Cisco Container Platform 2.1.0 Installation Guide

First Published: 2018-11-01

Last Modified: 2018-11-27

Americas Headquarters
Cisco Systems, Inc.
170 West Tasman Drive
San Jose, CA 95134-1706
USA
http://www.cisco.com
Tel: 408 526-4000
800 553-NETS (6387)
Fax: 408 527-0883

THE SPECIFICATIONS AND INFORMATION REGARDING THE PRODUCTS IN THIS MANUAL ARE SUBJECT TO CHANGE WITHOUT NOTICE. ALL STATEMENTS, INFORMATION, AND RECOMMENDATIONS IN THIS MANUAL ARE BELIEVED TO BE ACCURATE BUT ARE PRESENTED WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED. USERS MUST TAKE FULL RESPONSIBILITY FOR THEIR APPLICATION OF ANY PRODUCTS.

THE SOFTWARE LICENSE AND LIMITED WARRANTY FOR THE ACCOMPANYING PRODUCT ARE SET FORTH IN THE INFORMATION PACKET THAT SHIPPED WITH THE PRODUCT AND ARE INCORPORATED HEREIN BY THIS REFERENCE. IF YOU ARE UNABLE TO LOCATE THE SOFTWARE LICENSE OR LIMITED WARRANTY, CONTACT YOUR CISCO REPRESENTATIVE FOR A COPY.

NOTWITHSTANDING ANY OTHER WARRANTY HEREIN, ALL DOCUMENT FILES AND SOFTWARE OF THESE SUPPLIERS ARE PROVIDED “AS IS" WITH ALL FAULTS. CISCO AND THE ABOVE-NAMED SUPPLIERS DISCLAIM ALL WARRANTIES, EXPRESSED OR IMPLIED, INCLUDING, WITHOUT LIMITATION, THOSE 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 THIS MANUAL, EVEN IF CISCO OR ITS SUPPLIERS HAVE BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

Any Internet Protocol (IP) addresses and phone numbers used in this document are not intended to be actual addresses and phone numbers. Any examples, command display output, network topology diagrams, and other figures included in the document are shown for illustrative purposes only. Any use of actual IP addresses or phone numbers in illustrative content is unintentional and coincidental.

All printed copies and duplicate soft copies of this document are considered uncontrolled. See the current online version for the latest version.

Cisco has more than 200 offices worldwide. Addresses and phone numbers are listed on the Cisco website at www.cisco.com/go/offices.

CONTENTS

CHAPTER 1

Cisco Container Platform

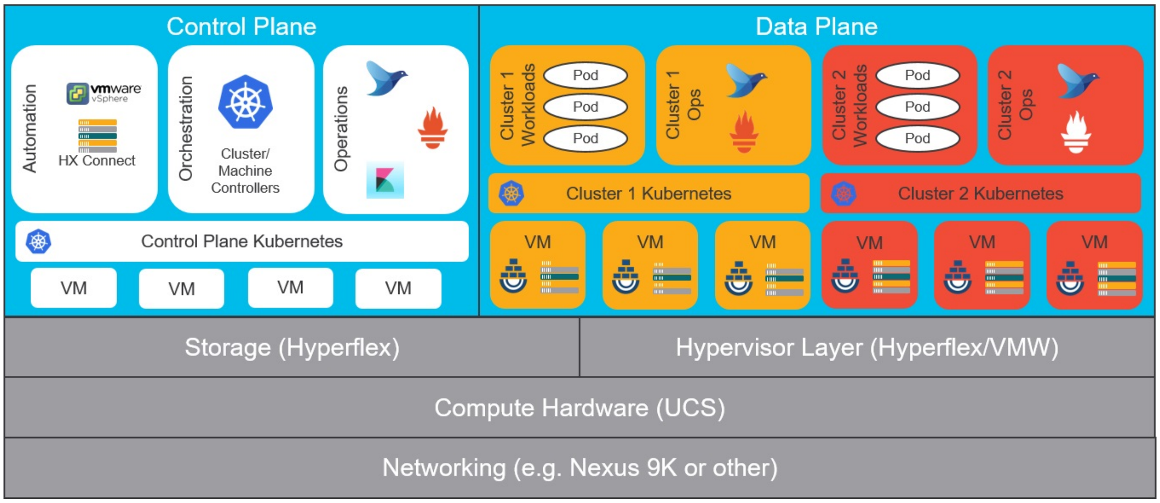
Cisco Container Platform is a turnkey, production grade, extensible platform to deploy and manage multiple Kubernetes clusters. It runs on 100% upstream Kubernetes. Cisco Container Platform offers seamless container networking, enterprise-grade persistent storage, built-in production-grade security, integrated logging, monitoring and load balancing.

This chapter contains the following topics:

Cisco Container Platform Architecture Overview

The following figure shows the architecture of Cisco Container Platform deployment with HyperFlex and ACI integration.

Figure 1: Cisco Container Platform Architecture Overview



Note Cisco Container Platform can run on top of an ACI networking fabric as well as on a non-ACI networking fabric that performs standard L3 switching.

At the bottom of the stack, there is an ACI fabric that consists of Nexus switches, Application Policy Infrastructure Controllers (APICs) and Fabric Interconnects (FIs). The next layer up is the UCS servers running the HyperFlex software. HyperFlex provides virtualized compute resources through VMware, and distributed storage resources through the HyperFlex converged data platform.

The next layer up is the Cisco Container Platform Control Plane and Data Plane. In the preceeding figure, Cisco Container Platform Control Plane runs on the four VMs on the left.

Kubernetes tenant clusters are preconfigured to support Persistent Volumes using vSphere Cloud Provider and FlexVolumes using HyperFlex volume plugin. Both implementations use the underlying replicated, highly available HyperFlex data platform for storage.

Components of Cisco Container Platform

The following table describes the components of Cisco Container Platform.

Function	Component
Container Runtime	Docker CE
Operating System	Ubuntu
Orchestration	Kubernetes
IaaS	vSphere
Infrastructure	HyperFlex
Container Network Interface (CNI)	ACI, Contiv, Calico
SDN	ACI
Container Storage	HyperFlex Flex Driver
Load Balancing	NGINX, Envoy
Service Mesh	Istio, Envoy
Monitoring	Prometheus, Grafana
Logging	Elasticsearch, Fluentd, and Kibana (EFK) stack

Sample Deployment Topology

This section describes a sample deployment topology of the Cisco Container Platform and illustrates the network topology requirements at a conceptual level. Future sections of the document such as System

Requirements, on page 5 and Installing Cisco Container Platform on vSphere Web Client, on page 15 provide additional configuration details based on these concepts.

