Hat	NS openshift-cnv	NS openshift-cnv	Succeeded Up to date	Nov 4, 2024, 10:35 AM	OpenShift Virtualization Deployment HostPathProvisioner Deployment

Post Installation Configuration

The procedures in this section are typically performed after OpenShift Virtualization is installed. You can configure the components that are relevant for your environment:

- Node placement rules for OpenShift Virtualization Operators, workloads, and controllers: The default scheduling for virtual machines (VMs) on bare metal nodes is appropriate. Optionally, you can specify the nodes where you want to deploy OpenShift Virtualization Operators, workloads, and controllers by configuring node placement rules. For detailed options on VM scheduling and placement options, see: https://docs.openshift.com/container-platform/4.16/virt/post_installation_configuration/virt-node-placement-virt-components.html#virt-node-placement-virt-components
- Storage Configuration: Storage profile must be configured for OpenShift virtualization. A storage profile provides recommended settings based on the associated storage class. When the Portworx Enterprise is deployed on the OpenShift Cluster, it deploys several storage classes with different settings for different use-cases. OpenShift Virtualization automatically creates a storage profile with the recommended storage settings based on the associated storage class. Hence there is no need for additional configuration.
- A Default Storage Class must be configured for the OpenShift Cluster. Otherwise, the cluster cannot receive automated boot source updates. To configure an existing storage class (created by Portworx) as Default Storage Class run the following command:

  ```
  kubectl patch storageclass <StorageClassName> -p '{"metadata":
  {"annotations":{"storageclass.kubernetes.io/is-default-class":"true"}}}'
  ```

- Network configuration: By default, OpenShift Virtualization is installed with a single, internal pod network. After you install OpenShift Virtualization, you can install networking Operators and configure additional networks.

The following sections provides more details on network configurations validated in this solution.

OpenShift NMState Operator

By default, the virtual machines on the OpenShift cluster are connected to the default Pod network (10.x.x.x/14) running within the OpenShift cluster and therefore these VMs can be accessed inside the cluster only. There may be cases, where in these VMs should be accessed from the existing external network and these VMs need access to the services and resources that are hosted out the OpenShift cluster. For such scenarios, by using NMState operator, OpenShift enables us to customize the network configurations on the worker nodes.

The Kubernetes NMState Operator provides a Kubernetes API for performing state-driven network configuration across the OpenShift Container Platform cluster's nodes with NMState. The Kubernetes NMState Operator provides users with functionality to configure various network interface types, DNS, and routing on cluster nodes. Additionally, the daemons on the cluster nodes periodically report on the state of each node's network interfaces to the API server. For more details on the NMState operator and its CRDs, see: https://docs.openshift.com/container-platform/4.17/networking/k8s_nmstate/k8s-nmstate-about-the-k8s-nmstate-operator.html

In this solution, the NMState operator and its CRDs are used for achieving the following requirements of the virtual machines hosted on Red Hat OpenShift cluster:

- Allow the virtual machines to access external resources and services such as Active Directory, DNS, NFS shares and so on.
- To access the VMs (using RDP, SSH, and so on) from the already existing management network.
- Allow the VMs to directly access the iSCSI storage targets directly using "In-Guest" iSCSI initiator.

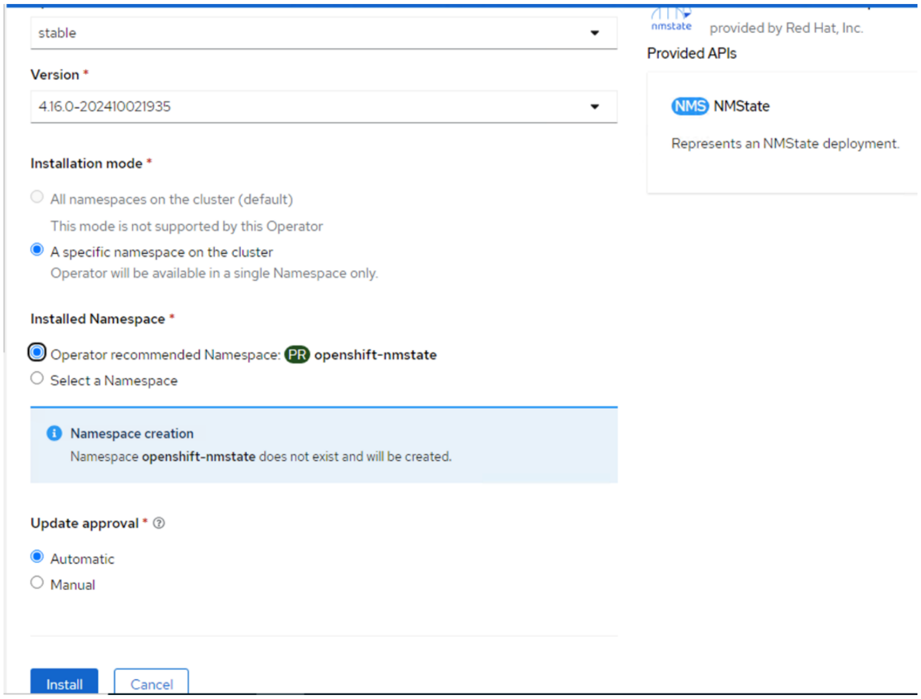
To meet these requirements, custom Node Network Configuration Policies are defined using NMState operator to create the following networks on each worker nodes.

Table 14. Customer Network Configurations

Network Name	Type of network created on each worker node	Worker Interface Used	VLAN	Subnets and GW	Purpose
VM-Management	bridge	eno8	1062	10.106.2.0/24 10.106.2.254	To allow the VMs to access already existing customer services and resources residing outside the cluster To access the VMs from already existing management network.
VM-iSCSI-A	bridge	eno9	3010	192.168.51.0/24	To allow the VMs to access iSCSI storage target via fabric-A
VM-iSCSI-B	bridge	eno10	3020	192.168.52.0/24	To allow the VMs to directly access iSCSI storage targets via fabric-B for using ‘In-Guest’ iSCSI

Procedure 1. Create Custom networks for Virtual Machines (optional)

Step 1. Log into OpenShift web console, go to **Operators > OperatorHub** and search for **NMstate**. Click **Kubernetes NMState Operator** and click **Install**. In the Install Operator window, click **Install** with default settings.



stable

Version *
4.16.0-202410021935

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.

Update approval *
● Automatic
○ Manual

nmstate provided by Red Hat, Inc.

Provided APIs

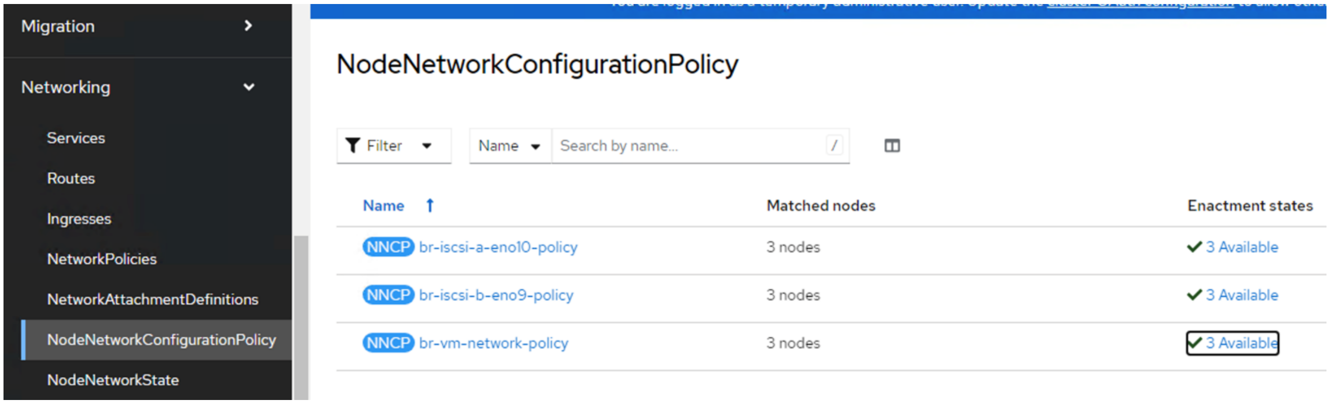
NMS NMState
Represents an NMState deployment.

Step 2. When NMState operator is installed, go to **Operators > Installed Operators**, click **Details of NMState** operator, click **Create Instance** and create NMState instance with default settings. When NMState is created, Refresh the browser to see the new options (NodeNetworkConfigurationPolicy, NetworkAttachmentDefinitions, NodeNetworkState) under the Network tab.

Step 3. Use the following manifests to create NodeNetworkConfigurationPolicy policies for each type of network:

```
## NodeNetworkConfiguration policy for creating a linux bridge for Vm-Management network by using eno8
physical interface of the worker.
## cat vm-network-bridge.yaml
apiVersion: nmstate.io/v1
kind: NodeNetworkConfigurationPolicy
metadata:
  name: br-vm-network-policy
spec:
  nodeSelector:
    node-role.kubernetes.io/worker: ''
  desiredState:
    interfaces:
      - name: br-vm-network
        description: Linux bridge with eno8 as a port
        type: linux-bridge
        state: up
        ipv4:
          enabled: false
        bridge:
          options:
            stp:
              enabled: false
          port:
            - name: eno8

#### NodeNetworkConfiguration policy for creating a linux bridge for Vm-iSCSI-A network by using eno9
physical interface of the worker.

## cat iscsi-a-bridge-fs.yaml
apiVersion: nmstate.io/v1
kind: NodeNetworkConfigurationPolicy
metadata:
  name: br-iscsi-a-eno9-policy
spec:
  nodeSelector:
    node-role.kubernetes.io/worker: ''
  desiredState:
    interfaces:
      - name: br-iscsi-a
        description: Linux bridge with eno9 as a port
        type: linux-bridge
        state: up
        ipv4:
          enabled: false
        bridge:
          options:
            stp:
              enabled: false
          port:
            - name: eno9

#### NodeNetworkConfiguration policy for creating a linux bridge for Vm-iSCSI-B network by using eno10
physical interface of the worker.

##cat iscsi-b-bridge-fs.yaml
apiVersion: nmstate.io/v1
kind: NodeNetworkConfigurationPolicy
metadata:
  name: br-iscsi-b-eno11-policy
spec:
  nodeSelector:
    node-role.kubernetes.io/worker: ''
  desiredState:
    interfaces:
      - name: br-iscsi-b
        description: Linux bridge with eno10 as a port
        type: linux-bridge
        state: up
        ipv4:
          enabled: false
        bridge:
          options:
            stp:
              enabled: false
          port:
            - name: eno10
```

When these polices are created, they will be applied on each worker node as shown below:



NodeNetworkConfigurationPolicy

Name	Matched nodes	Enactment states
NNCP br-iscsi-a-eno10-policy	3 nodes	✓ 3 Available
NNCP br-iscsi-b-eno9-policy	3 nodes	✓ 3 Available
NNCP br-vm-network-policy	3 nodes	✓ 3 Available

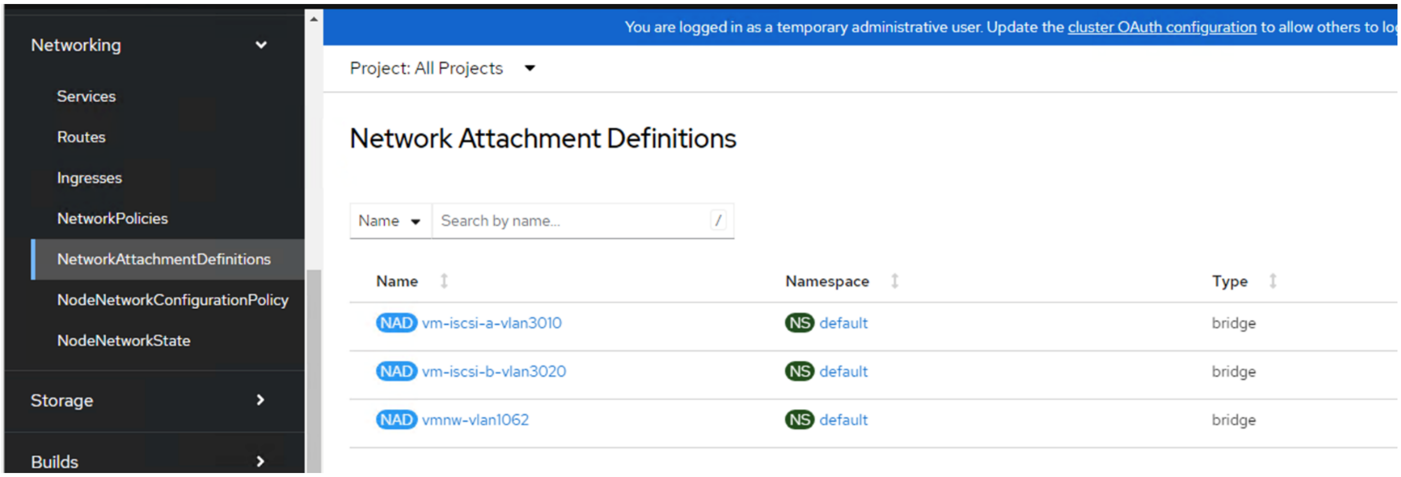
Note: When these polices are applied on the worker nodes, you will no longer see the eno8,eno9 and eno10 interfaces on the worker nodes. Instead, corresponding Linux-Bridges will be created on these physical interfaces.

Step 4. Use the following manifest files to create NetworkAttachmentDefinitions using each of the network configuration policies:

```
## NetworkAttachment policy Vm-Management network.
## cat vmnw-vlan1062-attachment.yml
apiVersion: k8s.cni.cncf.io/v1
kind: NetworkAttachmentDefinition
metadata:
  annotations:
    description: VM Management NW
    k8s.v1.cni.cncf.io/resourceName: bridge.network.kubevirt.io/br-vm-network
  name: vmnw-vlan1062
  namespace: default
spec:
  config: '{"name":"vmnw-vlan1062","type":"bridge","bridge":"br-vm-network","macspoofchk":false}'

## NetworkAttachment policy iSCSI-A-VLAN3010 network.
## vm-iscsi-a-vlan3010-attachment.yaml
apiVersion: k8s.cni.cncf.io/v1
kind: NetworkAttachmentDefinition
metadata:
  annotations:
    k8s.v1.cni.cncf.io/resourceName: bridge.network.kubevirt.io/br-vm-iscsi-a
  name: vm-iscsi-a-vlan3010-test
spec:
  config: '{"name":"vm-iscsi-a-vlan3010","type":"bridge","bridge":"br-vm-iscsi-
a","mtu":9000,"macspoofchk":false}'

## NetworkAttachment policy iSCSI-B-VLAN3020 network.
## cat vm-iscsi-b-vlan3020-attachment.yaml
apiVersion: k8s.cni.cncf.io/v1
kind: NetworkAttachmentDefinition
metadata:
  annotations:
    k8s.v1.cni.cncf.io/resourceName: bridge.network.kubevirt.io/br-vm-iscsi-b
  name: vm-iscsi-b-vlan3020
spec:
  config: '{"name":"vm-iscsi-b-vlan3020","type":"bridge","bridge":"br-vm-iscsi-
b","mtu":9000,"macspoofchk":false}'
```

Step 5. Set mtu to **9000** for the iSCSI traffic as shown in above manifests.

Step 6. When these polices are created, you can view them under **Network > NetworkAttachmentDefinitions**.



You are logged in as a temporary administrative user. Update the cluster OAuth configuration to allow others to lo

Network Attachment Definitions

Name	Namespace	Type
NAD vm-iscsi-a-vlan3010	NS default	bridge
NAD vm-iscsi-b-vlan3020	NS default	bridge
NAD vmnw-vlan1062	NS default	bridge

Now these NetworkAttachmentDefinitions can be consumed by virtual machines.

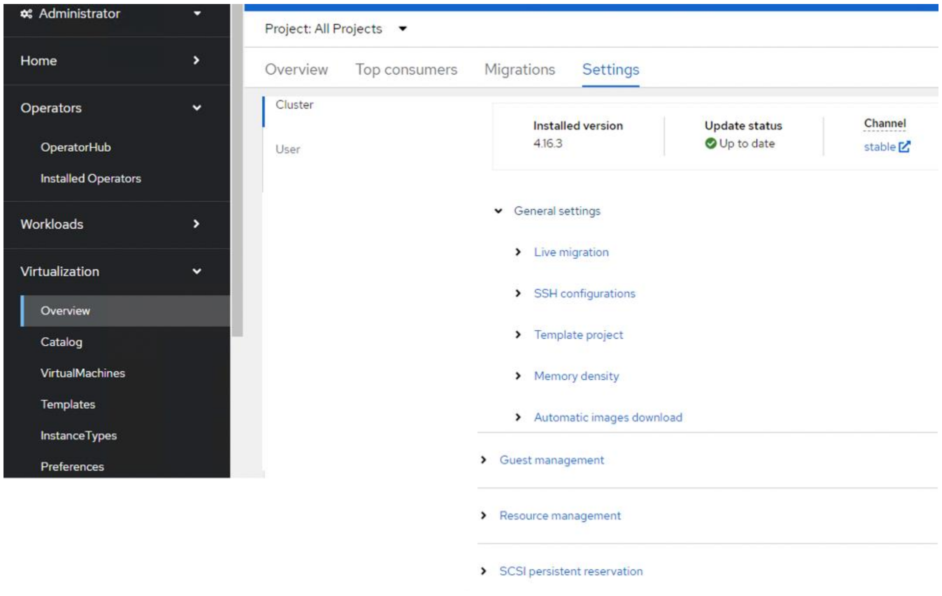
Note: For additional virtual machine management traffics with different VLANs (for instance, 1063,1064, and so on), additional NetworkAttachmentDefinitions can be defined on the same br-vm-network bridge and can be attached to the VMs.

Procedure 2. OpenShift Virtualization Settings

OpenShift Virtualization allows administrator to change some of the virtualization settings for granular control over the virtualization functionality.

Step 1. Log into the OpenShift console and go to **Virtualization > Overview > Settings** to see list of settings exposed to the administrators.

Step 2. Click the individual settings to change their values.



Installed version 4.16.3

Update status Up to date

Channel stable

Procedure 3. Create Linux-based Virtual Machines

Step 3. Create a Storage Class with specific NFS settings and Sharedv4 set true as outlined in the following manifest file. Sharedv4 volumes need to be created for Virtual Machines to facilitate live migration of the VMs.

```
## This SC is used to provision sharedv4 Service volumes (with specific NFS settings) with two replicas. Make
this SC as Default.
## cat px-rwx-kubevirt.yaml
apiVersion: storage.k8s.io/v1
kind: StorageClass
metadata:
  name: px-sharedv4-kubevirt
  annotations:
    storageclass.kubernetes.io/is-default-class: "true"
provisioner: pxd.portworx.com
parameters:
  repl: "2"
  sharedv4: "true"
  sharedv4_mount_options: vers=3.0,nolock
  sharedv4_svc_type: "ClusterIP"
  volumeBindingMode: immediate
reclaimPolicy: Retain
allowVolumeExpansion: true
```

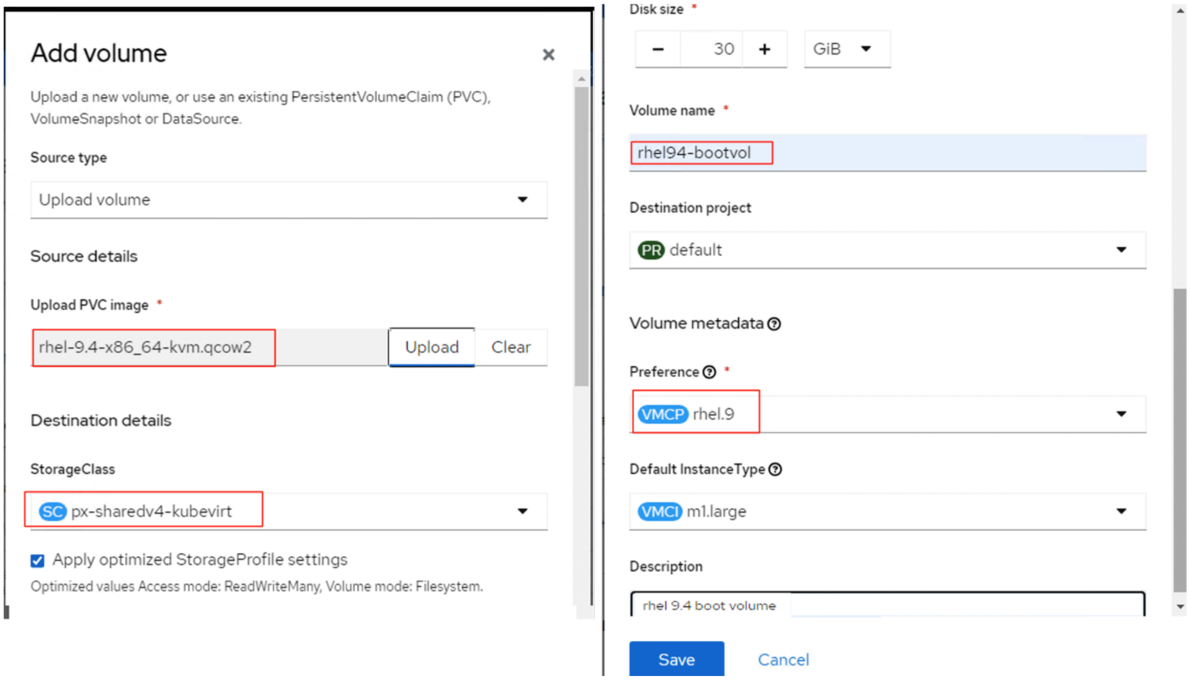
Step 4. The Automatic Images Download option, located here: **Virtualization > Overview > Settings > General Settings**, is enabled by default. The boot images for Red Hat 8/9, CentOS 9 and Fedora are downloaded automatically. The predefined templates will be automatically updated with these images as boot volumes.

Step 5. If the Automatic Images Download Option is turned off, you can download the required RedHat, Fedora and CentOS qcow2 (or KVM Guest Images) boot images from the following URLs and use the following steps to create boot volumes using downloaded boot image files.

- CentOS 7/8/9: https://cloud.centos.org/centos/
- Fedora: https://dl.fedoraproject.org/pub/fedora/linux/releases/39/
- RHEL 6/7: https://access.redhat.com/downloads/content/69
- RHEL 8/9:https://access.redhat.com/downloads/content/479

Step 6. Connect to the Red Hat OpenShift web console and go to **Virtualization > Catalog > Instance Types**. Click **Add Volume**.

Step 7. In the Add Volume screen, set Source type as **Upload Volume**. For Upload PVC Image, browse and select one of the QCOW2 images downloaded in the previous step. For this example, RHEL 9 qcow2 file is selected. Select **px-sharedv4-kubevirt** for the StorageClass and provide the details for the rest of the fields as shown below. Click **Save** after updating all the fields.



Add volume

Upload a new volume, or use an existing PersistentVolumeClaim (PVC), VolumeSnapshot or DataSource.

Source type: Upload volume

Source details

Upload PVC image *: rhel-9.4-x86_64-kvm.qcow2

Destination details

StorageClass: SC px-sharedv4-kubevirt

☑ Apply optimized StorageProfile settings

Optimized values Access mode: ReadWriteMany, Volume mode: Filesystem.

Disk size *: 30 GiB

Volume name *: rhel94-bootvol

Destination project: PR default

Volume metadata

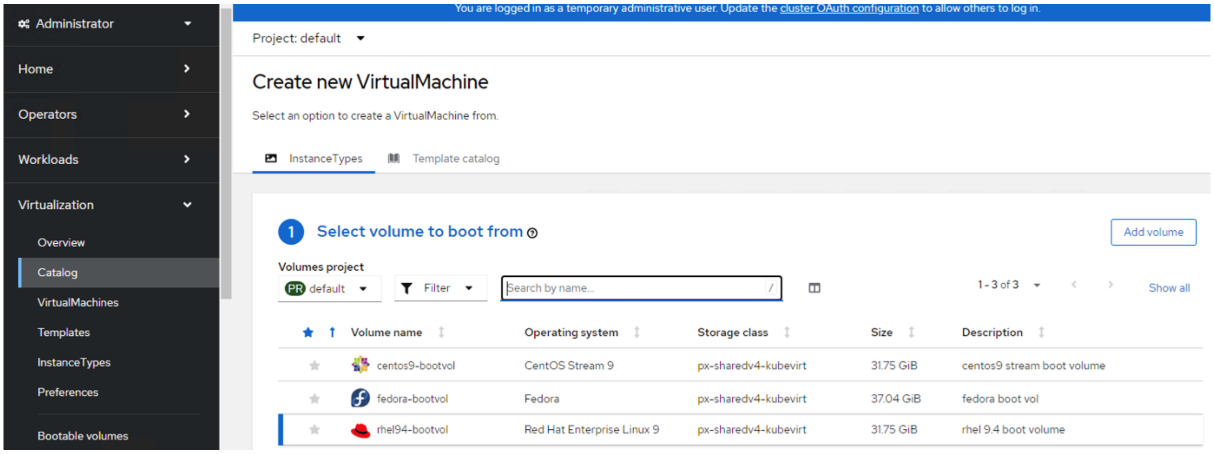
Preference *: VMCP rhel.9

Default InstanceType: VMCI m1.large

Description: rhel 9.4 boot volume

Note: When you upload the qcow2 image, a CDI (Containerized Data Importer) pod will be created to format the qcow2 to KVM compatible format. The pod will disappear after successfully importing the qcow2 image.

Step 8. Repeat steps 1 through 7 to create boot volumes for all required Linux based Operating Systems. You can view list of all the boot volumes under **Catalog** as shown below. The following image shows three boot volumes for Fedora, CentOS 9 and Red Hat 9 versions.



You are logged in as a temporary administrative user. Update the cluster OAuth configuration to allow others to log in.

Project: default

Create new VirtualMachine

Select an option to create a VirtualMachine from.

1 Select volume to boot from

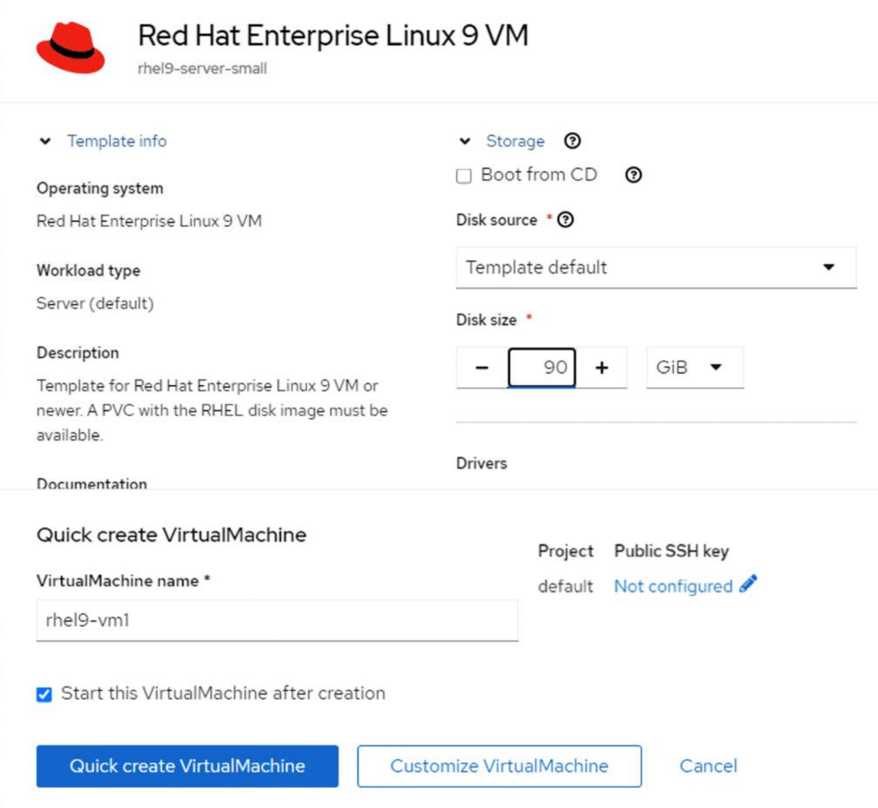
Volumes project: PR default

Volume name	Operating system	Storage class	Size	Description
centos9-bootvol	CentOS Stream 9	px-sharedv4-kubevirt	31.75 GiB	centos9 stream boot volume
fedora-bootvol	Fedora	px-sharedv4-kubevirt	37.04 GiB	fedora boot vol
rhel94-bootvol	Red Hat Enterprise Linux 9	px-sharedv4-kubevirt	31.75 GiB	rhel 9.4 boot volume

Note: Based on different versions of the uploaded OS boot images, the corresponding predefined OpenShift Virtual Machine templates will be updated to use these volumes as source volumes for booting into the Operating system. For all the templates that are updated with these volumes as boot sources volumes, you will notice these templated are marked with Source available in dark blue.

Step 9. To create a VM, go to **Virtualization > Virtual Machines > Create > From template**. Select one of the templates which has the Source available label on it.

Step 10. Provide a name to the VM and change the Disk size to the required size. Click **Customize Virtual Machine.**



Red Hat Enterprise Linux 9 VM

rhel9-server-small

Template info

Operating system: Red Hat Enterprise Linux 9 VM

Workload type: Server (default)

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

Documentation

Storage

□ Boot from CD

Disk source *: Template default

Disk size *: 90 GiB

Drivers

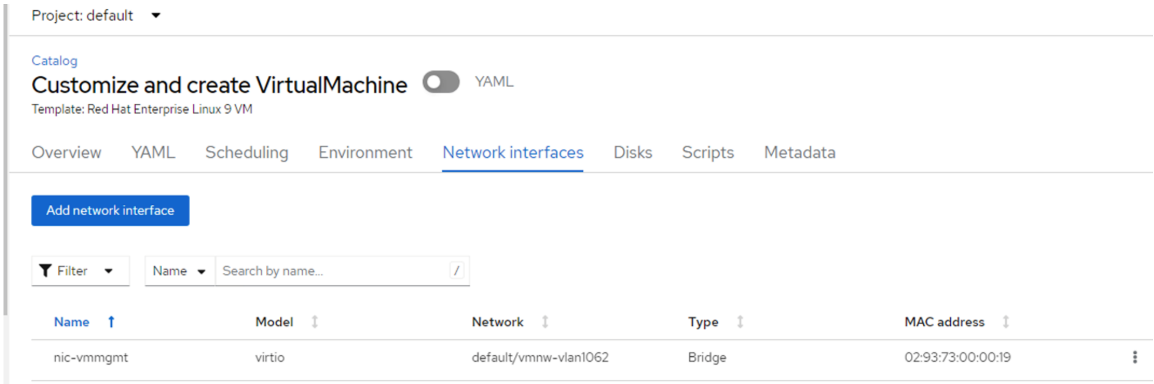
Quick create VirtualMachine

VirtualMachine name *: rhel9-vm1

Project	Public SSH key
default	Not configured

☑ Start this VirtualMachine after creation

Step 11. From the Network Interface tab, remove the preconfigured Pod network interface and add new interface on vmnw-vlan1062 network as shown below:



Project: default

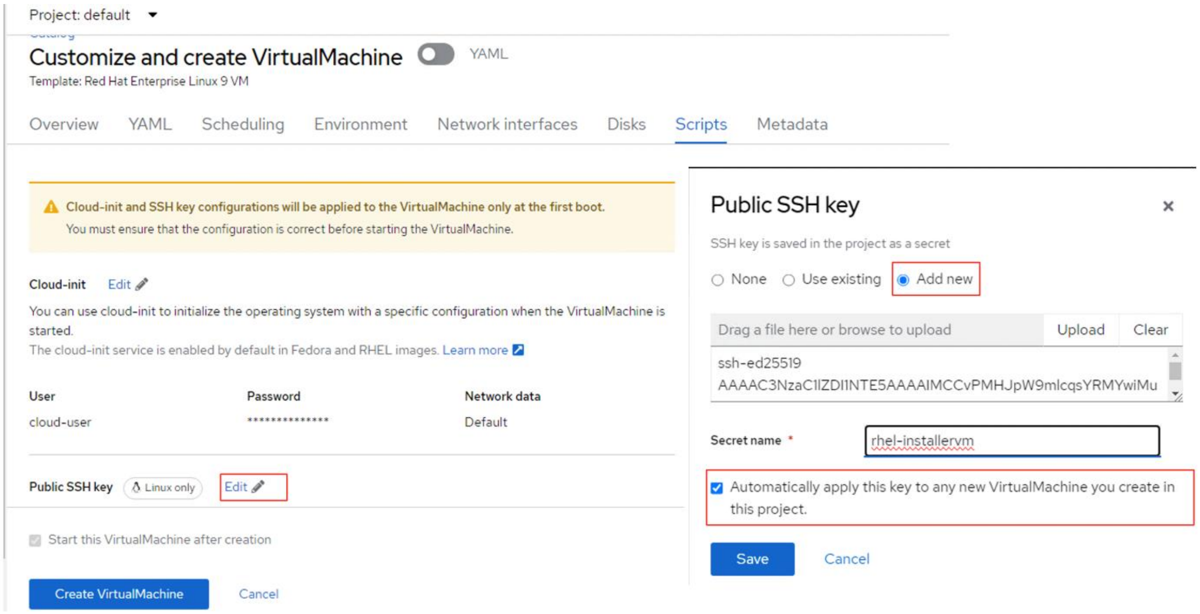
Catalog

Customize and create VirtualMachine

Template: Red Hat Enterprise Linux 9 VM

Name	Model	Network	Type	MAC address
nic-vmmgmt	virtio	default/vmnw-vlan1062	Bridge	02:93:73:00:00:19

Step 12. From the Scripts tab, click the pen icon near Public SSH. Click **Add New** radio button and upload the public key of your installer VM. Provide a name for the public key and turn on the checkbox to use this key for all the VMs you create in future. Make a note of the default user name (cloud-user) and the default password. You can click the pen icon and change default user and password.



Project: default ▾

Customize and create VirtualMachine YAML

Template: Red Hat Enterprise Linux 9 VM

⚠ Cloud-init and SSH key configurations will be applied to the VirtualMachine only at the first boot.
You must ensure that the configuration is correct before starting the VirtualMachine.

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	**************	Default

Public SSH key Linux only Edit

☑ Start this VirtualMachine after creation

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 Upload Clear

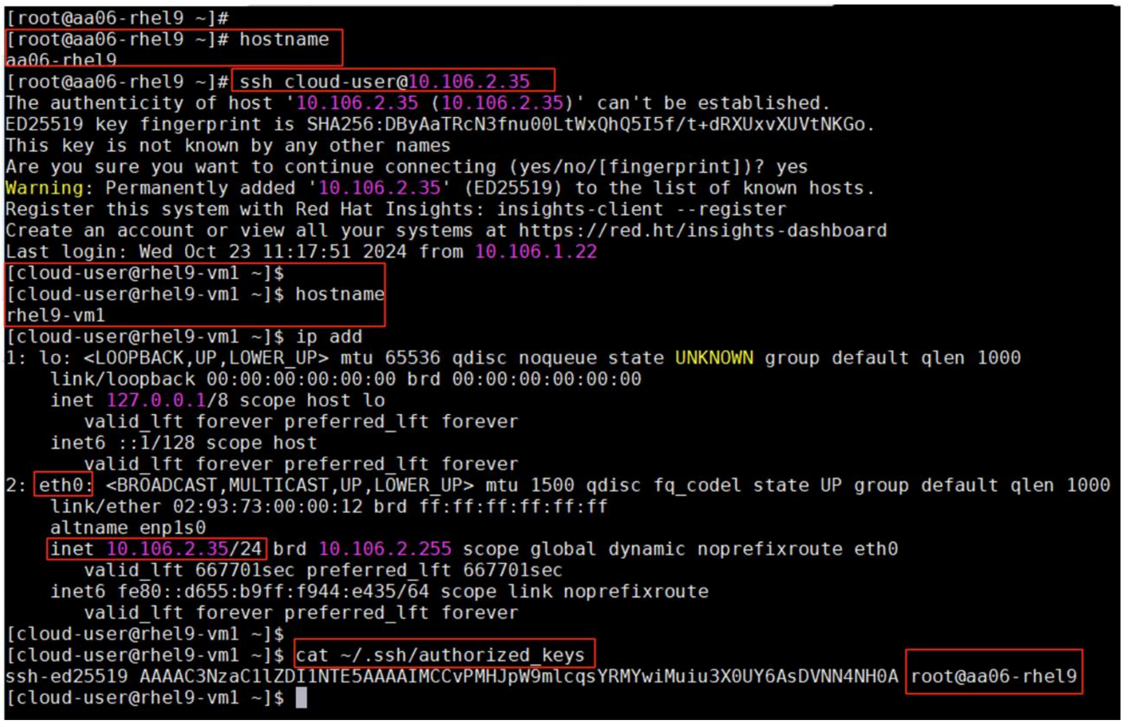
```
ssh-ed25519
AAAAC3NzaC1lZDI1NTE5AAAAIMCCvPMHJpW9mlcqsYRMYwiMu
```