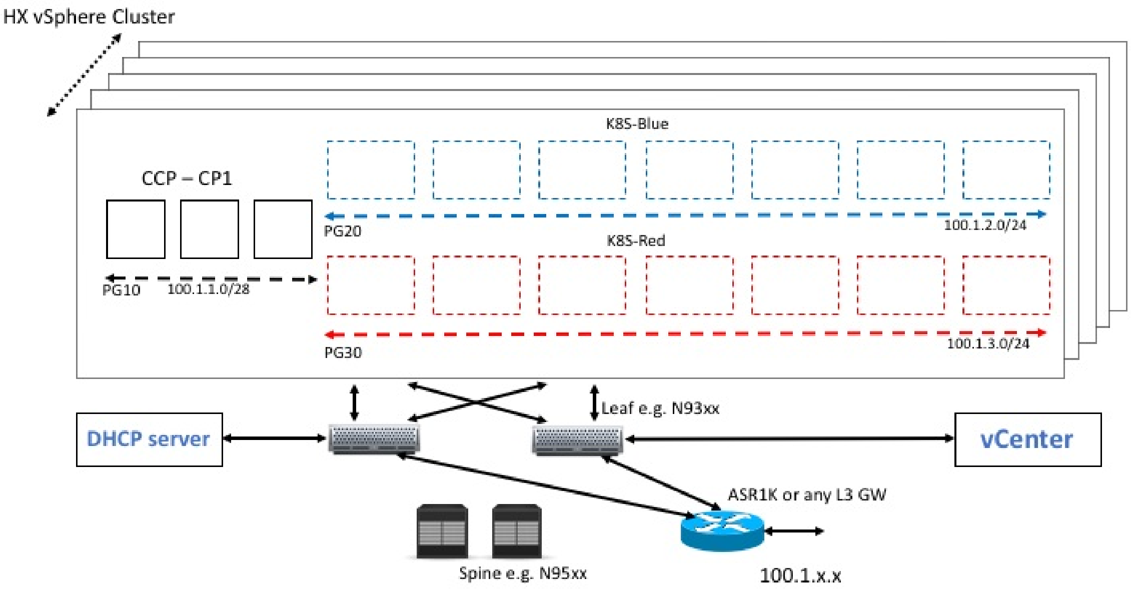
Note

In this example, the deployment target is a VMware vSphere virtualization platform, and Cisco Container Platform is using a non-ACI CNI such as Calico or Contiv. Other deployment environments are conceptually similar but with some slight differences appropriate to those environments.

In this case, it is expected that the vSphere based cluster is set up, provisioned and fully functional for virtualization and Virtual Machine functionality before any installation of Cisco Container Platform. You can refer to the standard VMware documentation for details on vSphere installation.

The following figure illustrates an example vSphere cluster on which Cisco Container Platform is to be deployed.

Figure 2: Example vSphere Cluster



Once the vSphere cluster is ready to provision VMs, the admin then provisions one or more VMWare port groups (for example PG10, PG20 and PG30 in the figure) on which virtual machines will subsequently be provisioned as container cluster nodes. Basic L2 switching using VMWare vswitch functionality can be used to implement these port groups. IP subnets should be set aside for use on these port groups and the VLANs used to implement these port groups should be terminated on an external L3 gateway (such as the ASR1K shown in the figure). The control plane cluster and tenant plane Kubernetes clusters of Cisco Container Platform can then be provisioned on these port groups.

All provisioned Kubernetes clusters may choose to use a single shared port group or separate port groups may be provisioned (1 per Kubernetes cluster) depending on the isolation needs of the deployment. Layer 3 network isolation may be used between these different port groups as long as the following conditions are met:

- There must be L3 IP connectivity between the port group used for the Control Plane cluster and each of the Tenant cluster port groups

- The IP address of the vCenter server must be reachable from the Control plane cluster
- For current versions of Cisco Container Platform, a DHCP server must be provisioned for assigning IPs to the cluster nodes and it must be reachable from all cluster nodes

The simplest functional topology would be to use a single shared port group for all clusters with a single IP subnet to be used to assign IPs for all container cluster VMs. This IP subnet can be used to assign one IP per cluster VM and up to four virtual IPs per Kubernetes cluster, but would not be used to assign individual Kubernetes pod IPs. Hence, a reasonable capacity planning estimate for the size of this IP subnet is as follows:

(The expected total number of container cluster VMs across all clusters) + 3 x (The total number of expected Kubernetes clusters)

Container Network Interface Plugins

Cisco Container Platform supports multiple Kubernetes CNI plugins such as:

- ACI is the recommended plugin for use with an ACI fabric. It is optimized for use with an ACI fabric. ACI is fully supported by Cisco.
- Calico is recommended when an ACI fabric is not used. It can be used for quick evaluation of Cisco Container Platform. Calico is an integrated CNI plugin and is not fully supported under the Cisco commercial support agreement.
- Contiv (Tech Preview) is a user space switch that is optimized for high performance and scale.

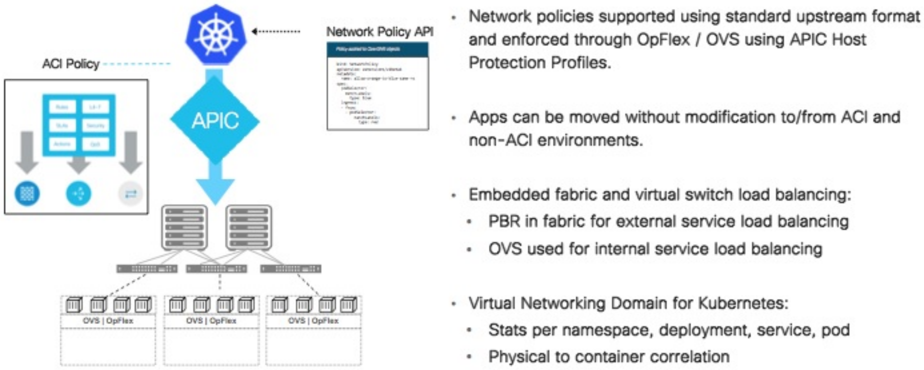
Operationally, all the CNI plugins offer the same experience to the customer. The container network connectivity is seamless and network policies are applied using Kubernetes NetworkPolicies. Under-the-hood, both ACI and Contiv offer advanced feature support. ACI allows you to map CNI NetworkPolicies to an ACI fabric and supports richer underlay policies such as common policies for containers/virtual machines/physical servers and inter-Kubernetes cluster policies. Additionally, ACI supports Kubernetes Type LoadBalancer using PBR policies in the ACI fabric.

ACI

ACI is tightly integrated with the ACI fabric. It supports underlay integration with the ACI fabric and hardware accelerated load balancing.

The following figure shows the architecture of ACI.

Figure 3: Architecture of ACI



- Network policies supported using standard upstream format and enforced through OpFlex / OVS using APIC Host Protection Profiles.
- Apps can be moved without modification to/from ACI and non-ACI environments.
- Embedded fabric and virtual switch load balancing:
  - PBR in fabric for external service load balancing
  - OVS used for internal service load balancing
- Virtual Networking Domain for Kubernetes:
  - Stats per namespace, deployment, service, pod
  - Physical to container correlation

System Requirements

This section describes the software, hardware, storage, and network requirements that are necessary to deploy Cisco Container Platform.

Supported Version Matrix

Cisco Container Platform uses various software and hardware components. The following table provides information on the validated versions of each component.

Component	Validated Version
Kubernetes	1.10 1.11
ACI	3.2(2o)
HyperFlex software	3.0(1b)+ 3.5
vSphere	vSphere 6.0 (u2)+ vSphere 6.5

Note Cisco Container Platform is supported on all hardware configurations that are supported by the required HyperFlex software versions. For more information on HyperFlex hardware configurations, refer to the UCS HyperFlex product documentation.

Requirements for VMware on UCS

The following are the requirements for a topology where VMware is deployed on a UCS platform, and HyperFlex is not used:

- Use the Enterprise Plus license to set up the VMware clusters with HA and DRS enabled. For more information on the supported versions of VMware, see Supported Version Matrix, on page 5.
- Use a processor that has an Ivy Bridge (UCS C220 M4) or a newer microarchitecture so that the CPU RDRAND instruction set is available.
- If you are enabling VMware EVC Mode, you must use an Ivy Bridge or a later micro-architecture so that the CPU RDRAND instruction set is available.
- Ensure that the following requirements are met on the network that you want to use for deploying the Cisco Container Platform VMware instances:
  - DHCP is enabled.
  - A static pool of IP addresses is available in the network for assigning IP addresses to the master nodes and the VIP pools.
  - The network is routable to and from the VMware vCenter server.
  - The client install machine is routable to the network during the Cisco Container Platform control plane install.
  - The network allows communication between Cisco Container Platform VM instances. You must not use a private LAN.
- By default, the Cisco Container Platform control plane VXLAN network uses 192.168.0.0/16 network. If you have routed IP addresses in that space, you must assign another RFC1918 range for your VXLAN network. It does not need to be a full /16 network, a /22 network is adequate for the Cisco Container Platform control plane.
- If the DHCP server does not provide the location of the NTP service, enter the NTP address in the Installer UI, under **Control Plane Settings** > **Advanced Settings**.
- Ensure that a shared datastore that is accessible to the VMware hosts such as NFS or iSCSI or FC, with at least 200 GB of storage is available.

Software Requirements

Ensure that the following software applications are installed in the deployment environment:

- VMware vCenter server 6.5
- VMware client integration plugin
- vSphere Flash client
- A supported version of HyperFlex software

  For more information on installing HyperFlex and accessing the HyperFlex Connect UI, refer to the latest HperFlex documentation.

Hardware Requirements

- In Cisco Container Platform 1.3.0 or later, the hypervisor hosts need to run CPUs with an Ivy Bridge (UCS C220 M4) or newer micro-architecture.

Storage Requirements

Once HyperFlex is installed, you need to configure a shared datastore that is accessible to the hosts in the cluster for the following purposes:

- Persistent volume storage
- Deploying the Cisco Container Platform tenant base VM

Configuring Shared Datastore

Step 1 Log in to the **HX Connect UI** using the VMware vCenter SSO administrator credentials.

Step 2 In the left pane, click **Manage** > **Datastores**.

Step 3 Perform these steps to create a datastore for provisioning the Kubernetes persistent volume storage and deploying the Cisco Container Platform tenant base VM:

a) In the right pane, click **Create Datastore**.

b) In the **Name** field, enter **`ds1`**, and then enter a size and block size for the datastore.

Note We recommend that you use **`1TB`** size and **`8K`** block size.

c) Click **Create Datastore**.

The newly created datastore is available on vCenter.

Configuring Link-local Network for HyperFlex iSCSI Communication

The FlexVolume plug-in requires a host-only link between each VM that runs Kubernetes and the Internet Small Computer System Interface (iSCSI) target on the ESX host.

For HyperFlex 3.5+

Step 1 Log in to the **HX Connect UI**.

Step 2 Choose **Settings** > **Integrations** > **Kubernetes**.

Step 3 Click **Enable All Node** and wait until the **KUBERNETES STORAGE PROVISIONING** option is enabled.
The HyperFlex infrastructure is configured and ready to use for Cisco Container Platform with Kubernetes persistent volume support.

For HyperFlex 3.0.x

Step 1 Open an SSH session to the HyperFlex 3.0 Platform Installer VM or one of the HyperFlex Controller VMs and log in as a root user.

Step 2 Perform these steps to get the vCenter details that you need to enter when you run the `add_vswitch.py` script.

a) Run the following command to get the vCenter datacenter name and vCenter cluster name.

```
stcli cluster info | grep -i vcenter
```

b) Run the following command to validate the reachability of vCenter IP address.

```
ping <vcenter URL>
```

Step 3 Navigate to the following location:

`/usr/share/springpath/storfs-misc/hx-scripts/`

Step 4 Run the `add_vswitch.py` script.

```
python add_vswitch.py --vcenter-ip <vCenter IP address>
```

When prompted, specify the vCenter credentials, datacenter name, and cluster name that you got from the output of Step 2.

The HyperFlex infrastructure is configured and ready to use for Cisco Container Platform with Kubernetes persistent volume support.

Resource Management Requirements

Enabling DRS and HA on Clusters

It is required that you enable DRS and HA on vCenter for the following reasons:

- DRS continuously monitors resource utilization across vSphere servers and intelligently balances VMs on the servers.
- HA provides easy to use, cost-effective high availability for applications running on virtual machines.

Step 1 In the vSphere Web Client, navigate to the host or cluster on which you want to deploy Cisco Container Platform.

Step 2 Click the **Configure** tab.

Step 3 Under **Services**, click **vSphere DRS**, and then click **Edit**.

Step 4 In the right pane of the **Edit Cluster Settings** window, check the **Turn ON vSphere DRS** check box, and then click **OK**.

Step 5 Under **Services**, click **vSphere Availability**, and then click **Edit**.

Step 6 In the right pane of the **Edit Cluster Settings** window, check the **Turn ON vSphere HA** check box, and then click **OK**.

Network Requirements

Provisioning a Port Group for Cisco Container Platform VM Deployment

Cisco Container Platform creates VMs that are attached to a Port Group on either a vSphere Standard Switch (VSS) or a Distributed Virtual Switch (DVS). The HyperFlex installer creates VSS switches in vSphere for the networks that are defined during installation. You need to create either VSS or DVS Switches for managing the VM traffic.

The following topics provide information on configuring a VSS or a DVS.

Configuring vSphere Standard Switch

Step 1 In the vSphere Web Client, navigate to the host or cluster on which you want to deploy Cisco Container Platform.

Step 2 Click the **Configure** tab.

Step 3 Expand **Networking**, and then select **Virtual switches**.

Step 4 Click **Add host networking**.

Step 5 Choose **Virtual Machine Port Group for a Standard Switch** as the connection type for which you want to use the new standard switch and click **Next**.

Step 6 Select **New standard switch** and click **Next**.

Step 7 Add physical network adapters to the new standard switch.

Step 8 Under **Assigned adapters**, click **Add adapters**.

Step 9 Select one or more physical network adapters from the list.

Step 10 From the **Failover order group** drop-down list, choose from the Active or Standby failover lists.

Step 11 For higher throughput and to provide redundancy, configure at least two physical network adapters in the Active list.

Step 12 Click **OK**.

Step 13 Enter connection settings for the adapter or the port group as follows:

a) Enter a network Label or the port group, or accept the generated label.
b) Set the VLAN ID to configure VLAN handling in the port group.

Step 14 On the **Ready to Complete** screen, click **OK**.

Configuring DHCP Server

Cisco Container Platform requires a DHCP server to be present. The Cisco Container Platform installer VM, Control Plane VMs and tenant cluster VMs get their primary interface IP addresses from the DHCP server. You must ensure that you have configured a DHCP server.

Reserving IP Addresses for Static Allocation

A static IP address is used during Cisco Container Platform installation for the **CCP Control Plane master node virtual IP** to support Cisco Container Platform upgrades. Additionally, Virtual IP address (VIP) is used as an external IP address for each Kubernetes cluster. VIPs are configured using VIP pools. You can obtain this IP address from the same or a different subnet and you must ensure that it is not part of a DHCP pool.

Static and DHCP IP Address Requirements

The following table summarizes the static and DHCP IP address requirements for the Cisco Container Platform components.

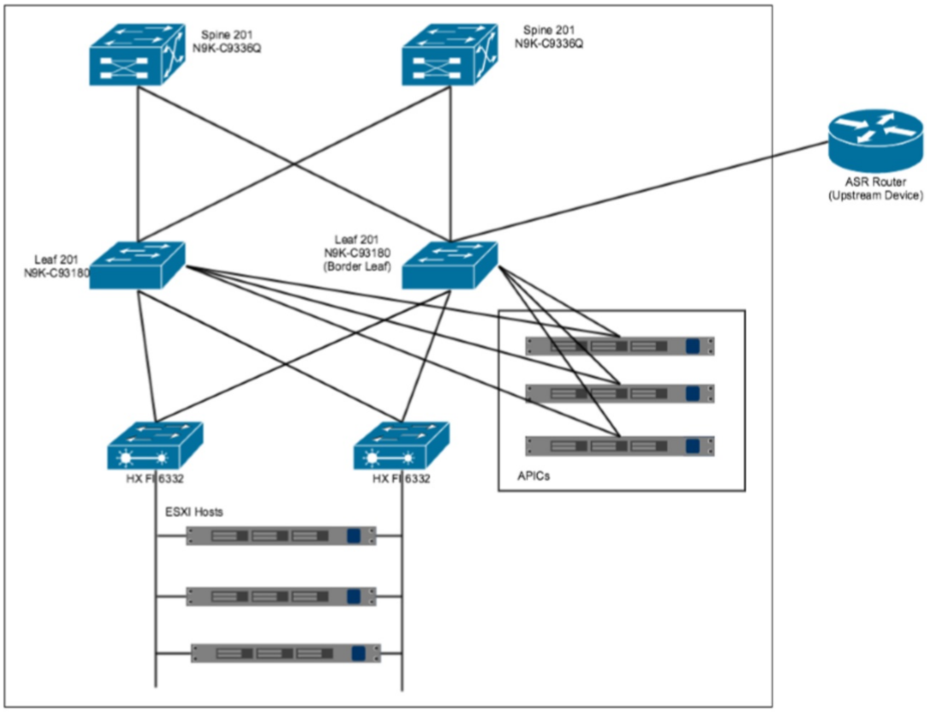
Component	Static IP	DHCP IP
Cisco Container Platform web interface	1	4
Installer VM	-	1
Tenant clusters	2 + Number of load balancer VIPs desired for applications	1 + Number of workers