Secret name * rhel-installervm

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

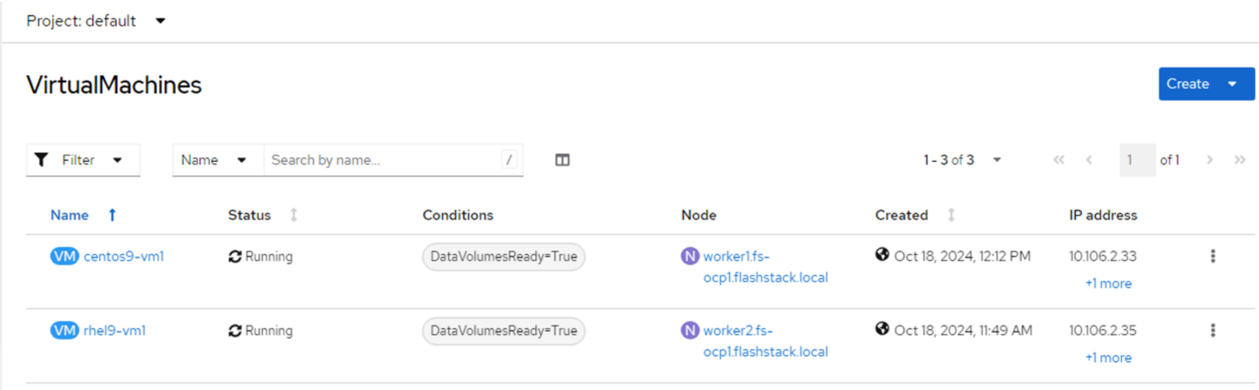
Step 13. Optionally, from the Overview tab, change the CPUs and memory resources to be allocated to the VM. Click **Create Virtual Machine.**

Step 14. In a few seconds a new virtual machine will be provisioned. The interface of the VM will get DHCP IP from **vmnw-vlan1062** network and the VM can be accessed directly from **rhel-installer** VM (**aa06-rhel9**) using its public ssh key without using password as shown below.



```
[root@aa06-rhel9 ~]#
[root@aa06-rhel9 ~]# hostname
aa06-rhel9
[root@aa06-rhel9 ~]# ssh cloud-user@10.106.2.35
The authenticity of host '10.106.2.35 (10.106.2.35)' can't be established.
ED25519 key fingerprint is SHA256:DByAaTRcN3fnu00LtWxQhQ5I5f/t+dRXUxvXUVtNKGo.
This key is not known by any other names
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added '10.106.2.35' (ED25519) to the list of known hosts.
Register this system with Red Hat Insights: insights-client --register
Create an account or view all your systems at https://red.ht/insights-dashboard
Last login: Wed Oct 23 11:17:51 2024 from 10.106.1.22
[cloud-user@rhel9-vm1 ~]$
[cloud-user@rhel9-vm1 ~]$ hostname
rhel9-vm1
[cloud-user@rhel9-vm1 ~]$ ip add
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default qlen 1000
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
       valid_lft forever preferred_lft forever
    inet6 ::1/128 scope host
       valid_lft forever preferred_lft forever
2: eth0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP group default qlen 1000
    link/ether 02:93:73:00:00:12 brd ff:ff:ff:ff:ff:ff
    altname enp1s0
    inet 10.106.2.35/24 brd 10.106.2.255 scope global dynamic noprefixroute eth0
       valid_lft 667701sec preferred_lft 667701sec
    inet6 fe80::d655:b9ff:f944:e435/64 scope link noprefixroute
       valid_lft forever preferred_lft forever
[cloud-user@rhel9-vm1 ~]$
[cloud-user@rhel9-vm1 ~]$ cat ~/.ssh/authorized_keys
ssh-ed25519 AAAAC3NzaC1lZDI1NTE5AAAAIMCCvPMHJpW9mlcqsYRMYwiMuiu3X0UY6AsDVNN4NH0A root@aa06-rhel9
[cloud-user@rhel9-vm1 ~]$
```

Step 15. Repeat steps 1 through 14 to create CentOS and Fedora linux VMs.



Project: default

VirtualMachines

Name	Status	Conditions	Node	Created	IP address
VM centos9-vm1	Running	DataVolumesReady=True	N worker1.fs-ocp1.flashstack.local	Oct 18, 2024, 12:12 PM	10.106.2.33 +1 more
VM rhel9-vm1	Running	DataVolumesReady=True	N worker2.fs-ocp1.flashstack.local	Oct 18, 2024, 11:49 AM	10.106.2.35 +1 more

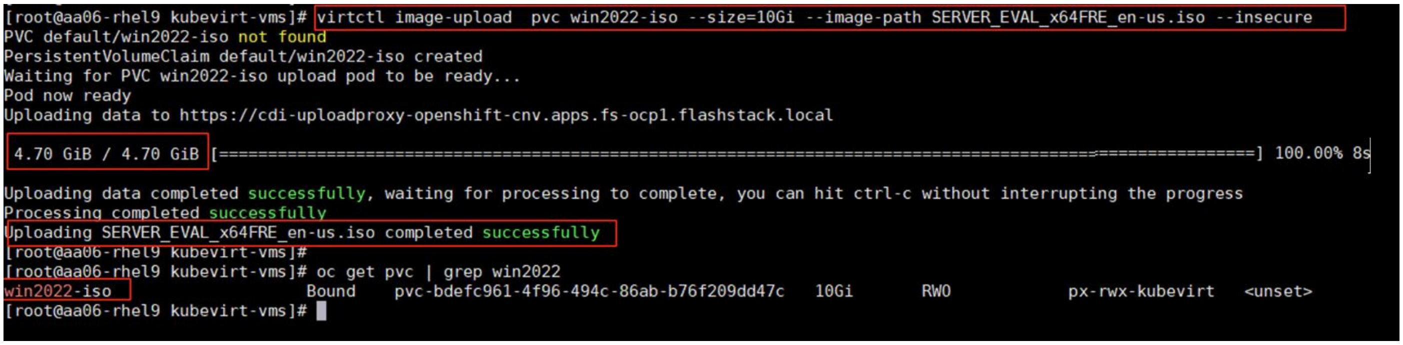
Procedure 4. Create Windows Template and provisioning Windows VMs from Template

For Windows Server Operating System, qcow2 image is not available. We can generate KVM compatible image from Windows Server ISO which can be downloadable from Microsoft website.

This section provides detailed steps to create a Windows VM from Windows Server ISO image and then configure the VM to directly access the Pure Storage FlashArray volumes using In-Guest iSCSI.

Follow this procedure to create a temporary windows Server 2022 virtual machine using Windows Server ISO, then install and configure the VM with all the required software components. Use this VM to create sysprep image. Then use this sysprep image as gold image for all the future windows Server 2022 VMs.

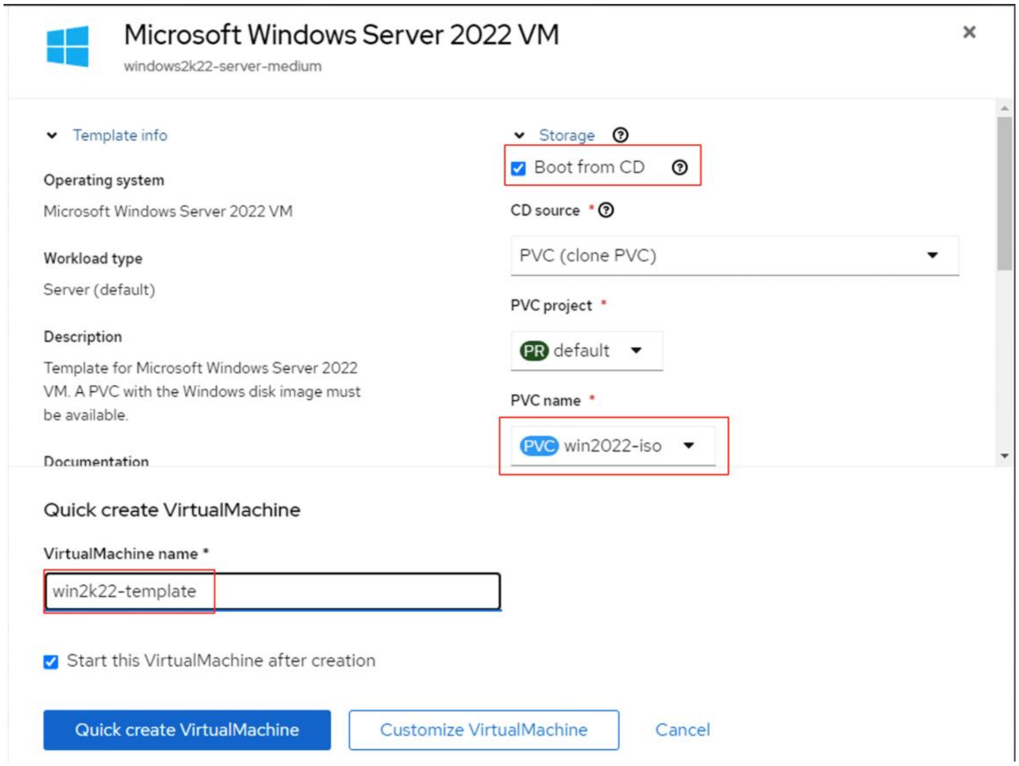
Step 1. Download the **Windows Server 2022 ISO** and upload to a PVC using the **Upload Data to PVC** option in the console or use virtctl utility to do so. The following screenshot shows uploading window server 2022 ISO image to wind2022-iso PVC using virtctl.



```
[root@aa06-rhel9 kubevirt-vms]# virtctl image-upload  pvc win2022-iso --size=10Gi --image-path SERVER_EVAL_x64FRE_en-us.iso --insecure
PVC default/win2022-iso not found
PersistentVolumeClaim default/win2022-iso created
Waiting for PVC win2022-iso upload pod to be ready...
Pod now ready
Uploading data to https://cdi-uploadproxy-openshift-cnv.apps.fs-ocp1.flashstack.local

 4.70 GiB / 4.70 GiB [==========================================================================================================] 100.00% 8s

Uploading data completed successfully, waiting for processing to complete, you can hit ctrl-c without interrupting the progress
Processing completed successfully
Uploading SERVER_EVAL_x64FRE_en-us.iso completed successfully
[root@aa06-rhel9 kubevirt-vms]#
[root@aa06-rhel9 kubevirt-vms]# oc get pvc | grep win2022
win2022-iso                    Bound    pvc-bdefc961-4f96-494c-86ab-b76f209dd47c   10Gi       RWO            px-rwx-kubevirt   <unset>
[root@aa06-rhel9 kubevirt-vms]#
```

Step 2. Using win2022-iso PVC as boot volume, create a temporary windows VM. Go to **Virtualization > VirtualMachines > Create > Using template**. Select **Windows Server 2022**.

Step 3. On the Create VM window, turn on the Boot From CD check bot and select the **win2022-iso** PVC as shown below.



Microsoft Windows Server 2022 VM

windows2k22-server-medium

Template info

Operating system
Microsoft Windows Server 2022 VM

Workload type
Server (default)

Description
Template for Microsoft Windows Server 2022 VM. A PVC with the Windows disk image must be available.

Documentation

Storage

☑ Boot from CD

CD source *
PVC (clone PVC)

PVC project *
PR default

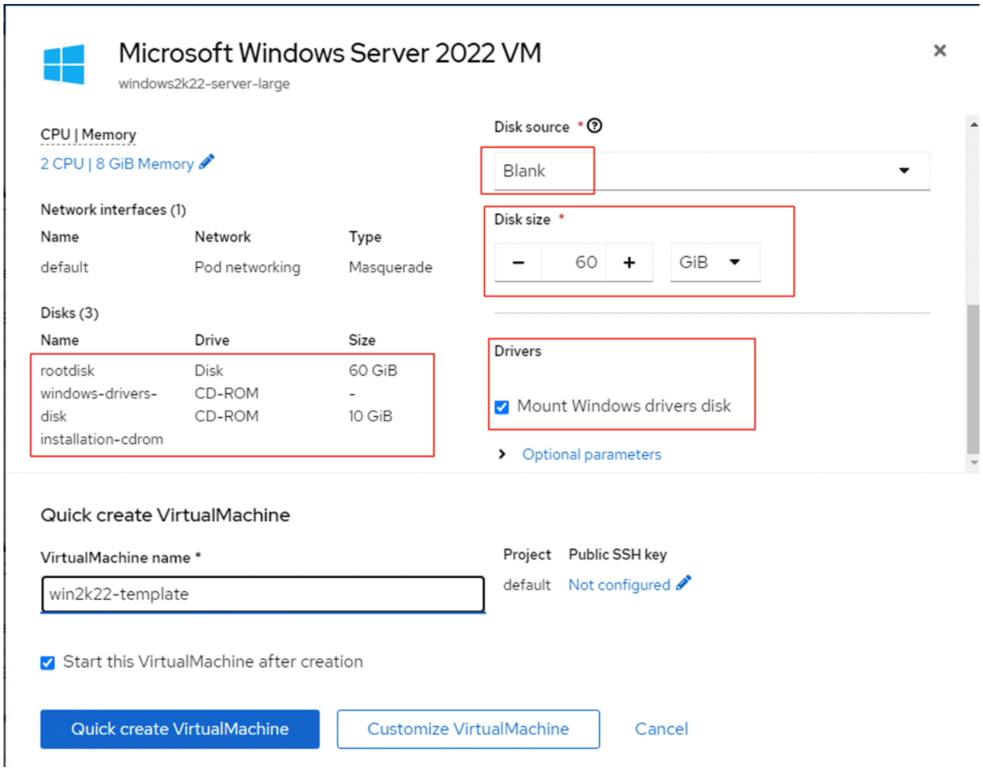
PVC name *
PVC win2022-iso

Quick create VirtualMachine

VirtualMachine name *
win2k22-template

☑ Start this VirtualMachine after creation

Step 4. Scroll down the scroll bar and Set Blank for Data Source and change the Disk size to from default 30 to 60GiB. Ensure the Windows Drivers check box is selected. Click **Customize VirtualMachine**.



Microsoft Windows Server 2022 VM

windows2k22-server-large

CPU | Memory
2 CPU | 8 GiB Memory

Network interfaces (1)

Name	Network	Type
default	Pod networking	Masquerade

Disks (3)

Name	Drive	Size
rootdisk	Disk	60 GiB
windows-drivers-disk	CD-ROM	-
installation-cdrom	CD-ROM	10 GiB

Disk source *
Blank

Disk size *
− 60 + GiB

Drivers
☑ Mount Windows drivers disk

Optional parameters

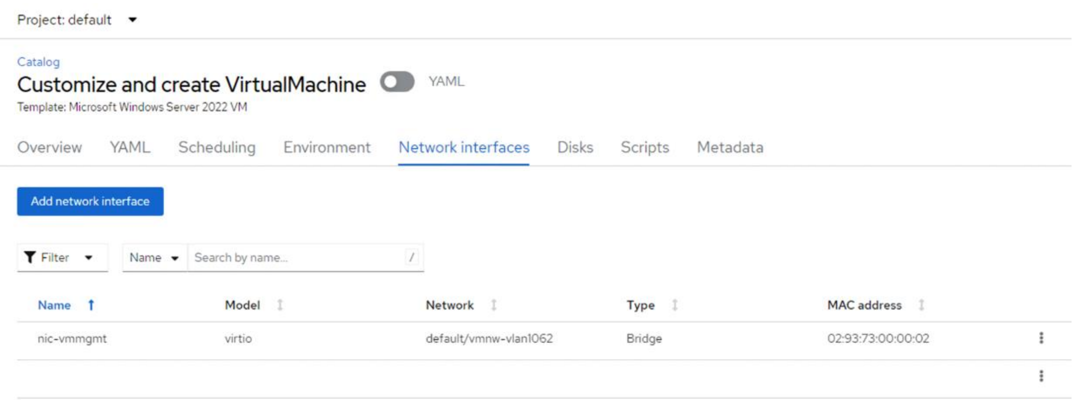
Quick create VirtualMachine

VirtualMachine name *
win2k22-template

Project: default
Public SSH key: Not configured

☑ Start this VirtualMachine after creation

Step 5. From the Network Interface tab, remove the preconfigured Pod network interface and add new interface on vmnw-vlan1062 network as shown below:

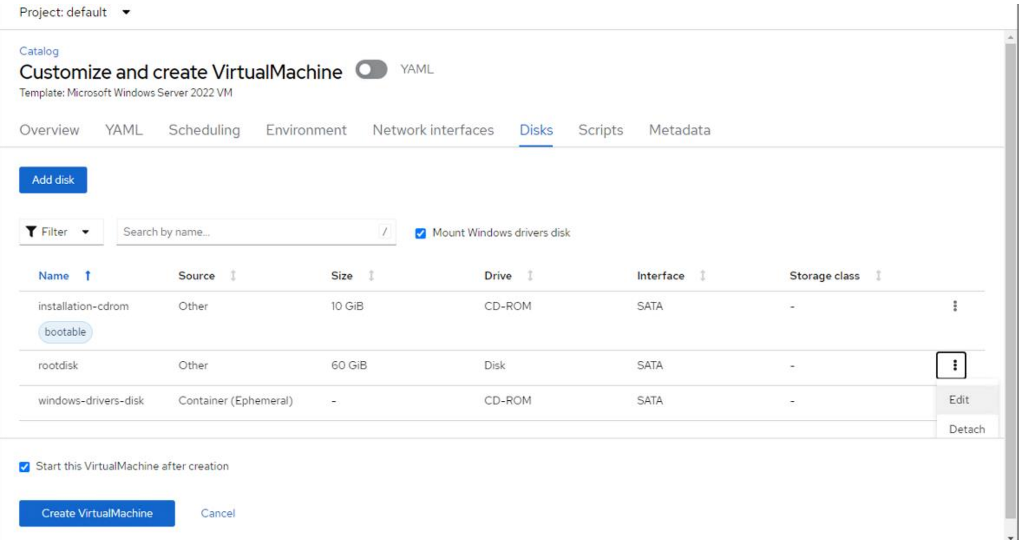


Customize and create VirtualMachine

Template: Microsoft Windows Server 2022 VM

Name	Model	Network	Type	MAC address
nic-vmmgmt	virtio	default/vmnw-vlan1062	Bridge	02:93:73:00:00:02

Step 6. From the Disks tab, click the ellipses of the rootdisk and set the StorageClass to **px-sharedv4-kubevirt**. Click **Create VirtualMachine**.