ACI Integration Requirements

Cisco ACI enables you to group your application into End Point Groups (EPGs), define policies for the EPGs, and then deploy network policies on the ACI fabric. The policy enforcement is implemented using the spine and leaf architecture of the ACI fabric.

The following figure shows the components of a Cisco Container Platform ACI integrated network topology.

Figure 4: Cisco Container Platform ACI Integrated Network Topology



The main components of the network topology are as follows:

- **ACI Fabric** includes two spine nodes, two leaf nodes, and three APIC controllers. You can choose the number of the spine and leaf nodes and APIC controllers as per your network requirement.
- **HyperFlex Fabric Interconnect (FI)** includes two fabric interconnect switches connected between the ESXi hosts and the ACI leaf switches.
- **ESXi Hosts** includes a UCS server such as UCS C220 M4.
- **ASR router** is connected to an ACI border leaf for external internet access.

APIC Controller Requirements

If you are using ACI, ensure that you have configured the following settings on the APIC controller:

- Assign a port number other than 4094 for Infra VLAN as 4094 is reserved for provisioning HyperFlex fabric interconnect
- Create a common tenant
- Create a Virtual Route Forwarder (VRF) in the common tenant
- Create at least one L3OUT
- Create an Access Entity Profile (AEP) for the ACI tenant physical domain
- Create an AEP for L3OUT
- Create a Virtual Machine Manager (VMM) domain which connects to vSphere

For more information on configuring an APIC controller, refer to the latest ACI documentation.

HyperFlex FI Requirements

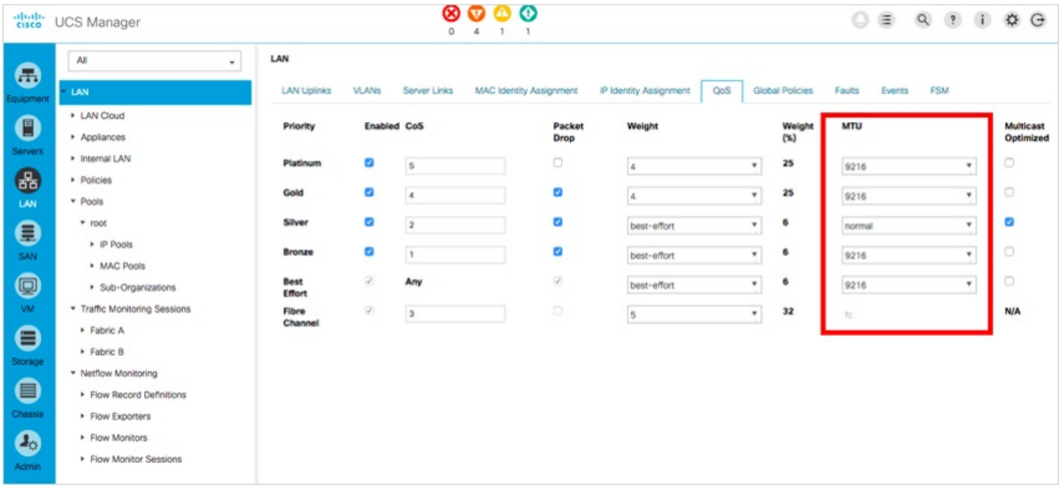
Ensure that you have configured the following settings on HyperFlex FI:

- Configure QOS

1. From the left pane, click **LAN**.
2. From the right pane, click the **QoS** tab, and then configure QoS.

Note Using the **MTU** configuration, you must set the priority that is associated with the QoS policy of the vNIC template.

Figure 5: QoS Tab



Priority	Enabled	CoS	Packet Drop	Weight	Weight (%)	MTU	Multicast Optimized
Platinum	☑	5	□	4	25	9216	□
Gold	☑	4	☑	4	25	9216	□
Silver	☑	2	☑	best-effort	6	normal	☑
Bronze	☑	1	☑	best-effort	6	9216	□
Best Effort	☑	Any	☑	best-effort	6	9216	□
Fibre Channel	☑	3	□	5	32	fc	N/A

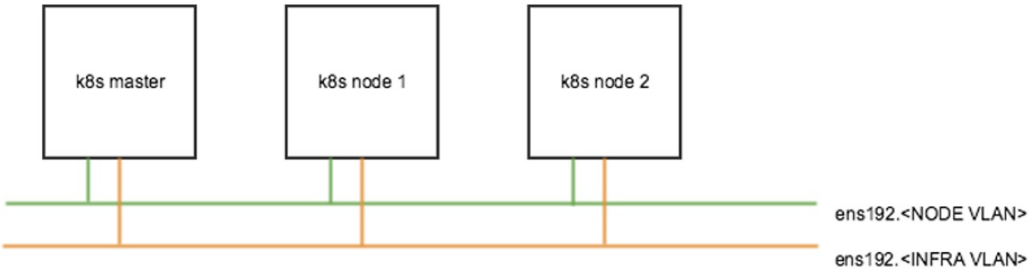
- Ensure that the tenant VLAN is allowed

Once Cisco Container Platform Control Plane and management node networking are configured, you can access the HyperFlex cluster on vSphere and install Cisco Container Platform. Each time you create a tenant cluster, the ACI constructs such as L3OUT, VRF, and AEP stored in the common tenant cluster are reused.

Tenant Cluster with ACI Deployment

With an ACI deployment, each tenant cluster is required to have its own routable subnet. The node VLAN, pod subnet, and multicast subnet range should not overlap between clusters. Cisco Container Platform ensures that the VLAN and subnet do not overlap.

Unlike other CNI, an ACI tenant cluster requires two VLAN subinterfaces, one for the Node VLAN, and another for the Infra VLAN. As shown in the following figure, Cisco Container Platform assigns unique Node VLAN IDs. You need to assign a unique Infra VLAN ID for clusters during cluster creation.



For more information on creating tenant clusters, refer to the *Creating Kubernetes Clusters* section of the *Cisco Container Platform User Guide*.

For more information on the ACI and CNI plugin, refer to the latest documentation on Cisco ACI and Kubernetes Integration.

Getting Cisco Container Platform Software

This chapter contains the following topics:

Downloading the Software

Before you begin the installation, you need to download the required software assets.

Step 1 Go to the Product Support Page of Cisco Container Platform.

Step 2 Under **Support Documentation And Software**, click **Download Software**.

The **Software Download** page appears displaying the latest release assets.

Step 3 Log in using your Cisco username and password that is associated with a valid service contract.

Step 4 Download the Installer and Tenant images.

Unpacking the Software

Step 1 Browse to the directory where you have downloaded the software.

Step 2 Open the Shell command prompt and extract each `tar.gz` file.