Customize and create VirtualMachine

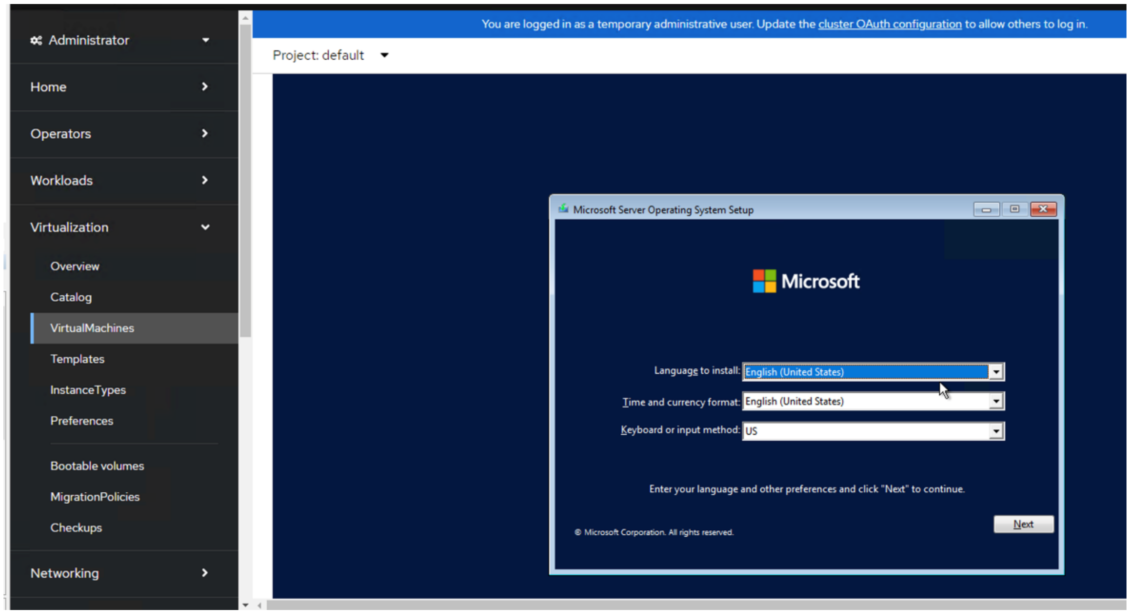
Template: Microsoft Windows Server 2022 VM

☑ Mount Windows drivers disk

Name	Source	Size	Drive	Interface	Storage class
installation-cdrom bootable	Other	10 GiB	CD-ROM	SATA	-
rootdisk	Other	60 GiB	Disk	SATA	-
windows-drivers-disk	Container (Ephemeral)	-	CD-ROM	SATA	-

☑ Start this VirtualMachine after creation

Step 7. When Widows VM started, press any key to start the Windows Server 2022 installation.



You are logged in as a temporary administrative user. Update the cluster OAuth configuration to allow others to log in.

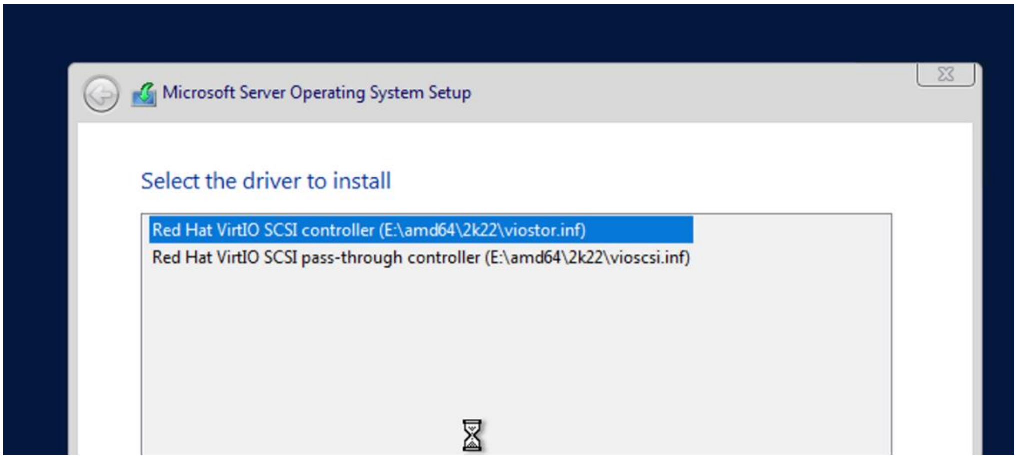
Language to install: English (United States)

Time and currency format: English (United States)

Keyboard or input method: US

Enter your language and other preferences and click "Next" to continue.

Step 8. Windows ISO cannot detect the rootdisk (60G) due to missing OpenShift virtio drivers. Click **Browse** to the mounted virtio drivers and select 2k22 folder. Click **OK**. Then Select Red Hat VirtIO SCSI Controller and click **OK** again.

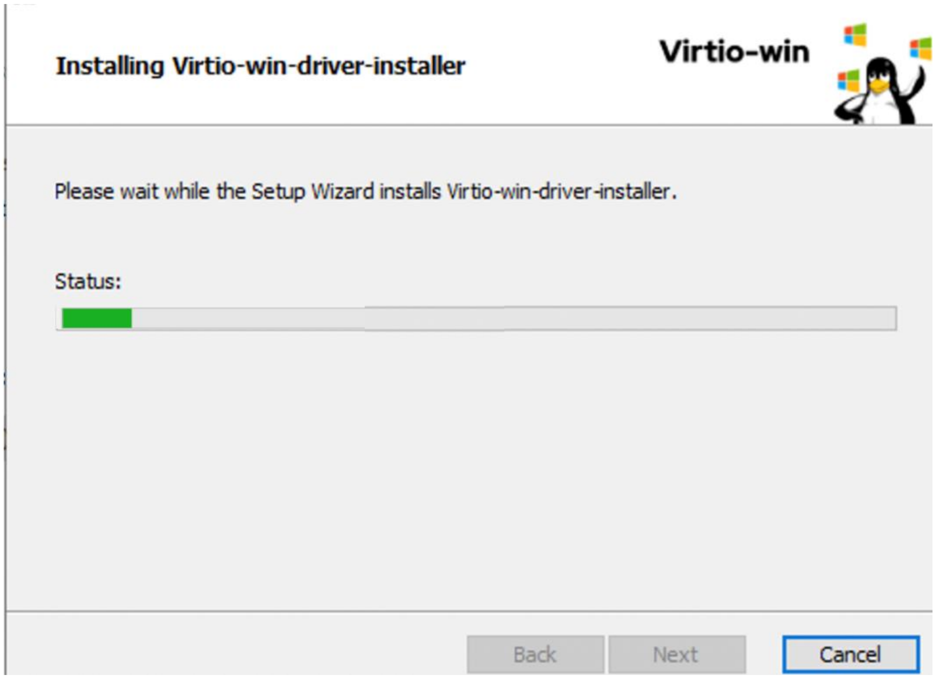


Select the driver to install

Red Hat VirtIO SCSI controller (E:\amd64\2k22\viostor.inf)
Red Hat VirtIO SCSI pass-through controller (E:\amd64\2k22\vioscsi.inf)

Step 9. Windows ISO media detects the rootdisk. Select the disk and complete Windows Server installation.

Step 10. When the Windows is installed successfully, go to E:\ disk and install the virtio driver by double-clicking the virtio-win-gt-x64.msi. Complete the drivers installation with default options.



Installing Virtio-win-driver-installer

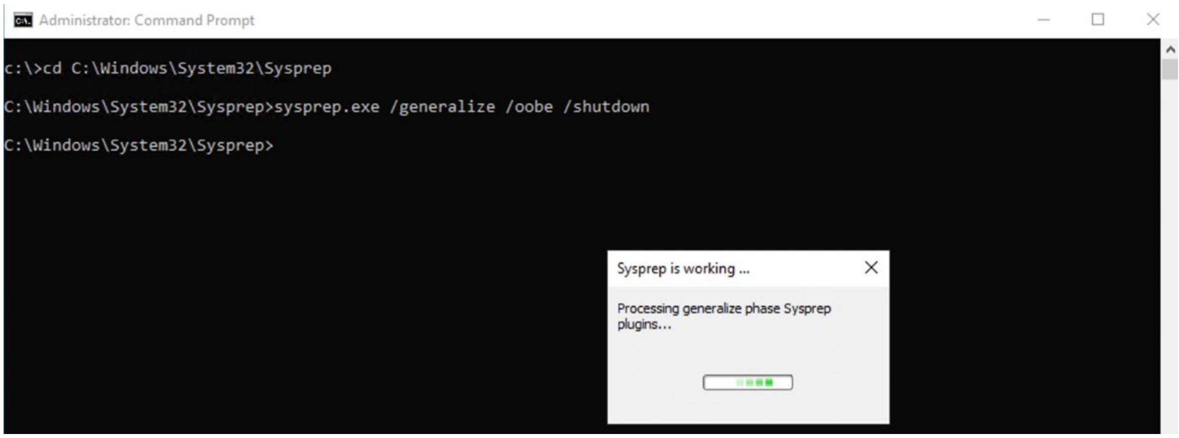
Please wait while the Setup Wizard installs Virtio-win-driver-installer.

Status:

Step 11. When the drivers are installed, the network interfaces will come up and get assigned with DHCP IP address. Turn off the firewalls temporarily to check if the VM can reach the outside services like AD/DNS. Enable Remote Desktop.

Note: You can install the required software, tools and drivers like MPIO before the VM is converted into an gold image.

Step 12. Convert this temporary VM into sysprep image by executing the sysprep command as shown below. Once sysprep is completed, the VM tries to restart. Before it restarts, stop the VM from OpenShift console.



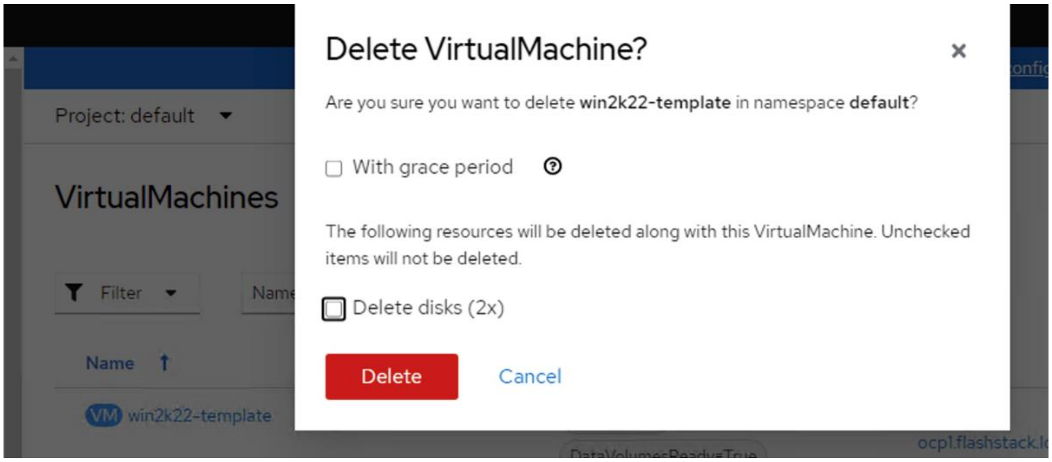
Administrator: Command Prompt

```
c:\>cd C:\Windows\System32\Sysprep

C:\Windows\System32\Sysprep>sysprep.exe /generalize /oobe /shutdown

C:\Windows\System32\Sysprep>
```

Sysprep is working ...

Processing generalize phase Sysprep plugins...

Step 13. Delete this temporary VM and ensure you DO NOT delete the PVCs by deselecting the checkbox as shown below.



Delete VirtualMachine?

Are you sure you want to delete **win2k22-template** in namespace **default**?

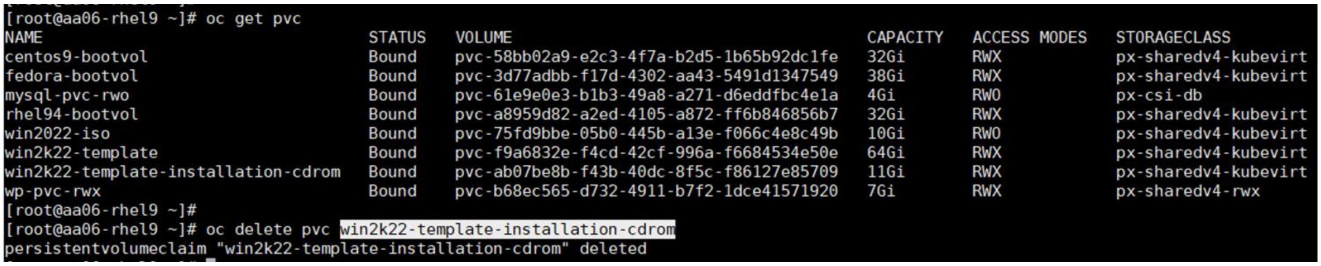
☐ With grace period

The following resources will be deleted along with this VirtualMachine. Unchecked items will not be deleted.

☐ Delete disks (2x)

Delete Cancel

Step 14. Optionally, delete the PVC CDROM disk which was created during the temporary VM creation.

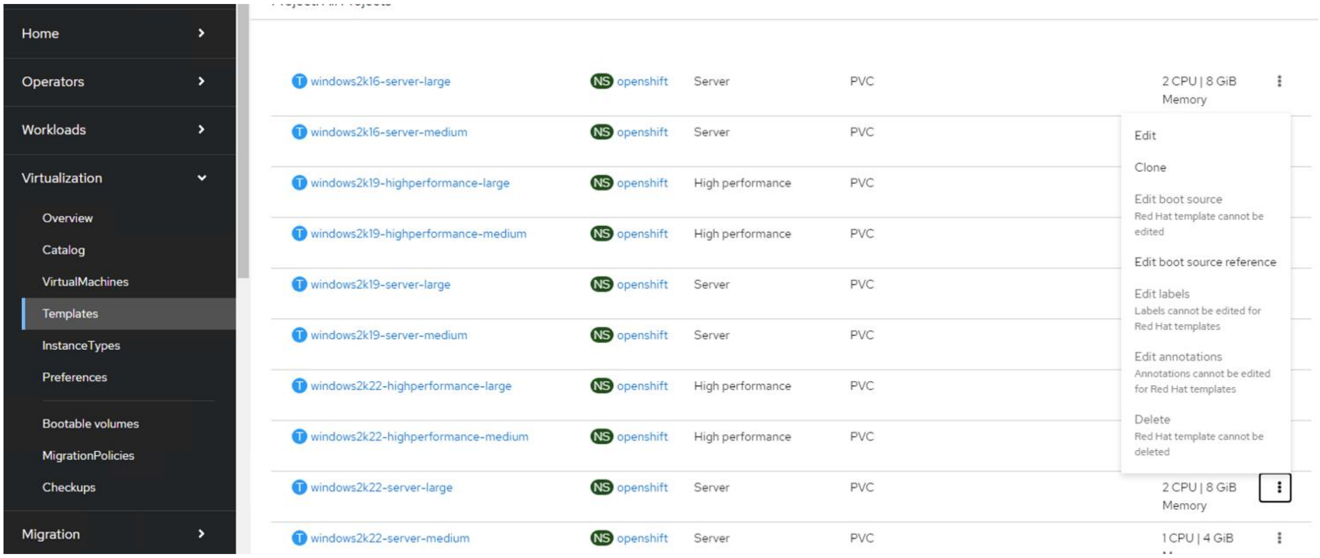


```
[root@aa06-rhel9 ~]# oc get pvc
NAME                               STATUS   VOLUME                                     CAPACITY   ACCESS MODES   STORAGECLASS
centos9-bootvol                    Bound    pvc-58bb02a9-e2c3-4f7a-b2d5-1b65b92dc1fe   32Gi       RWX            px-sharedv4-kubevirt
fedora-bootvol                     Bound    pvc-3d77adbb-f17d-4302-aa43-5491d1347549   38Gi       RWX            px-sharedv4-kubevirt
mysql-pvc-rwo                      Bound    pvc-61e9e0e3-b1b3-49a8-a271-d6eddfbc4e1a   4Gi        RWO            px-csi-db
rhel94-bootvol                     Bound    pvc-a8959d82-a2ed-4105-a872-ff6b846856b7   32Gi       RWX            px-sharedv4-kubevirt
win2022-iso                        Bound    pvc-75fd9bbe-05b0-445b-a13e-f066c4e8c49b   10Gi       RWO            px-sharedv4-kubevirt
win2k22-template                   Bound    pvc-f9a6832e-f4cd-42cf-996a-f6684534e50e   64Gi       RWX            px-sharedv4-kubevirt
win2k22-template-installation-cdrom Bound   pvc-ab07be8b-f43b-40dc-8f5c-f86127e85709   11Gi       RWX            px-sharedv4-kubevirt
wp-pvc-rwx                         Bound    pvc-b68ec565-d732-4911-b7f2-1dce41571920   7Gi        RWX            px-sharedv4-rwx
[root@aa06-rhel9 ~]#
[root@aa06-rhel9 ~]# oc delete pvc win2k22-template-installation-cdrom
persistentvolumeclaim "win2k22-template-installation-cdrom" deleted
```

Now you can use the remaining PVC (win-2k22-template) which has syspred windows server 2022 gold image to map to an existing Windows Server 2022 template or you can create a new template from the existing windows Server 2022 template.

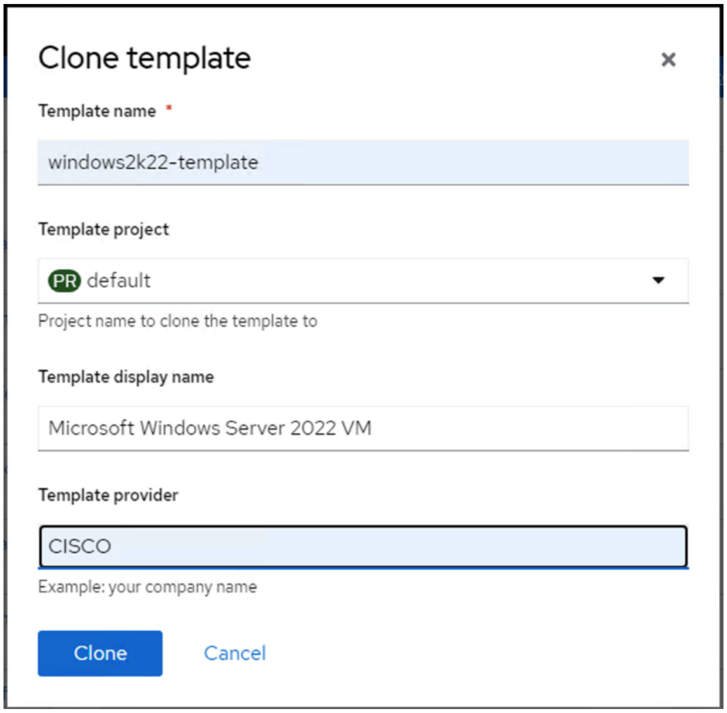
Step 15. To create a new custom Windows Server 2022 template from the existing templates, go to **Virtualization > Template > All projects**.

Step 16. Select the existing Windows Server 2022 template and click the ellipses and click **Clone**.



windows2k16-server-large	NS openshift	Server	PVC	2 CPU | 8 GiB Memory
windows2k16-server-medium	NS openshift	Server	PVC	
windows2k19-highperformance-large	NS openshift	High performance	PVC	
windows2k19-highperformance-medium	NS openshift	High performance	PVC	
windows2k19-server-large	NS openshift	Server	PVC	
windows2k19-server-medium	NS openshift	Server	PVC	
windows2k22-highperformance-large	NS openshift	High performance	PVC	
windows2k22-highperformance-medium	NS openshift	High performance	PVC	
windows2k22-server-large	NS openshift	Server	PVC	2 CPU | 8 GiB Memory
windows2k22-server-medium	NS openshift	Server	PVC	1 CPU | 4 GiB

Step 17. From the Clone Template windows, provide a name for the template (windows2k22-template), change the project to default and provide a name for Template Provider and click **Clone**.



Clone template

Template name *

windows2k22-template

Template project

PR default

Project name to clone the template to

Template display name

Microsoft Windows Server 2022 VM

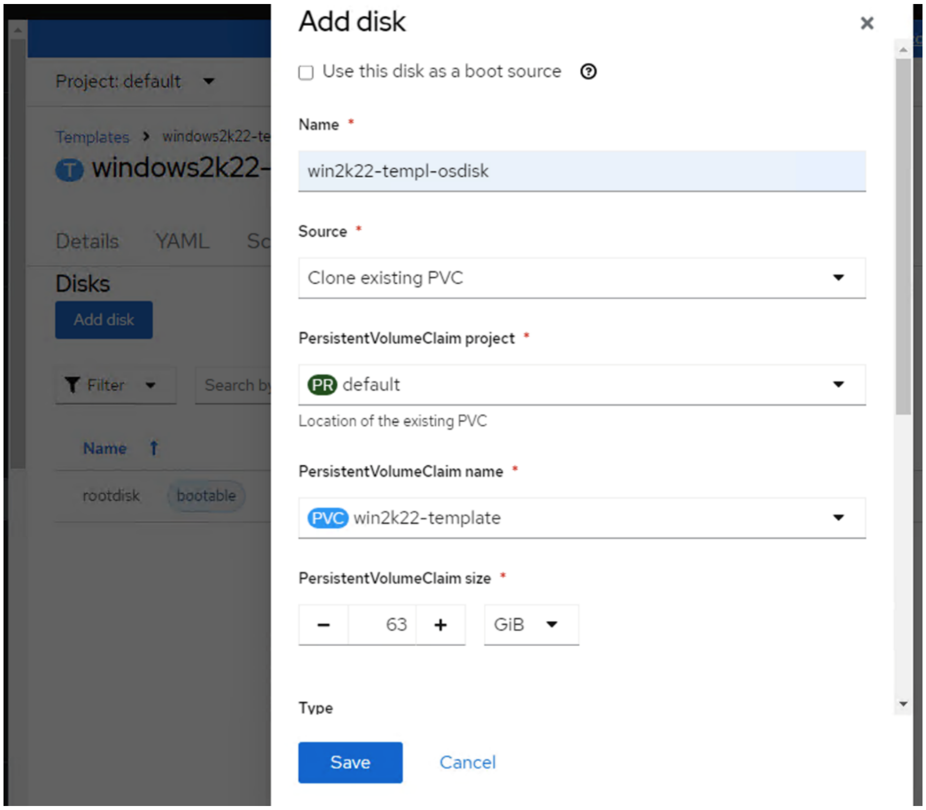
Template provider

CISCO

Example: your company name

Step 18. Go to this newly created windows2k22-template custom template, set boot source volume to the PVC which has windows sysprep image.

Step 19. Go to **Virtualization > Templates > Default Project**, click the template, go to the Disk tab and click **Add disk** to add a new disk. Provide a name for the disk and select the PVC (Win2k22-template) that has windows sysprep image as shown below.



Add disk

□ Use this disk as a boot source

Name *

win2k22-templ-osdisk

Source *

Clone existing PVC

PersistentVolumeClaim project *

PR default

Location of the existing PVC

PersistentVolumeClaim name *

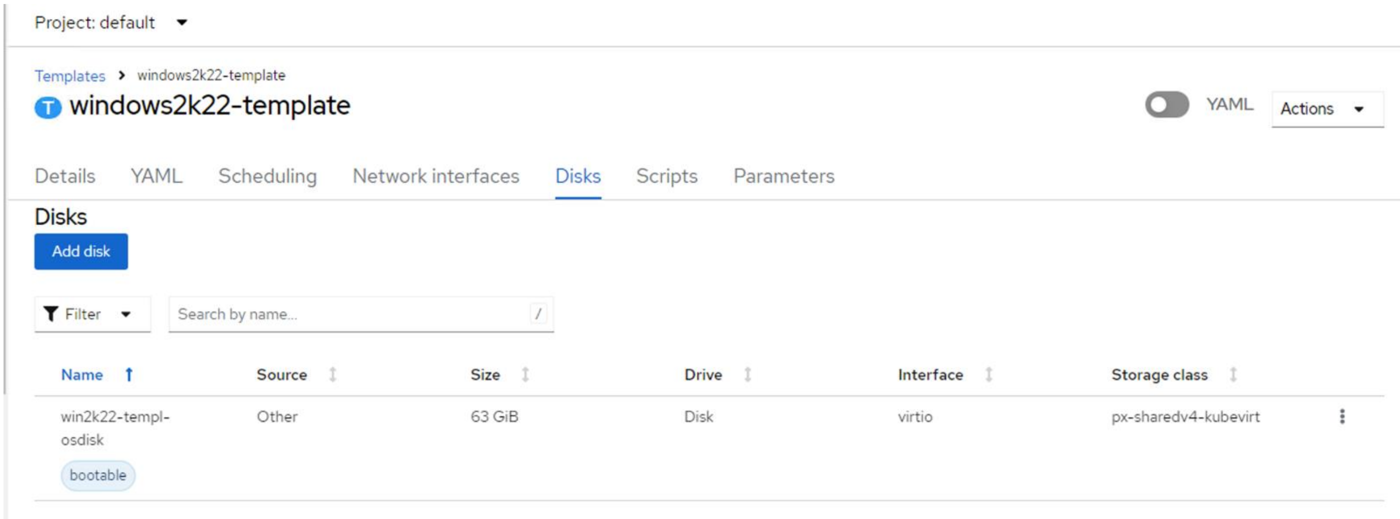
PVC win2k22-template

PersistentVolumeClaim size *

63 GiB

Type

Step 20. Delete the existing boot disk and set the win2k22-templ-osdisk to boot disk.



Project: default

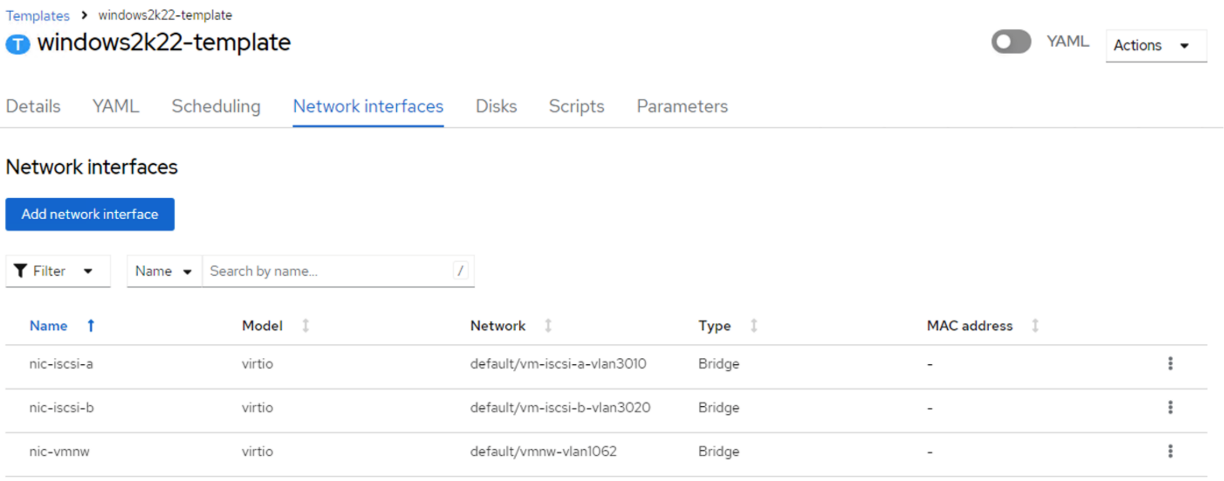
Templates > windows2k22-template

windows2k22-template

Disks

Name	Source	Size	Drive	Interface	Storage class
win2k22-templ-osdisk bootable	Other	63 GiB	Disk	virtio	px-sharedv4-kubevirt

Step 21. Go to **Network Interfaces**, add two network interfaces for iSCSI traffic as shown below. The Management interface is already added in the previous steps.



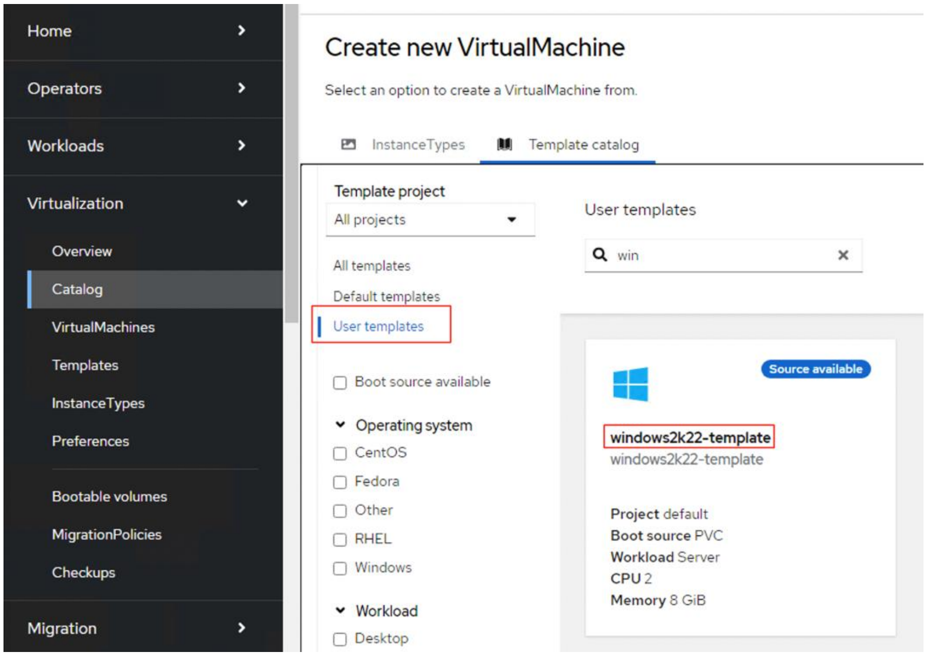
Network interfaces

Name	Model	Network	Type	MAC address
nic-iscsi-a	virtio	default/vm-iscsi-a-vlan3010	Bridge	-
nic-iscsi-b	virtio	default/vm-iscsi-b-vlan3020	Bridge	-
nic-vmnw	virtio	default/vmnw-vlan1062	Bridge	-

This template is configured to boot from the sysprep image, and also configured with a total of three interfaces. One for management and two for iSCSI traffic.

Step 22. Create a fresh Windows Server 2022 virtual machine using windows2k22-template template.

Step 23. Go to **Virtualization > VirtualMachines > Create > From template**. Click **User templates.** Select the new Windows2k22-template.



Create new VirtualMachine

Select an option to create a VirtualMachine from.

Template project: All projects

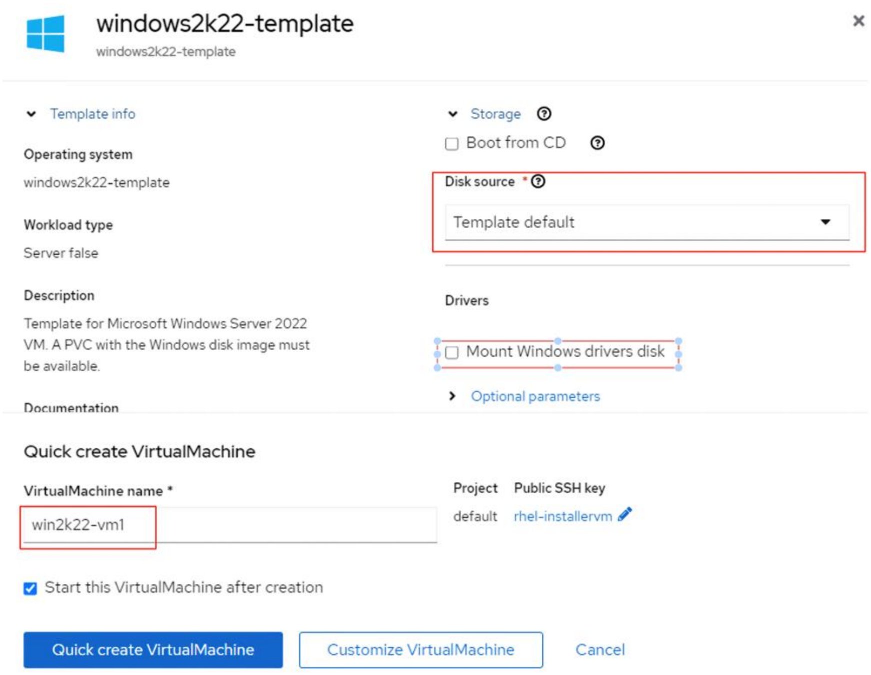
User templates

win

windows2k22-template
windows2k22-template

Project default
Boot source PVC
Workload Server
CPU 2
Memory 8 GiB

Step 24. Provide a name for the VM and click **Quick Create VirtualMachine**. Since the template is pre-configured with everything according to our requirements, nothing has to be changed to create the virtual machine. The VMs displays in seconds.



windows2k22-template

windows2k22-template

Template info

Operating system
windows2k22-template

Workload type
Server false

Description
Template for Microsoft Windows Server 2022 VM. A PVC with the Windows disk image must be available.

Documentation

Storage

□ Boot from CD

Disk source *
Template default

Drivers

□ Mount Windows drivers disk

Optional parameters

Quick create VirtualMachine

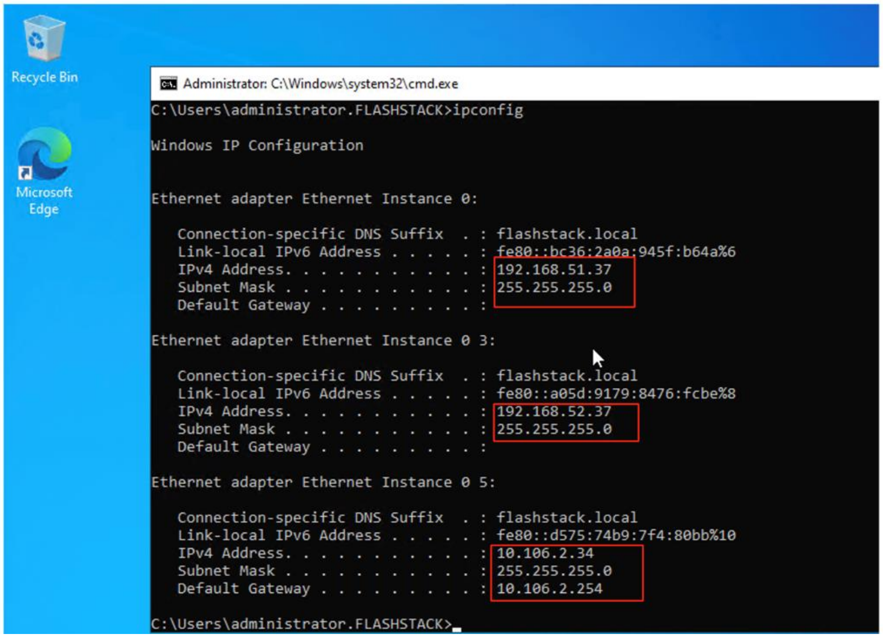
VirtualMachine name *
win2k22-vm1

Project
default

Public SSH key
rhel-installervm

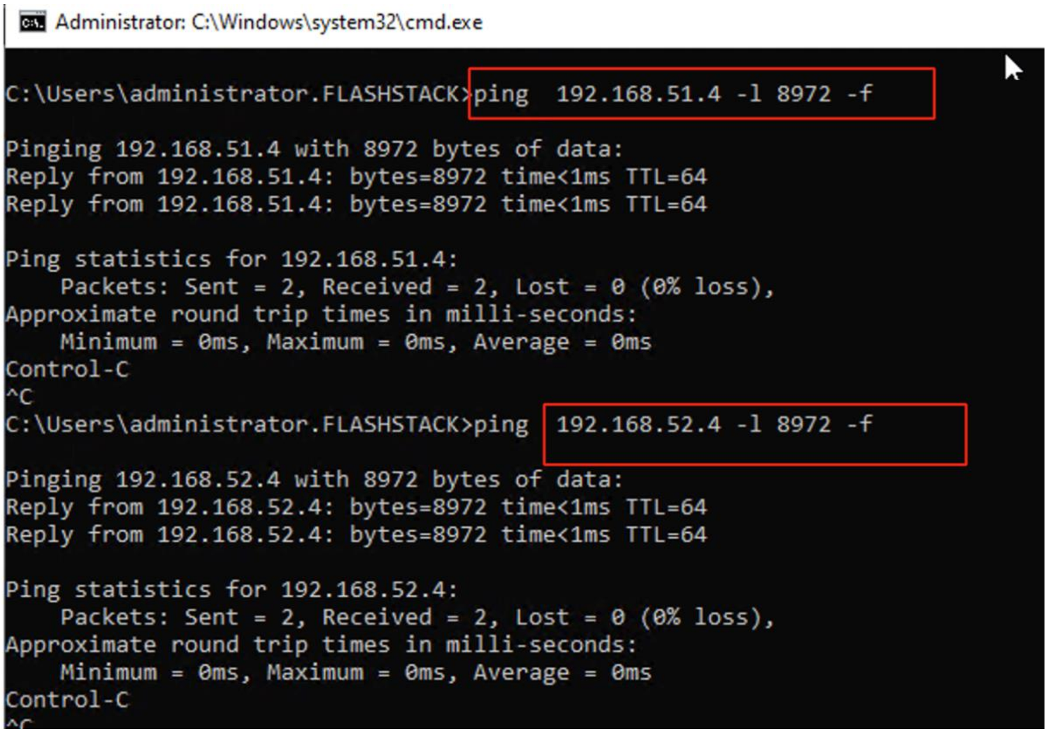
☑ Start this VirtualMachine after creation

Step 25. When is VM is fully provisioned, verify all the three interfaces get corresponding DHCP IP address.



```
C:\Users\administrator.FLASHSTACK>ipconfig

Windows IP Configuration


Ethernet adapter Ethernet Instance 0:

   Connection-specific DNS Suffix  . : flashstack.local
   Link-local IPv6 Address . . . . . : fe80::bc36:2a0a:945f:b64a%6
   IPv4 Address. . . . . . . . . . . : 192.168.51.37
   Subnet Mask . . . . . . . . . . . : 255.255.255.0
   Default Gateway . . . . . . . . . :

Ethernet adapter Ethernet Instance 0 3:

   Connection-specific DNS Suffix  . : flashstack.local
   Link-local IPv6 Address . . . . . : fe80::a05d:9179:8476:fcbe%8
   IPv4 Address. . . . . . . . . . . : 192.168.52.37
   Subnet Mask . . . . . . . . . . . : 255.255.255.0
   Default Gateway . . . . . . . . . :

Ethernet adapter Ethernet Instance 0 5:

   Connection-specific DNS Suffix  . : flashstack.local
   Link-local IPv6 Address . . . . . : fe80::d575:74b9:7f4:80bb%10
   IPv4 Address. . . . . . . . . . . : 10.106.2.34
   Subnet Mask . . . . . . . . . . . : 255.255.255.0
   Default Gateway . . . . . . . . . : 10.106.2.254

C:\Users\administrator.FLASHSTACK>_
```

Now you should reach the Pure Storage FlashArray target IP addresses with large packet size without fragmenting the packet since the Jumbo Frames/MTU is already configured on the iSCSI interfaces.



```
C:\Users\administrator.FLASHSTACK>ping  192.168.51.4 -l 8972 -f

Pinging 192.168.51.4 with 8972 bytes of data:
Reply from 192.168.51.4: bytes=8972 time<1ms TTL=64
Reply from 192.168.51.4: bytes=8972 time<1ms TTL=64

Ping statistics for 192.168.51.4:
    Packets: Sent = 2, Received = 2, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 0ms, Maximum = 0ms, Average = 0ms
Control-C
^C
C:\Users\administrator.FLASHSTACK>ping  192.168.52.4 -l 8972 -f

Pinging 192.168.52.4 with 8972 bytes of data:
Reply from 192.168.52.4: bytes=8972 time<1ms TTL=64
Reply from 192.168.52.4: bytes=8972 time<1ms TTL=64

Ping statistics for 192.168.52.4:
    Packets: Sent = 2, Received = 2, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 0ms, Maximum = 0ms, Average = 0ms
Control-C
^C
```

Step 26. To directly access the volume created in the Pure Storage FlashArray, run the following PowerShell commands to start iSCSI service and connect to the Pure Storage FlashArray directly.