Example

```
$ tar -zxvf kcp-vm-$VERSION.tar.gz
kcp-vm-$VERSION/
kcp-vm-$VERSION/ee.pem
kcp-vm-$VERSION/ccp_image_signing_release_v1_pubkey.der
kcp-vm-$VERSION/root_ca.pem
kcp-vm-$VERSION/kcp-vm-$VERSION.ova.signature
kcp-vm-$VERSION/kcp-vm-$VERSION.ova
kcp-vm-$VERSION/verify
kcp-vm-$VERSION/sub_ca.pem
kcp-vm-$VERSION/README
```

The `.ova` file contains the Cisco Container Platform image.

Verifying the Software

Before you begin

Ensure that your system has python 3.5.2 or later and OpenSSL installed.

Step 1 Browse to the directory where you have unpacked the software.

Step 2 Open the Shell command prompt and run the script to verify the software.

Note You must run the verification steps for each release image.

Example

```
$ ./verify --type release --signature kcp-vm-$VERSION.ova.signature --image kcp-vm-$VERSION.ova
Verifying sha512 hash of ./root_ca.pem
Successfully verfied sha512 hash of ./root_ca.pem
Verifying sha512 hash of ./sub_ca.pem
Successfully verfied sha512 hash of ./sub_ca.pem
Verifying root(./root_ca.pem) and subca(./sub_ca.pem)
Successfully verified root and subca.
Verifying cert(./ee.pem) against root(./root_ca.pem) and subca(./sub_ca.pem)
Successfully verified end entity cert.
Extracting pubkey(kcp-vm-$VERSION/ee.pubkey) from ./ee.pem
Successfully extrated public key to kcp-vm-$VERSION/ee.pubkey.
Verifying signature(kcp-vm-$VERSION.ova.signature) of kcp-vm-$VERSION.ova using
kcp-vm-$VERSION/ee.pubkey
Successfully verified signature.
```

CHAPTER 2

Installing Cisco Container Platform on vSphere Web Client

This chapter contains the following topics:

- Installing Cisco Container Platform, on page 15
- Upgrading Cisco Container Platform, on page 24
- Uninstalling Cisco Container Platform, on page 26
- Backing Up and Restoring Cisco Container Platform, on page 27

Installing Cisco Container Platform

Installing Cisco Container Platform is a three-step process:

- Importing Cisco Container Platform Tenant Base VM

  The Cisco Container Platform tenant base VM contains the container image and the files that are necessary to create the tenant Kubernetes clusters that are used for configuring monitoring, logging, container network interfaces (CNI), and persistent volumes.

- Deploying Installer VM, on page 18

  The Installer VM contains the VM image and the files for installing other components such as Kubernetes and the Cisco Container Platform application.

- Deploying Cisco Container Platform, on page 20

  The Cisco Container Platform Control Plane is set up using an installer UI. After the installer VM is switched on, the URL of the installer appears on the vCenter **Web console**.

Importing Cisco Container Platform Tenant Base VM

Before you begin

- Ensure that you have configured the storage and networking requirements. For more information, see Storage Requirements, on page 7 and Network Requirements, on page 9.

- Ensure that vSphere has an Enterprise Plus license, which supports DRS and vSphere HA.

Step 1 Log in to the **VMware vSphere Web Client** as an administrator.

Step 2 In the **Navigation** pane, right-click the cluster on which you want to deploy Cisco Container Platform, and then choose **Deploy OVF Template**.
The **Deploy OVF Template** wizard appears.

Step 3 In the **Select template** screen, perform these steps:

a) Click the **URL** radio button, and enter the URL of the Cisco Container Platform Tenant OVA.

   Alternatively, click the **Local file** radio button, and browse to the location where the Cisco Container Platform tenant OVA is saved on your computer.

   Note The format of the Tenant OVA filename is as follows:

   ```
   ccp-tenant-image-x.y.z-ubuntuXX-a.b.c.ova
   ```

   Where `x.y.z` corresponds to the version of Kubernetes and `a.b.c` corresponds to the version of Cisco Container Platform.

   The Version Mapping Table, on page 35 provides the Cisco Container Platform version, Kubernetes version and image names mapping for each release.

b) Click **Next**.

Step 4 In the **Select name and location** screen, perform these steps:

a) In the **Name** field, enter a name for the Cisco Container Platform tenant base VM.

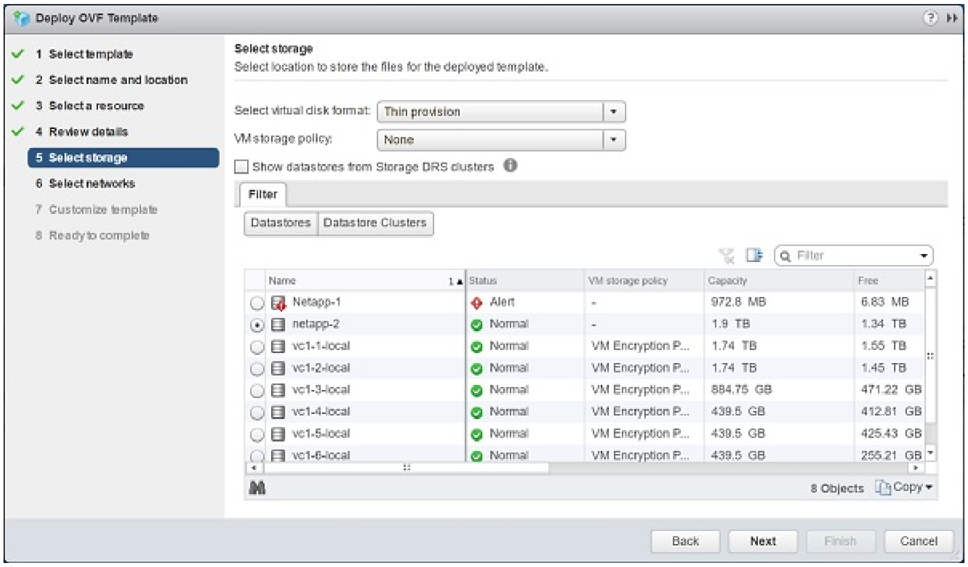
   Note You need to note down the Cisco Container Platform tenant base VM name as you will need to specify it while creating a cluster.

b) In the **Browse** tab, choose the data center where you want to deploy Cisco Container Platform.
c) Click **Next**.

Step 5 In the **Select a resource** screen, choose a cluster where you want to run the Cisco Container Platform tenant base VM, and then click **Next**.

Step 6 In the **Review details** screen, verify the Cisco Container Platform tenant base VM details, and then click **Next**.
The **Select storage** screen appears.

Figure 6: Select Storage Screen



Deploy OVF Template

1 Select template
2 Select name and location
3 Select a resource
4 Review details
5 Select storage
6 Select networks
7 Customize template
8 Ready to complete

Select storage
Select location to store the files for the deployed template.