Note: You would need to connect to the Pure Storage FlashArray and create a host for this Windows Guest VM using its IQN and assign a volume to the newly created host.

```
## open PowerShell with Administrator right and execute the following steps.
Start-Service -Name MSiSCSI
Set-Service -Name MSiSCSI -StartupType Automatic
(Get-InitiatorPort).NodeAddress

New-IscsiTargetPortal -TargetPortalAddress 192.168.51.4 -InitiatorPortalAddress 192.168.51.37
New-IscsiTargetPortal -TargetPortalAddress 192.168.51.5 -InitiatorPortalAddress 192.168.51.37
New-IscsiTargetPortal -TargetPortalAddress 192.168.52.4 -InitiatorPortalAddress 192.168.52.37
New-IscsiTargetPortal -TargetPortalAddress 192.168.52.5 -InitiatorPortalAddress 192.168.52.37

## Now log into FlashArray and create a host for this Guest using its IQN, create a vlume and assign the
volume to this host.

Create Host for Guest VM, create volume and connect to it
$target = Get-IscsiTarget
Connect-IscsiTarget -TargetPortalAddress 192.168.51.4 -InitiatorPortalAddress 192.168.51.37 -NodeAddress
$target.NodeAddress -IsMultipathEnabled $true -IsPersistent $true
Connect-IscsiTarget -TargetPortalAddress 192.168.51.5 -InitiatorPortalAddress 192.168.51.37 -NodeAddress
$target.NodeAddress -IsMultipathEnabled $true -IsPersistent $true
Connect-IscsiTarget -TargetPortalAddress 192.168.52.4 -InitiatorPortalAddress 192.168.52.37 -NodeAddress
$target.NodeAddress -IsMultipathEnabled $true -IsPersistent $true
Connect-IscsiTarget -TargetPortalAddress 192.168.52.5 -InitiatorPortalAddress 192.168.52.37 -NodeAddress
$target.NodeAddress -IsMultipathEnabled $true -IsPersistent $true

## shows the FlashArray disk
Get-Disk

Number Friendly Name Serial Number                    HealthStatus         OperationalStatus      Total Size
Partition
Style
------ ------------- -------------                    ------------         -----------------      ----------
----------
1      PURE Flash... 59471BE632AA4FD20001C691         Healthy              Offline                    110 GB
GPT
```

```
0       Red Hat Vi...                                    Healthy            Online                    63 GB
GPT
```

Live Migration of VMs within the OpenShift Cluster

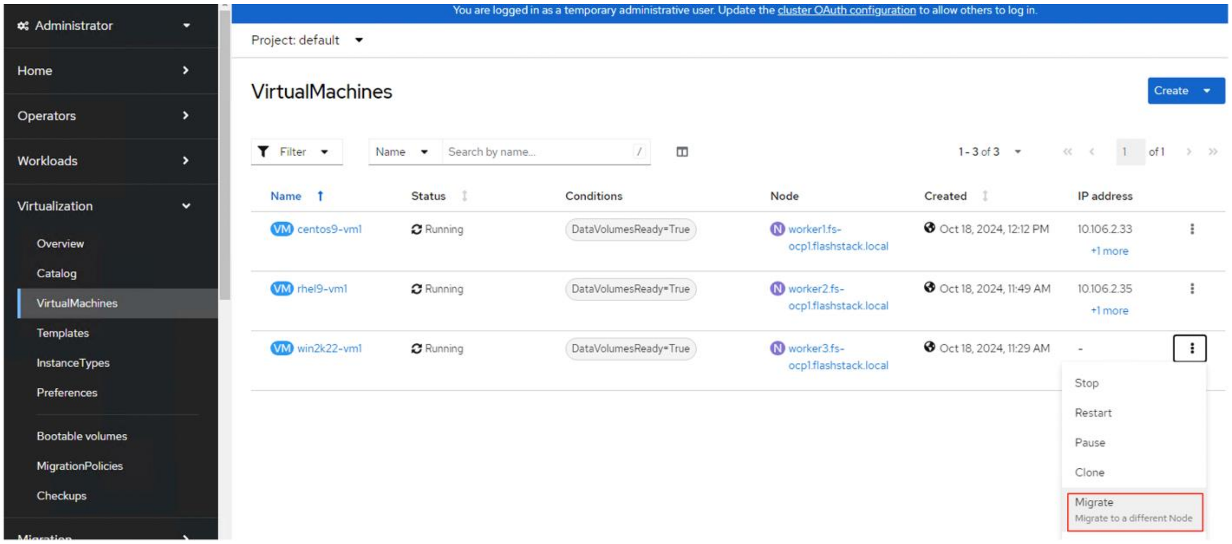
Live migration of VMs is non disruptive process of moving VMs from one worker node to other without downtime and has the following requirements:

- The OCP cluster must have shared storage with ReadWriteMany (RWX) access mode. An OCP cluster backed by Pure Storage FlashArray and Portworx Enterprise CSI driver already supports RWX access mode for filesystem. Any VM created with a StorageClass provisioned by Portworx already uses RWX PVCs and can be live migrated without downtime.
- The default number of migrations that can run in parallel in the cluster is 5, with a maximum of two 2 migrations per node. To change these settings, go to Virtualization > Overview > Settings > General Settings > Live Migration and change the settings according to your requirements.
- Using a dedicated network for Live migration is optional but recommended. Go to Virtualization > Overview > Settings > General Settings > Live Migration > Live migration network and select the required interface for live migration of VMs.

The virtual machines can be live migrated from one worker to other by selecting VM and click the ellipsis (three dots) and select the Migrate option.

A live migration can be triggered from the GUI, CLI, API, or automatically.

You can monitor the VM migration status using the command: oc describe vmi <vm_name> -n <namespace>



You are logged in as a temporary administrative user. Update the cluster OAuth configuration to allow others to log in.

Project: default

VirtualMachines

Name	Status	Conditions	Node	Created	IP address
VM centos9-vm1	Running	DataVolumesReady=True	N worker1.fs-ocp1.flashstack.local	Oct 18, 2024, 12:12 PM	10.106.2.33 +1 more
VM rhel9-vm1	Running	DataVolumesReady=True	N worker2.fs-ocp1.flashstack.local	Oct 18, 2024, 11:49 AM	10.106.2.35 +1 more
VM win2k22-vm1	Running	DataVolumesReady=True	N worker3.fs-ocp1.flashstack.local	Oct 18, 2024, 11:29 AM	-

Migrate Virtual Machines from the VMware vSphere Cluster to OpenShift Virtualization

The Migration Toolkit for Virtualization (MTV) is an operator-based functionality that enables us to migrate virtual machines at scale to Red Hat OpenShift Virtualization. MTV supports migration of virtual machines from VMware vSphere, Red Hat Virtualization, OpenStack, OVA and OpenShift Virtualization source providers to OpenShift Virtualization.

The following are some of the vSphere prerequisites to be met before planning for migration of virtual machines from vSphere environments:

- VMware vSphere version must be compatible with OpenShift virtualization. At a minimum, vSphere 6.5 or later is compatible with OpenShift Virtualization 4.14 or later. For more details on compatibility, go to: https://docs.redhat.com/en/documentation/migration_toolkit_for_virtualization/2.6/html-single/installing_and_using_the_migration_toolkit_for_virtualization/index#compatibility-guidelines_mtv
- Only the specific Guest Operating Systems can be migrated to OpenShift Virtualization. Here is the list of guest operating systems supported by OpenShift Virtualization: https://access.redhat.com/articles/973163?extIdCarryOver=true&sc_cid=701f2000001OH7JAAW#ocpvirt
- The guest OS must be supported by virt-v2v utility to convert them into OpenShift virtualization compatible images as listed here: https://access.redhat.com/articles/1351473
- Must have a user with at least the minimal set of VMware privileges. Required privileges listed here: https://docs.redhat.com/en/documentation/migration_toolkit_for_virtualization/2.6/html-single/installing_and_using_the_migration_toolkit_for_virtualization/index#vmware-prerequisites_mtv
- The Secure boot option must be disabled on the VMs.
- For a warm migration, changed block tracking (CBT) must be enabled on the VMs and on the VM disks. Here are the steps for enabling CBT on the VMs running on vSphere cluster: https://knowledge.broadcom.com/external/article/320557/changed-block-tracking-cbt-on-virtual-ma.html
- The MTV can use the VMware Virtual Disk Development Kit (VDDK) SDK to accelerate transferring virtual disks from VMware vSphere. Optionally, VDDK can be used for the faster migration. Here are the steps to configure VDDK: https://docs.redhat.com/en/documentation/migration_toolkit_for_virtualization/2.6/html-single/installing_and_using_the_migration_toolkit_for_virtualization/index#creating-vddk-image_mtv

For more information, see: https://docs.redhat.com/en/documentation/migration_toolkit_for_virtualization/2.6/html-single/installing_and_using_the_migration_toolkit_for_virtualization/index#rhv-prerequisites_mtv

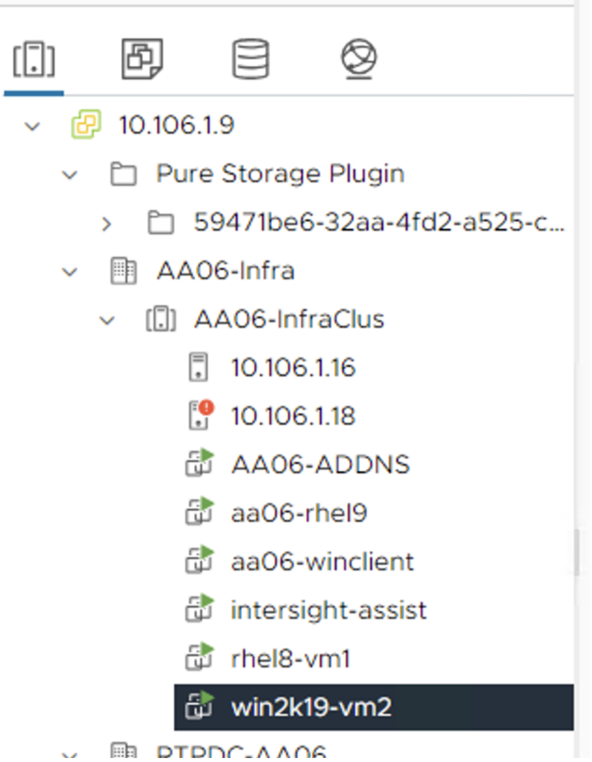
https://docs.redhat.com/en/documentation/migration_toolkit_for_virtualization/2.6/html-single/installing_and_using_the_migration_toolkit_for_virtualization/index#vmware-prerequisites_mtv

Note: The virtual machines with guest-initiated storage connections, such as Internet Small Computer Systems Interface (iSCSI) connections or Network File System (NFS) mounts, are not handled by MTV and could require additional planning before or reconfiguration after the migration.

This section describes the migration of the following two VMs running on vSphere 8.0 cluster:

VM Name	Number of Disks	Network Interfaces
Rhel8-vm1	OS Disk: 80GiB Data Disk: 90GiB	VMNW-VLAN1062
Win2k19-vm2	OS Disk: 90GiB Data Disk:20G (Guest Initiated using in-guest iscsi). MTV will ignore this disk	VMNW-VLAN1062 iSCSI-A iSCSI-B

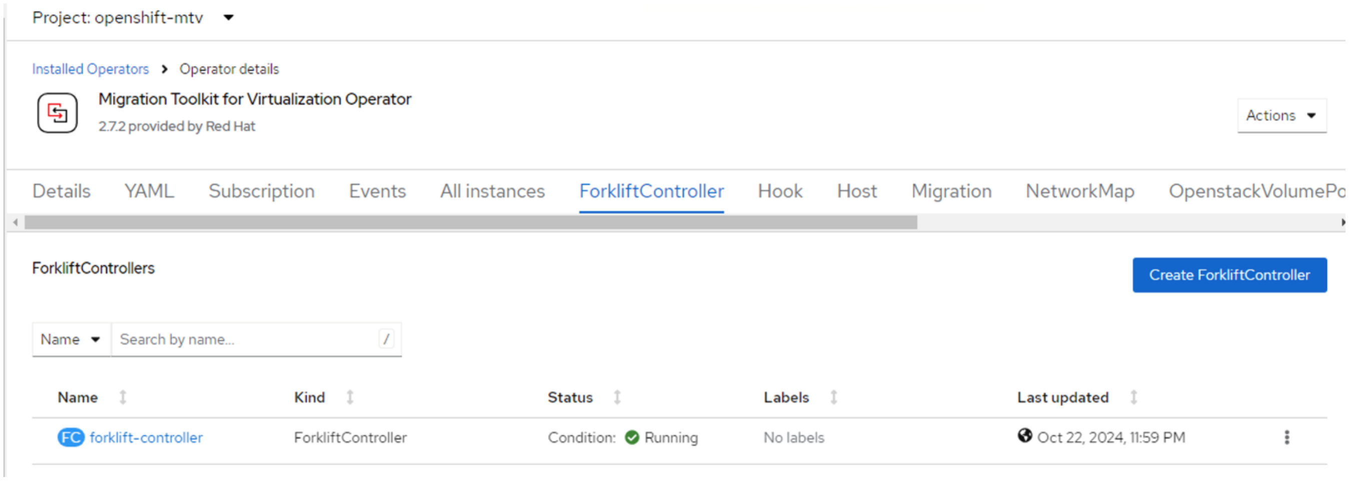
The following screenshot shows the vSphere cluster with above two VMs (rhe8-vm1 and win2k19-vm2) to be migrated to OpenShift cluster:



To migrate the VMs from vSphere cluster to OpenShift virtualization, following these procedures:

Procedure 1. Install the Migration Toolkit for Virtualization

Step 1. Install the MTV operator, go to **Operators > OperatorHub**, search for mtv, and select the **Migration Toolkit for Virtualization** operator and click **Install**. When the operator is installed successfully, click **Create ForkliftController** to install the ForkliftController. With the default option, complete the FortliftController installation. Refresh the browser to see the Migration tab on the console.



Migration Toolkit for Virtualization Operator
2.7.2 provided by Red Hat

ForkliftControllers

Name	Kind	Status	Labels	Last updated
FC forklift-controller	ForkliftController	Condition: ✓ Running	No labels	Oct 22, 2024, 11:59 PM

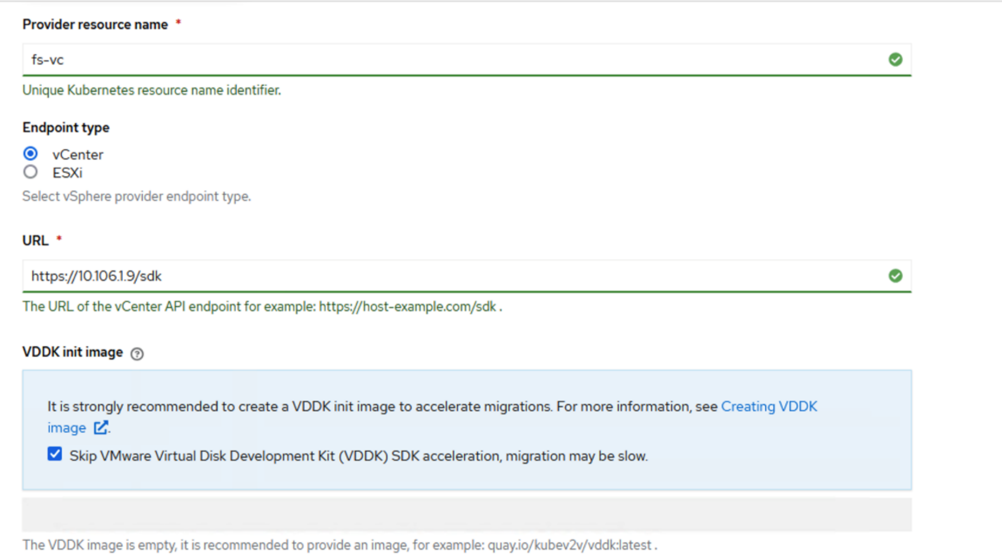
The actions to perform the migration of VMs from vSphere environment to OpenShift Virtualization are as follows:

- Identify the VMs on the vSphere environment and ensure the VMs are ready for migration by verifying that all the pre-requisites are met
- Create Provider
- Create Migration Plan
- Execute the migration plan

Procedure 2. Migrate VMs from vSphere to OpenShift Virtualization

Step 1. To create vSphere Provider, go to **Migration > Providers for Virtualization**, click **Create Provider**. Select the required project name.