Select virtual disk format: Thin provision
VM storage policy: None
☐ Show datastores from Storage DRS clusters

Filter
Datastores | Datastore Clusters

	Name	Status	VM storage policy	Capacity	Free
○	Netapp-1	Alert	-	972.8 MB	6.83 MB
●	netapp-2	Normal	-	1.9 TB	1.34 TB
○	vc1-1-local	Normal	VM Encryption P...	1.74 TB	1.55 TB
○	vc1-2-local	Normal	VM Encryption P...	1.74 TB	1.45 TB
○	vc1-3-local	Normal	VM Encryption P...	884.75 GB	471.22 GB
○	vc1-4-local	Normal	VM Encryption P...	439.5 GB	412.81 GB
○	vc1-5-local	Normal	VM Encryption P...	439.5 GB	425.43 GB
○	vc1-6-local	Normal	VM Encryption P...	439.5 GB	255.21 GB

8 Objects Copy

Step 7 In the **Select storage** screen, perform these steps:

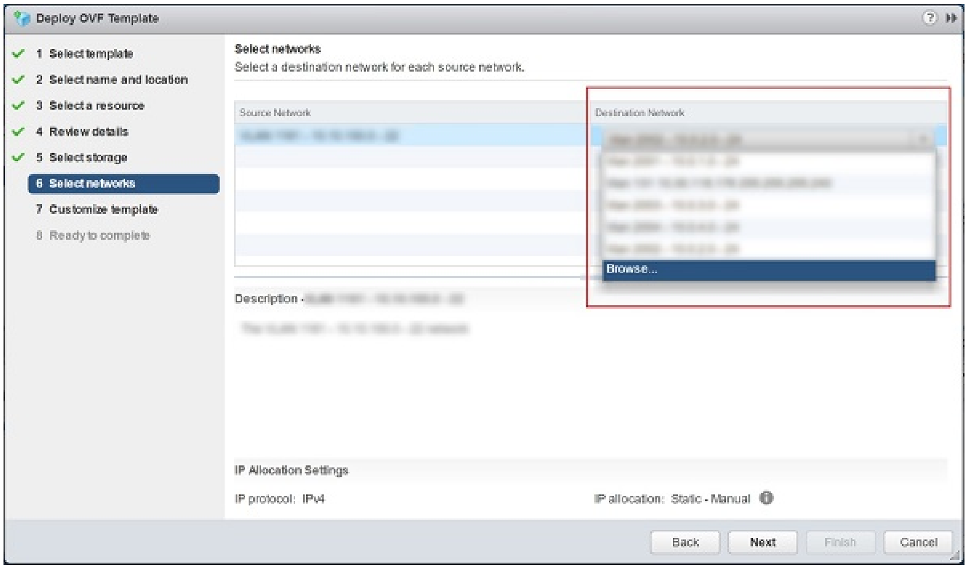
a) From the **Select virtual disk format** drop-down list, choose **Thin Provision** to allocate storage on demand.

b) In the **Filters** tab, choose a destination datastore for the Cisco Container Platform tenant base VM.

c) Click **Next**.

The **Select networks** screen appears.

Figure 7: Select Networks Screen



Deploy OVF Template

1 Select template
2 Select name and location
3 Select a resource
4 Review details
5 Select storage
6 Select networks
7 Customize template
8 Ready to complete

Select networks
Select a destination network for each source network.

Source Network	Destination Network
[illegible]	[illegible]

Description -

IP Allocation Settings
IP protocol: IPv4
IP allocation: Static - Manual

Step 8 In the **Select networks** screen, perform these steps:

a) From the **Destination Network** column, choose a network for each source network that is available in the Cisco Container Platform tenant base VM.

b) Click **Next**.

Step 9 In the **Customize template** screen, click **Next**.

Step 10 In the **Ready to complete** screen, verify the Cisco Container Platform tenant base VM settings, and then click **Finish**. The Cisco Container Platform tenant base VM import takes few minutes to complete.

Note You can leave the tenant base VM powered off and continue to Deploying Installer VM.

Deploying Installer VM

Before you begin

Note This deployment is for new installations of Cisco Container Platform. For upgrades, see Upgrading Cisco Container Platform, on page 24.

Ensure that you have imported the latest Cisco Container Platform tenant base VM to the vCenter instance. For more information, see Importing Cisco Container Platform Tenant Base VM, on page 15.

Step 1 Log in to the **VMware vSphere Web Client** as an administrator.

Step 2 In the **Navigation** pane, right-click the cluster on which you want to deploy Cisco Container Platform, and then choose **Deploy OVF Template**.
The **Deploy OVF Template** wizard appears.

Step 3 In the **Select template** screen, perform these steps:

a) Click the **URL** radio button, and enter the URL of the Installer OVA.

   Alternatively, click the **Local file** radio button, and browse to the location where the Installer OVA is saved on your computer.

   Note The format of the Installer OVA filename is as follows:

   ```
   kcp-vm-x.y.z.ova
   ```

   Where `x, y, z` corresponds to the major, minor, and patch release of Cisco Container Platform.

b) Click **Next**.

Step 4 In the **Select name and location** screen, perform these steps:

a) In the **Name** field, enter a name for the installer VM.
b) In the **Browse** tab, choose the data center where you want to deploy Cisco Container Platform.
c) Click **Next**.

Step 5 In the **Select a resource** screen, choose the cluster where you want to run the installer VM, and then click **Next**.

Step 6 In the **Review details** screen, verify the template details, and then click **Next**.

Step 7 In the **Select storage** screen, perform these steps:

a) From the **Select virtual disk format** drop-down list, choose **Thin Provision** to allocate storage on demand.
b) In the **Filters** tab, choose a destination datastore to store the installer VM.
c) Click **Next**.

Step 8 In the **Select networks** screen, perform these steps:

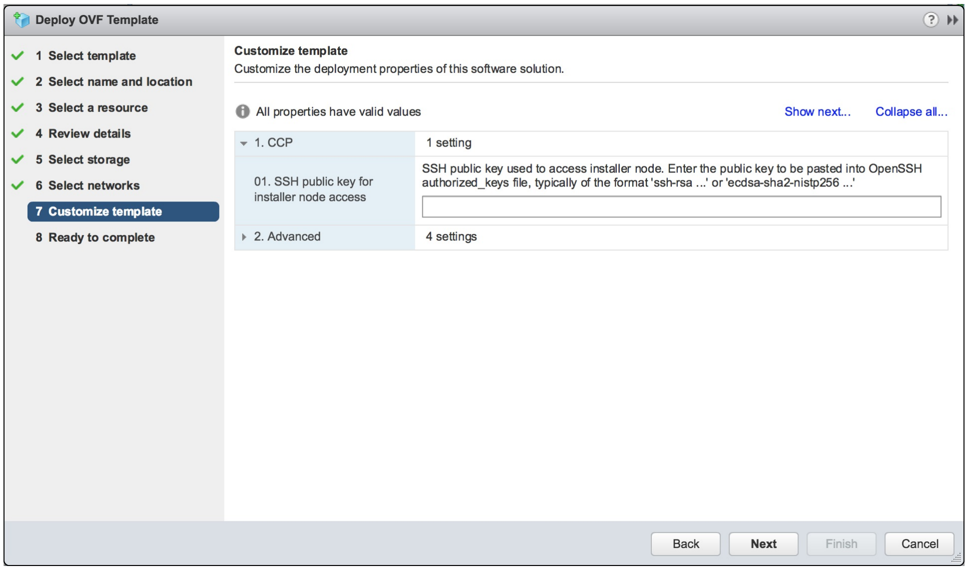
a) From the **Destination Network** column, choose a network for each source network that is available in the installer VM.

Note The selected network must have access to vCenter and the tenant VM networks.

b) Click **Next**.

The **Customize template** screen appears.

Figure 8: Customize Template Screen



Deploy OVF Template

- ✓ 1 Select template
- ✓ 2 Select name and location
- ✓ 3 Select a resource
- ✓ 4 Review details
- ✓ 5 Select storage
- ✓ 6 Select networks
- 7 Customize template
- 8 Ready to complete

Customize template
Customize the deployment properties of this software solution.

All properties have valid values

1. CCP	1 setting
01. SSH public key for installer node access	SSH public key used to access installer node. Enter the public key to be pasted into OpenSSH authorized_keys file, typically of the format 'ssh-rsa ...' or 'ecdsa-sha2-nistp256 ...'
2. Advanced	4 settings

Step 9 In the **Customize template** screen, enter the following optional parameters to customize the deployment properties:

a) Expand **CCP**, in the **SSH public key for installer node access** field, enter an ssh public key.

You can use this key to ssh to the installer VM.