Step 2. Click **vm vSphere.** Provide a name to the provider, URL to the provider sdk in the format https://host-example.com/sdk. You can either skip the VDDK or provide the repository path where the VDDK images is pushed.


Provider resource name *
fs-vc
Unique Kubernetes resource name identifier.

Endpoint type
● vCenter
○ ESXi
Select vSphere provider endpoint type.

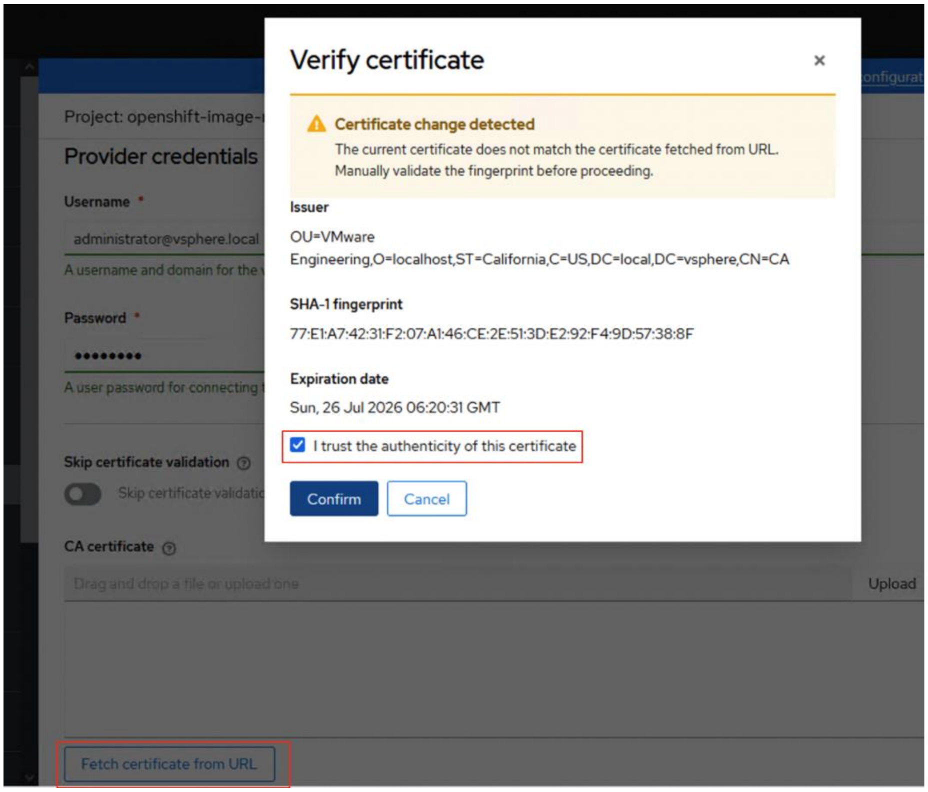
URL *
https://10.106.1.9/sdk
The URL of the vCenter API endpoint for example: https://host-example.com/sdk .

VDDK init image

It is strongly recommended to create a VDDK init image to accelerate migrations. For more information, see Creating VDDK image.
☑ Skip VMware Virtual Disk Development Kit (VDDK) SDK acceleration, migration may be slow.

The VDDK image is empty, it is recommended to provide an image, for example: quay.io/kubev2v/vddk:latest .

Step 3. Provide a vSphere user id and password details to use for the migration activity. Click **Fetch Certificate from URL** to generate the certificates and click **Confirm.** Click **Create Provider** to complete provider creation.


Project: openshift-image-
Provider credentials
Username *
administrator@vsphere.local
A username and domain for the
Password *
••••••••
A user password for connecting
Skip certificate validation
Skip certificate validatio
CA certificate
Drag and drop a file or upload one
Upload
Fetch certificate from URL

Verify certificate

⚠ **Certificate change detected**

The current certificate does not match the certificate fetched from URL. Manually validate the fingerprint before proceeding.

Issuer

OU=VMware Engineering,O=localhost,ST=California,C=US,DC=local,DC=vsphere,CN=CA

SHA-1 fingerprint

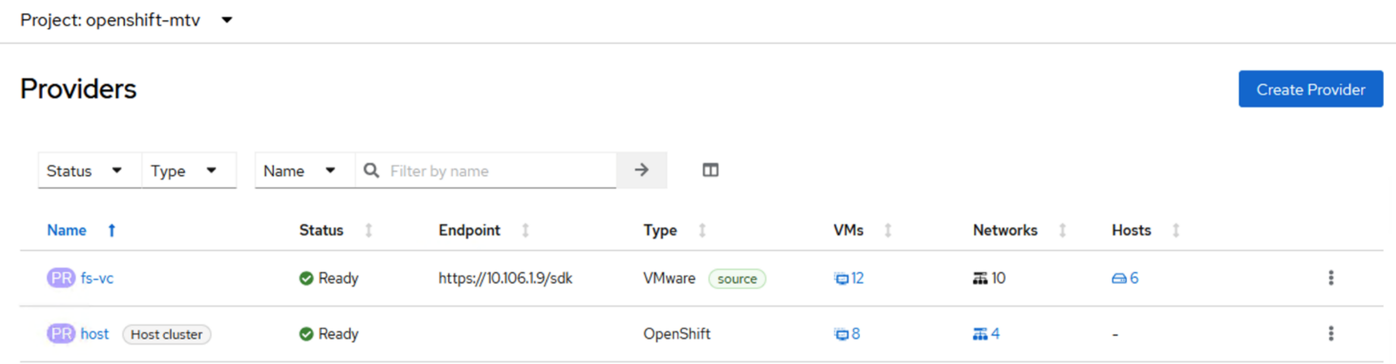
77:E1:A7:42:31:F2:07:A1:46:CE:2E:51:3D:E2:92:F4:9D:57:38:8F

Expiration date

Sun, 26 Jul 2026 06:20:31 GMT

☑ I trust the authenticity of this certificate

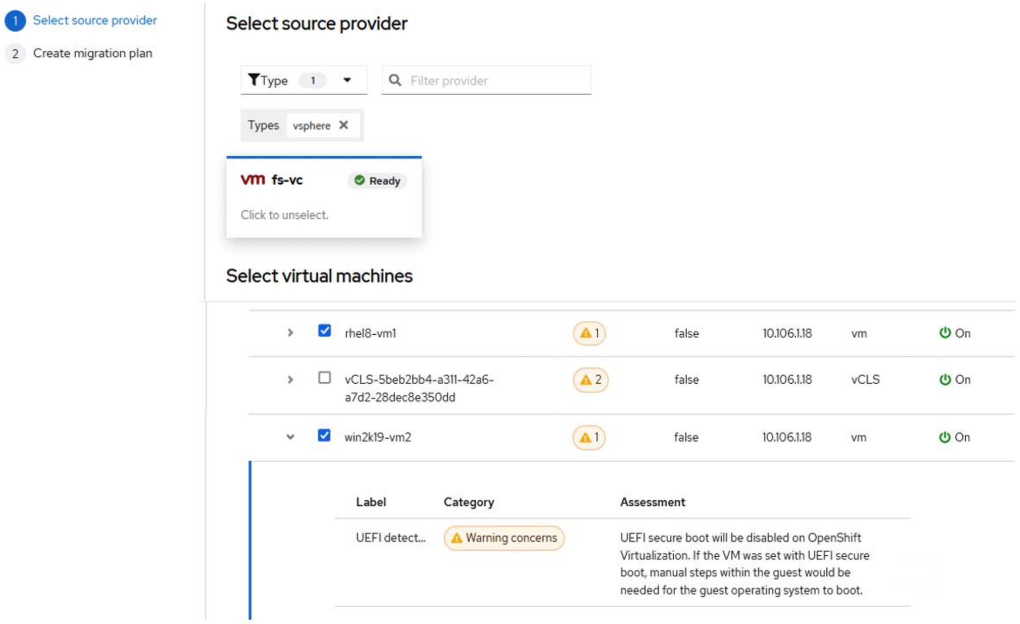
Confirm Cancel



Name	Status	Endpoint	Type	VMs	Networks	Hosts
PR fs-vc	✓ Ready	https://10.106.1.9/sdk	VMware source	12	10	6
PR host Host cluster	✓ Ready		OpenShift	8	4	-

You will see the vSphere provider (masked as source) just created along with the Default “host” provider for OpenShift Virtualization. Openshift will connect to the vSphere environment and fetches the inventory of the vSphere cluster.

Step 4. To create Migration Plan, go to **Migration > Plan for Virtualization** and click **Create Plan**. Select the vSphere provider previously created. Select the VMs which you would like to migrate from vSphere to OpenShift virtualization. Click **Next**.



1 Select source provider
2 Create migration plan

Select source provider

Type 1 | Filter provider

Types vsphere ✕

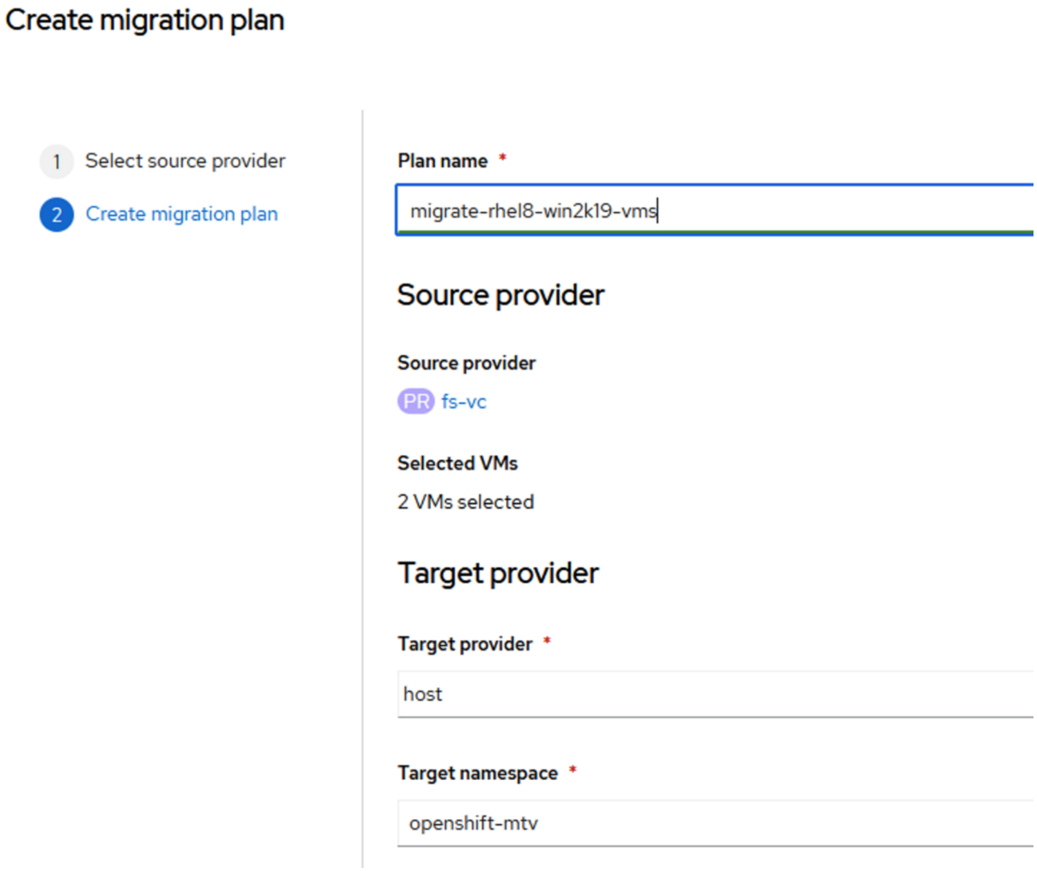
vm fs-vc ✔ Ready
Click to unselect.

Select virtual machines

›	☑	rhel8-vm1	⚠ 1	false	10.106.1.18	vm	⏻ On
›	□	vCLS-5beb2bb4-a311-42a6-a7d2-28dec8e350dd	⚠ 2	false	10.106.1.18	vCLS	⏻ On
⌄	☑	win2k19-vm2	⚠ 1	false	10.106.1.18	vm	⏻ On

Label	Category	Assessment
UEFI detect...	⚠ Warning concerns	UEFI secure boot will be disabled on OpenShift Virtualization. If the VM was set with UEFI secure boot, manual steps within the guest would be needed for the guest operating system to boot.

Step 5. Provide a name to the migration plan. Select the Target provider as host.



Create migration plan

1 Select source provider
2 Create migration plan

Plan name *
migrate-rhel8-win2k19-vms

Source provider

Source provider
PR fs-vc

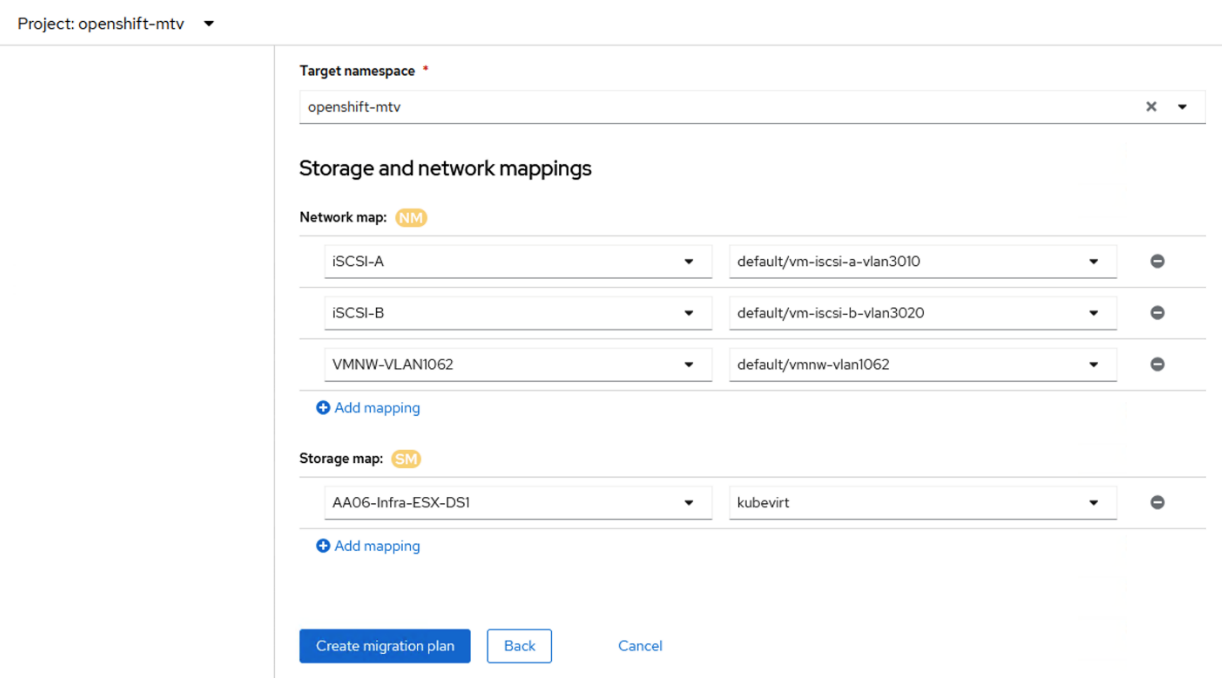
Selected VMs
2 VMs selected

Target provider

Target provider *
host

Target namespace *
openshift-mtv

Step 6. Adjust and map the VM's network and storage mappings from vSphere environment to OpenShift virtualization environment. Click **Create migration plan**.



Project: openshift-mtv

Target namespace *

openshift-mtv

Storage and network mappings

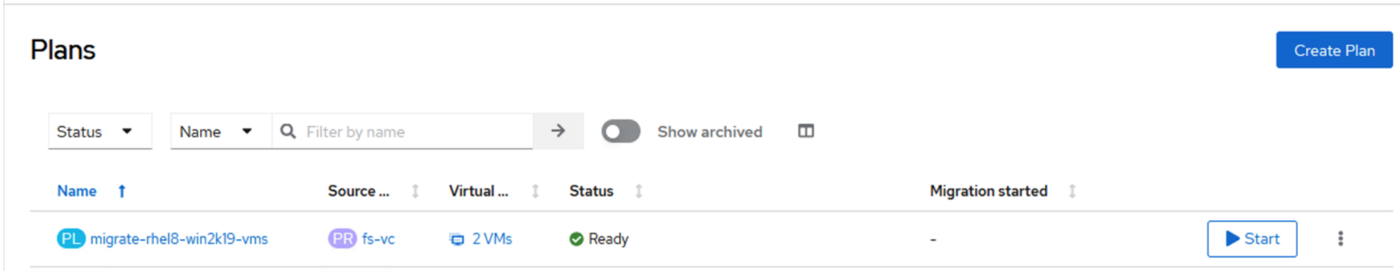
Network map: NM

iSCSI-A	default/vm-iscsi-a-vlan3010
iSCSI-B	default/vm-iscsi-b-vlan3020
VMNW-VLAN1062	default/vmnw-vlan1062

Storage map: SM

AA06-Infra-ESX-DS1	kubevirt

Now you will see the migration plan created under the Plans for Virtualization tab.

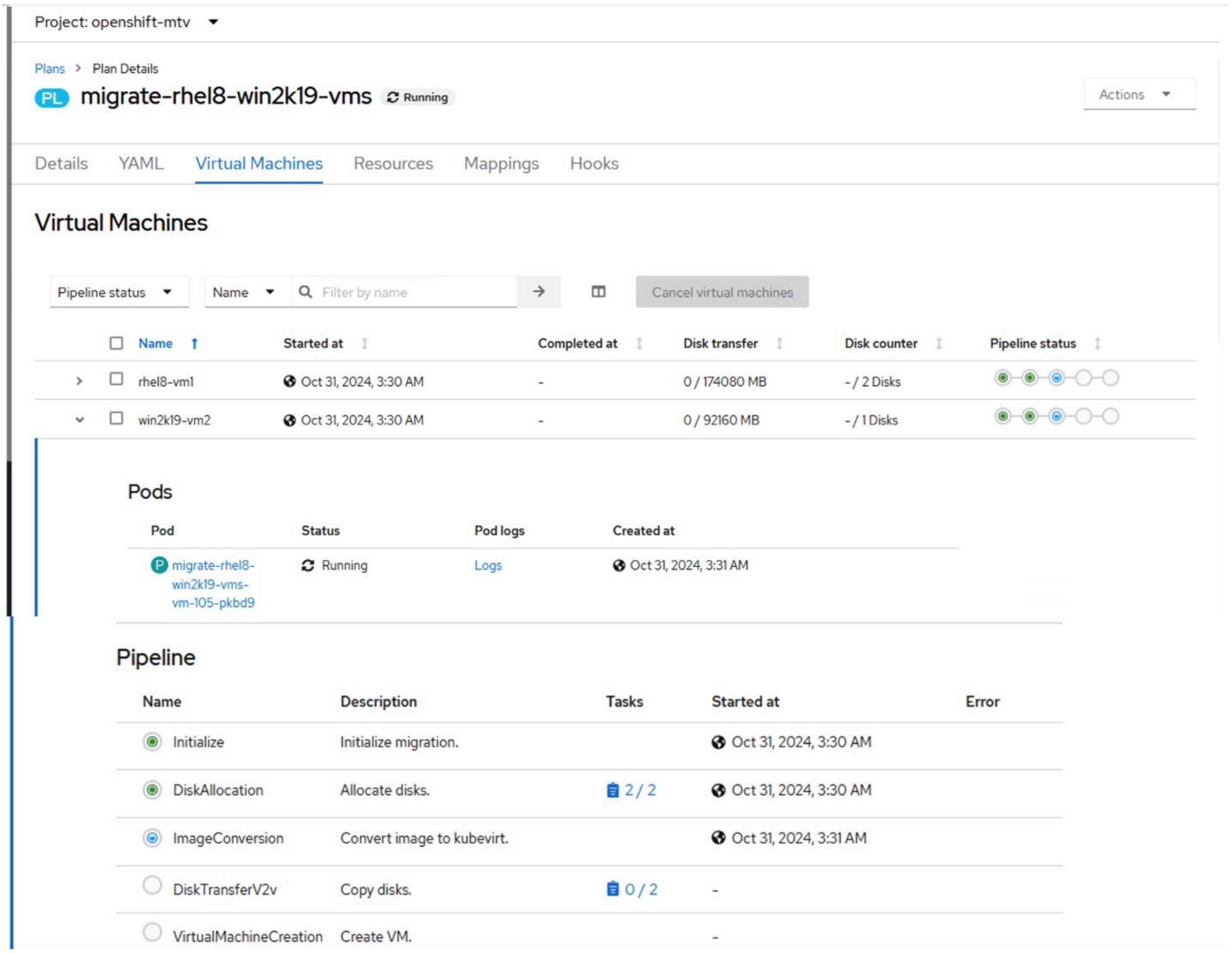


Plans

Name	Source ...	Virtual ...	Status	Migration started		
PL migrate-rhel8-win2k19-vms	PR fs-vc	2 VMs	Ready	-	Start	⋮

Step 7. To start the migration of the VMs, click **Start**.

Step 8. When the VM migration starts, you can see the status of the migration by navigating to the Migration plan and virtual machines tab as shown below. Scroll down to the Pipeline section to see the current task that is in progress.



Project: openshift-mtv

Plans > Plan Details

PL migrate-rhel8-win2k19-vms Running

Virtual Machines

	Name	Started at	Completed at	Disk transfer	Disk counter	Pipeline status
› □	rhel8-vm1	Oct 31, 2024, 3:30 AM	-	0 / 174080 MB	- / 2 Disks	●●●○○
⌄ □	win2k19-vm2	Oct 31, 2024, 3:30 AM	-	0 / 92160 MB	- / 1 Disks	●●●○○

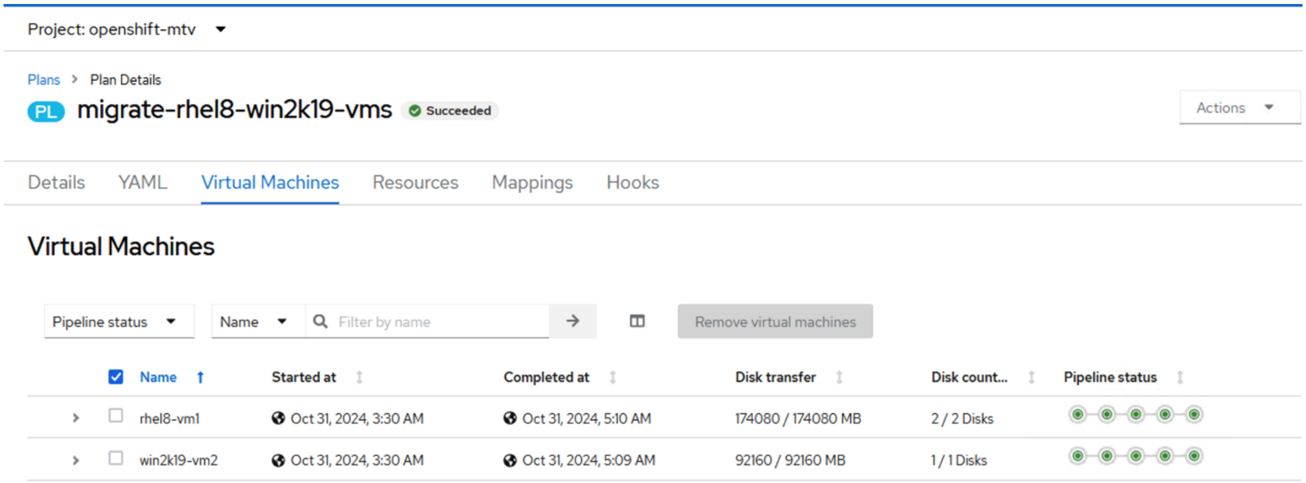
Pods

Pod	Status	Pod logs	Created at
P migrate-rhel8-win2k19-vms-vm-105-pkbd9	Running	Logs	Oct 31, 2024, 3:31 AM

Pipeline

Name	Description	Tasks	Started at	Error
● Initialize	Initialize migration.		Oct 31, 2024, 3:30 AM	
● DiskAllocation	Allocate disks.	2 / 2	Oct 31, 2024, 3:30 AM	
● ImageConversion	Convert image to kubevirt.		Oct 31, 2024, 3:31 AM	
○ DiskTransferV2v	Copy disks.	0 / 2	-	
○ VirtualMachineCreation	Create VM.		-	

When the VMs are migrated successfully, the migration plan will show as succeeded as shown below:



Project: openshift-mtv

Plans > Plan Details

PL migrate-rhel8-win2k19-vms Succeeded

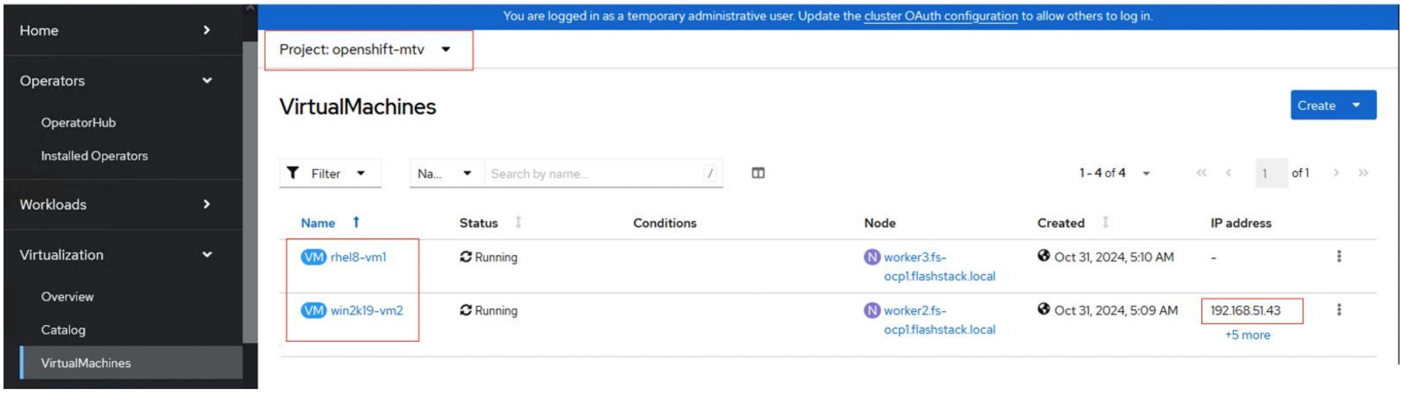
Virtual Machines

	Name	Started at	Completed at	Disk transfer	Disk count...	Pipeline status
› □	rhel8-vm1	Oct 31, 2024, 3:30 AM	Oct 31, 2024, 5:10 AM	174080 / 174080 MB	2 / 2 Disks	●●●●●
› □	win2k19-vm2	Oct 31, 2024, 3:30 AM	Oct 31, 2024, 5:09 AM	92160 / 92160 MB	1 / 1 Disks	●●●●●

The migrated VMs will be powered on and shown in the project which you selected in the plan.

Note: The In-Guest iSCSI disks of windows VM (win2k19-vm2) are discovered and connected automatically as the VM is mapped to the required iSCSi network interfaces.

Note: Virtual machines with guest-initiated storage connections are not handled by MTV and could require additional steps or reconfiguration after the VM migration to OpenShift environment. Especially, if you have different Storage Array with different IP addresses and VLANs in OpenShift environment compared to vSphere, you might need perform additional steps within the Guest VMs to connect to the guest-initiated storage volumes.



You are logged in as a temporary administrative user. Update the cluster OAuth configuration to allow others to log in.

Project: openshift-mtv

VirtualMachines

Name	Status	Conditions	Node	Created	IP address
VM rhel8-vm1	Running		N worker3.fs-ocp1.flashstack.local	Oct 31, 2024, 5:10 AM	-
VM win2k19-vm2	Running		N worker2.fs-ocp1.flashstack.local	Oct 31, 2024, 5:09 AM	192.168.51.43 +5 more

Post Migration Steps

The following additional steps must be taken after migrating VMs from vSphere environment to OpenShift Virtualization:

- Uninstall the VMware tools from the VMs since the OpenShift MTV installs the QEMU Guest agent on both linux and windows based virtual machines.
- Additional configuration steps are required for the Guest initiated storage disks since the MTV does not handle such storage devices.
- For Linux VMs, network interface names will be renamed to enp1sx while the original names on vSphere environment would be ensxx. You can change the interface name by editing the network settings and restart the VM.

About the Authors

Gopu Narasimha Reddy, Technical Marketing Engineer, Cisco Systems, Inc.

Gopu Narasimha Reddy is a Technical Marketing engineer with the UCS Solutions team at Cisco. He is currently focused on validating and developing solutions on various Cisco UCS platforms for enterprise database workloads with different operating environments including Windows, VMware, Linux, and Kubernetes. Gopu is also involved in publishing database benchmarks on Cisco UCS servers. His areas of interest include building and validating reference architectures, development of sizing tools in addition to assisting customers in database deployments.

Vijay Bhaskar Kulari, Solution Architect, Pure Storage, Inc.

Vijay Kulari works at Pure Storage part of Solutions team. He specializes in designing, developing, and optimizing solutions across Storage, Converged Infrastructure, Cloud, and Container technologies. His role includes establishing best practices, streamlining automation, and creating technical content. As an experienced Solution Architect, Vijay has a strong background in VMware products, storage solutions, converged and hyper-converged infrastructure, and container platforms

Acknowledgements

For their support and contribution to the design, validation, and creation of this Cisco Validated Design, the authors would like to thank:

- John George, Technical Marketing Engineer, Cisco Systems, Inc.
- Archana Sharma, Technical Marketing Engineer, Cisco Systems, Inc.
- Chris O’Brien, Senior Director, Technical Marketing, Cisco Systems, Inc.
- Craig Waters, Solutions Director, Pure Storage, Inc.
- Sanjay Naikwadi, Solutions Architect, Pure Storage, Pure Storage, Inc.
- Venkat Pinisetti, Member of Technical Staff, Pure Storage, Pure Storage, Inc.

Appendix

This appendix contains the following:

- Appendix A - References used in this guide

Appendix A – References used in this guide

Compute

Cisco Intersight: https://www.intersight.com

Cisco Intersight Managed Mode: https://www.cisco.com/c/en/us/td/docs/unified_computing/Intersight/b_Intersight_Managed_Mode_Configuration_Guide.html

Cisco Unified Computing System: http://www.cisco.com/en/US/products/ps10265/index.html

Cisco UCS 6500 Series Fabric Interconnects: https://www.cisco.com/c/en/us/products/collateral/servers-unified-computing/ucs6536-fabric-interconnect-ds.html

Cisco UCS 6536 Fabric Interconnect Data Sheet: https://www.cisco.com/c/en/us/products/collateral/servers-unified-computing/ucs6536-fabric-interconnect-ds.html

Network

Cisco Nexus 9300-GX Series Switches: https://www.cisco.com/c/en/us/products/collateral/switches/nexus-9000-series-switches/nexus-9300-gx-series-switches-ds.html

Pure Storage

Pure Storage FlashArray//X: https://www.purestorage.com/products/nvme/flasharray-x.html

Pure Storage FlashArray//XL: https://www.purestorage.com/products/nvme/flasharray-xl.html

Red Hat OpenShift

Documentation: https://docs.openshift.com/

Red Hat OpenShift Container Platform: https://www.redhat.com/en/technologies/cloud-computing/openshift/container-platform

Red Hat OpenShift Virtualization: https://docs.redhat.com/en/documentation/openshift_container_platform/4.16/html/virtualization/index

Red Hat Hybrid Cloud Console: https://cloud.redhat.com/

Portworx Enterprise

https://docs.portworx.com/

Interoperability Matrix

Cisco UCS Hardware Compatibility Matrix: https://ucshcltool.cloudapps.cisco.com/public/

Pure Storage FlashStack Compatibility Matrix. Note, this interoperability list will require a support login from Pure: https://support.purestorage.com/FlashStack/Product_Information/FlashStack_Compatibility_Matrix

Feedback

For comments and suggestions about this guide and related guides, join the discussion on Cisco Community at https://cs.co/en-cvds.

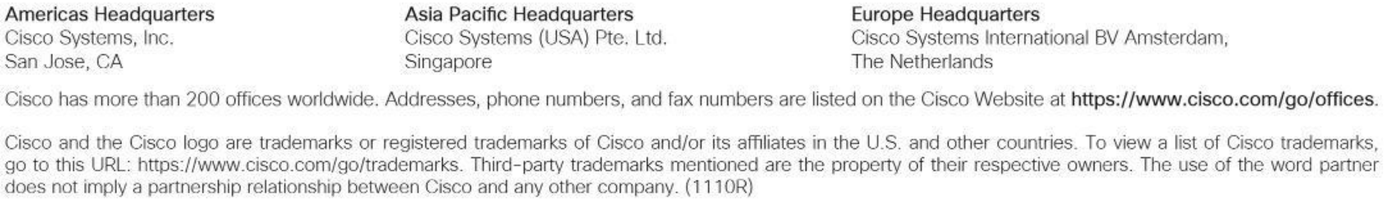
CVD Program

ALL DESIGNS, SPECIFICATIONS, STATEMENTS, INFORMATION, AND RECOMMENDATIONS (COLLECTIVELY, " DESIGNS") IN THIS MANUAL ARE PRESENTED " AS IS," WITH ALL FAULTS. CISCO AND ITS SUPPLIERS DISCLAIM ALL WARRANTIES, INCLUDING, WITHOUT LIMITATION, THE WARRANTY OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT OR ARISING FROM A COURSE OF DEALING, USAGE, OR TRADE PRACTICE. IN NO EVENT SHALL CISCO OR ITS SUPPLIERS BE LIABLE FOR ANY INDIRECT, SPECIAL, CONSEQUENTIAL, OR INCIDENTAL DAMAGES, INCLUDING, WITHOUT LIMITATION, LOST PROFITS OR LOSS OR DAMAGE TO DATA ARISING OUT OF THE USE OR INABILITY TO USE THE DESIGNS, EVEN IF CISCO OR ITS SUPPLIERS HAVE BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

THE DESIGNS ARE SUBJECT TO CHANGE WITHOUT NOTICE. USERS ARE SOLELY RESPONSIBLE FOR THEIR APPLICATION OF THE DESIGNS. THE DESIGNS DO NOT CONSTITUTE THE TECHNICAL OR OTHER PROFESSIONAL ADVICE OF CISCO, ITS SUPPLIERS OR PARTNERS. USERS SHOULD CONSULT THEIR OWN TECHNICAL ADVISORS BEFORE IMPLEMENTING THE DESIGNS. RESULTS MAY VARY DEPENDING ON FACTORS NOT TESTED BY CISCO.

CCDE, CCENT, Cisco Eos, Cisco Lumin, Cisco Nexus, Cisco StadiumVision, Cisco TelePresence, Cisco WebEx, the Cisco logo, DCE, and Welcome to the Human Network are trademarks; Changing the Way We Work, Live, Play, and Learn and Cisco Store are service marks; and Access Registrar, Aironet, AsyncOS, Bringing the Meeting To You, Catalyst, CCDA, CCDP, CCIE, CCIP, CCNA, CCNP, CCSP, CCVP, Cisco, the Cisco Certified Internetwork Expert logo, Cisco IOS, Cisco Press, Cisco Systems, Cisco Systems Capital, the Cisco Systems logo, Cisco Unified Computing System (Cisco UCS), Cisco UCS B-Series Blade Servers, Cisco UCS C-Series Rack Servers, Cisco UCS S-Series Storage Servers, Cisco UCS X-Series, Cisco UCS Manager, Cisco UCS Management Software, Cisco Unified Fabric, Cisco Application Centric Infrastructure, Cisco Nexus 9000 Series, Cisco Nexus 7000 Series. Cisco Prime Data Center Network Manager, Cisco NX-OS Software, Cisco MDS Series, Cisco Unity, Collaboration Without Limitation, EtherFast, EtherSwitch, Event Center, Fast Step, Follow Me Browsing, FormShare, GigaDrive, HomeLink, Internet Quotient, IOS, iPhone, iQuick Study, LightStream, Linksys, MediaTone, MeetingPlace, MeetingPlace Chime Sound, MGX, Networkers, Networking Academy, Network Registrar, PCNow, PIX, PowerPanels, ProConnect, ScriptShare, SenderBase, SMARTnet, Spectrum Expert, StackWise, The Fastest Way to Increase Your Internet Quotient, TransPath, WebEx, and the WebEx logo are registered trade-marks of Cisco Systems, Inc. and/or its affiliates in the United States and certain other countries. (LDW_P1)

All other trademarks mentioned in this document or website are the property of their respective owners. The use of the word partner does not imply a partnership relationship between Cisco and any other company. (0809R)



Americas Headquarters
Cisco Systems, Inc.
San Jose, CA

Asia Pacific Headquarters
Cisco Systems (USA) Pte. Ltd.
Singapore

Europe Headquarters
Cisco Systems International BV Amsterdam,
The Netherlands

Cisco has more than 200 offices worldwide. Addresses, phone numbers, and fax numbers are listed on the Cisco Website at **https://www.cisco.com/go/offices**.