Note
- Ensure that you enter the public key in a single line.
- If you do not have an SSH key pair, you can generate it using the **ssh-keygen** command.

b) Expand **Advanced**, in the **CIDR for kubernetes pod network** field, **192.168.0.0/16** is displayed as the default pod network CIDR of the Kubernetes cluster for the installer. If the CIDR IP addresses conflict with the tenant cluster VM network or the vCenter network, you need to set a different value for the CIDR.

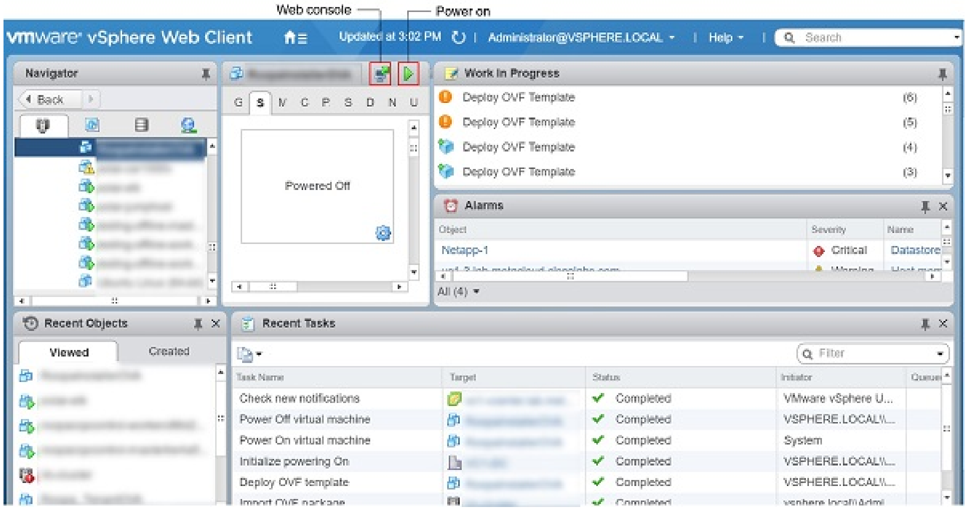
This CIDR is the single large CIDR from which smaller CIDRs are automatically allocated to each node for allocating IP addresses to the pods in the Kubernetes cluster. For more information, refer to https://kubernetes.io/docs/setup/scratch/#network-connectivity.

c) Click **Next**.

Step 10 In the **Ready to complete** screen, verify the installer VM deployment settings, and then click **Finish**.

Step 11 Click the **Power on** button to switch on the VM.

Figure 9: Switching on Installer VM



Work In Progress

Deploy OVF Template	(6)
Deploy OVF Template	(5)
Deploy OVF Template	(4)
Deploy OVF Template	(3)

Alarms

Object	Severity	Name
Netapp-1	Critical	Datastore

Recent Tasks

Task Name	Target	Status	Initiator	Queued
Check new notifications	[redacted]	Completed	VMware vSphere U...	
Power Off virtual machine	[redacted]	Completed	VSPHERE.LOCAL\\...	
Power On virtual machine	[redacted]	Completed	System	
Initialize powering On	[redacted]	Completed	VSPHERE.LOCAL\\...	
Deploy OVF template	[redacted]	Completed	VSPHERE.LOCAL\\...	
Import OVF package	[redacted]	Completed	vsphere.local\\Admi	

Once the installer VM is switched on, the installer UI takes a few minutes to become ready. You can view the status of the Installer UI using the Web console of vCenter. When the installer UI is ready, you can access it using the URL from the Web console.

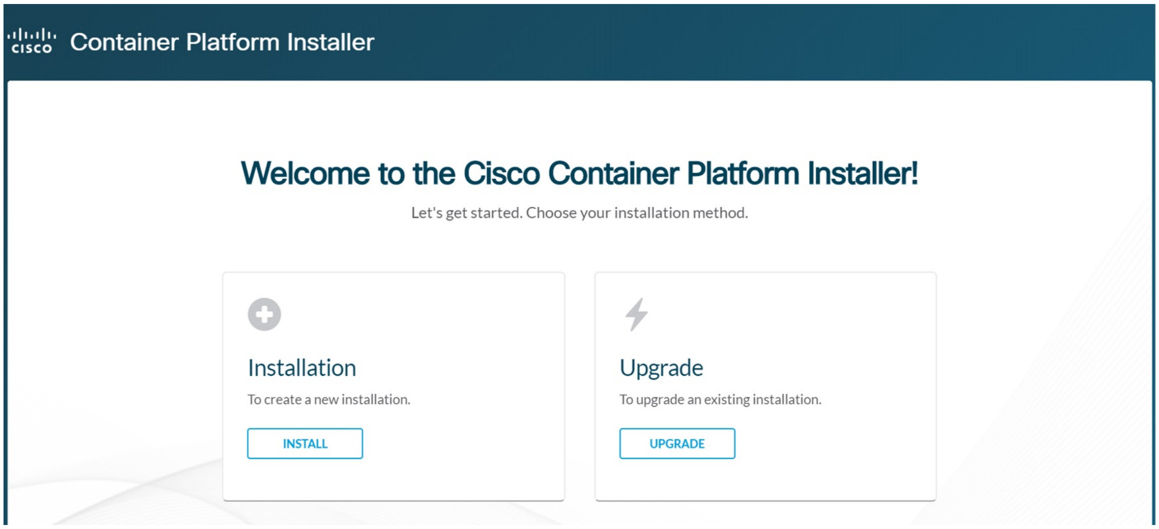
Deploying Cisco Container Platform

The Cisco Container Platform Control Plane is set up using an installer UI. After the installer VM is switched on, the URL of the installer appears on the vCenter **Web console**.

Step 1 Obtain the URL from the vCenter **Web console** and use a browser to open the installer UI.

The **Welcome** screen appears.

Figure 10: Welcome Screen



Welcome to the Cisco Container Platform Installer!

Let's get started. Choose your installation method.

Installation

To create a new installation.

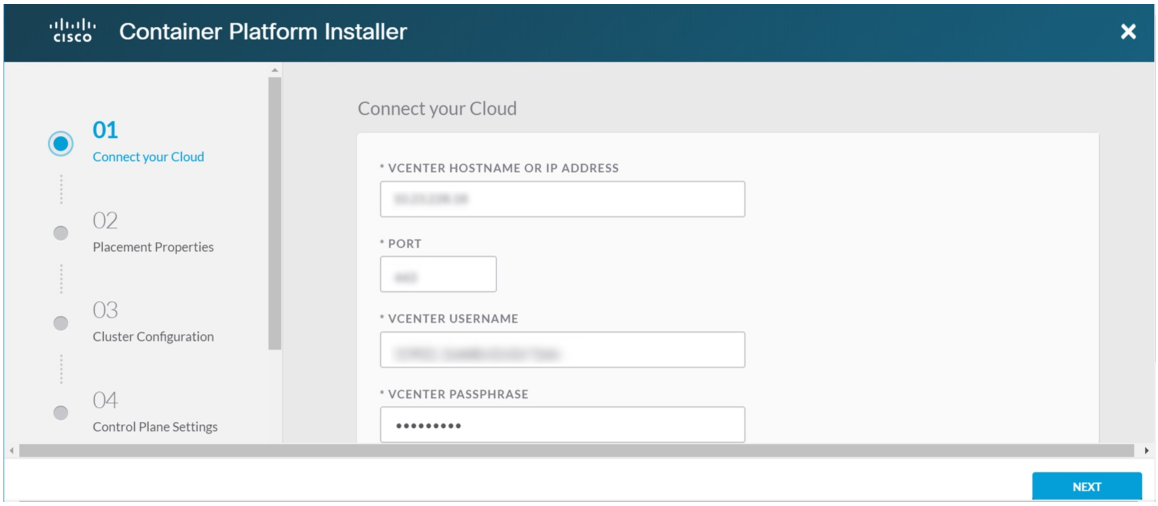
Upgrade

To upgrade an existing installation.

Step 2 Click **Install**.

The **Connect your Cloud** screen appears.

Figure 11: Connect your Cloud Screen



Container Platform Installer

Connect your Cloud

* VCENTER HOSTNAME OR IP ADDRESS

[redacted]

* PORT

[redacted]

* VCENTER USERNAME

[redacted]

* VCENTER PASSPHRASE

•••••••••

Step 3 In the **Connect your Cloud** screen, enter the following information:

a) In the **VCENTER HOSTNAME OR IP ADDRESS** field, enter the IP address of the vCenter instance that you want to use.

b) In the **PORT** field, enter the port number that your vCenter server uses.

Note The default port for vCenter is `443`.

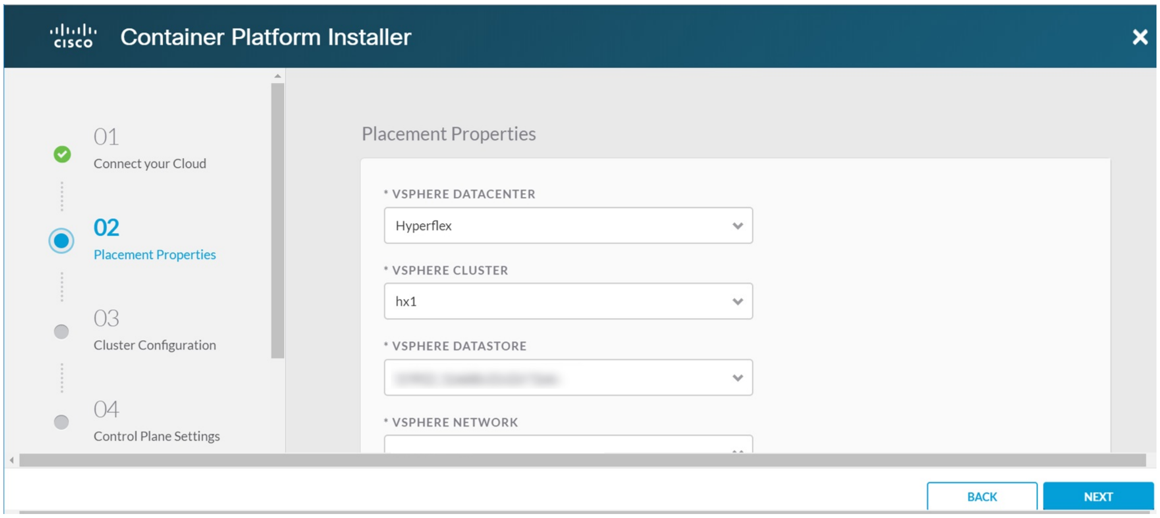
c) In the **VCENTER USERNAME** field, enter the username of the user with administrator access to the vCenter instance.

d) In the **VCENTER PASSPHRASE** field, enter the passphrase of the vCenter user.

e) Click **CONNECT**.

The **Placement Properties** screen appears.

Figure 12: Placement Properties Screen



Container Platform Installer

01 Connect your Cloud
02 Placement Properties
03 Cluster Configuration
04 Control Plane Settings

Placement Properties

* VSPHERE DATACENTER: Hyperflex

* VSPHERE CLUSTER: hx1

* VSPHERE DATASTORE: [redacted]

* VSPHERE NETWORK

Step 4 In the **Placement Properties** screen, enter the following information:

a) From the **VSPHERE DATACENTER** drop-down list, choose the datacenter.

b) From the **VSPHERE CLUSTER** drop-down list, choose the cluster.

c) From the **VSPHERE DATASTORE** drop-down list, choose the datastore.

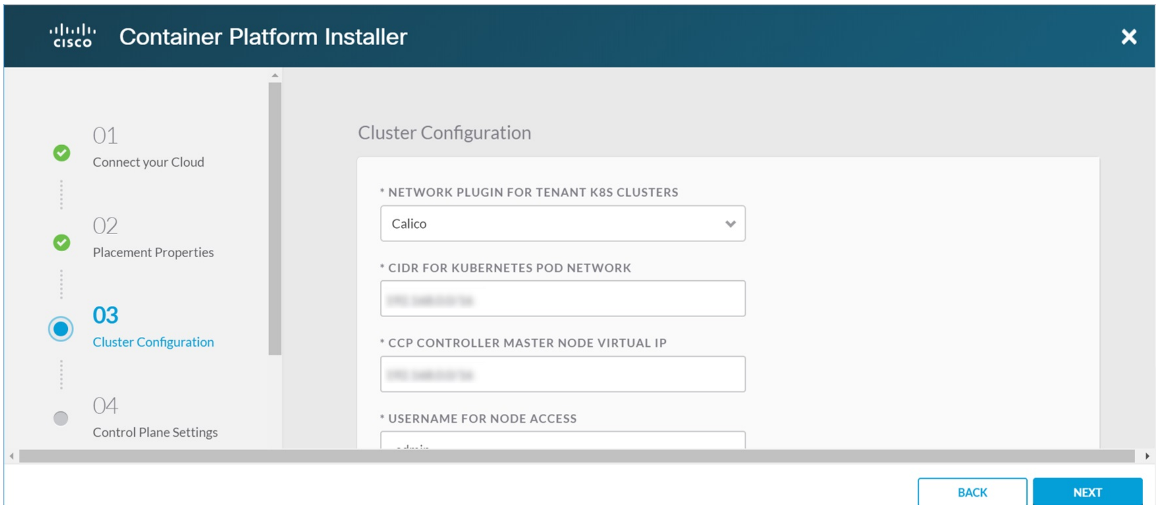
d) From the **VSPHERE NETWORK** drop-down list, choose the network.

e) In the **BASE VM IMAGE** field, enter the Cisco Container Platform tenant base VM name from Step 5 of the Importing Cisco Container Platform Tenant Base VM task.

f) Click **NEXT**.

The **Cluster Configuration** screen appears.

Figure 13: Cluster Configuration Screen



Container Platform Installer

01 Connect your Cloud
02 Placement Properties
03 Cluster Configuration
04 Control Plane Settings

Cluster Configuration

* NETWORK PLUGIN FOR TENANT K8S CLUSTERS: Calico

* CIDR FOR KUBERNETES POD NETWORK: [redacted]

* CCP CONTROLLER MASTER NODE VIRTUAL IP: [redacted]

* USERNAME FOR NODE ACCESS

Step 5 In the **Cluster Configuration** screen, enter the following information:

a) From the **NETWORK PLUGIN FOR TENANT K8S CLUSTERS** drop-down list, choose one of the following options for network connectivity:

- Calico
- ACI
- Contiv (Tech Preview)

Note For more information on the network plugins, see Container Network Interface Plugins, on page 4.

b) In the **CIDR FOR KUBERNETES POD NETWORK** field, **192.168.0.0/16** is displayed as the default pod network CIDR of the Kubernetes cluster for the installer. If the CIDR IP addresses conflict with the tenant cluster VM network or the vCenter network, you need to set a different value for the CIDR.

This CIDR is the single large CIDR from which smaller CIDRs are automatically allocated to each node for allocating IP addresses to the pods in the Kubernetes cluster. For more information, refer to https://kubernetes.io/docs/setup/scratch/#network-connectivity.

c) In the **CCP CONTROLLER MASTER NODE VIRTUAL IP** field, enter the IP address that is used to support a Cisco Container Platform upgrade.

This IP address needs to be in the same subnet, or it should be routable from the DHCP IP address for the controller VMs.

d) In the **USERNAME FOR NODE ACCESS** field, enter the username of the user who can ssh into the Cisco Container Platform Control Plane nodes.

e) In the **SSH PUBLIC KEY FOR INSTALLER NODE ACCESS** field, enter an ssh public key.

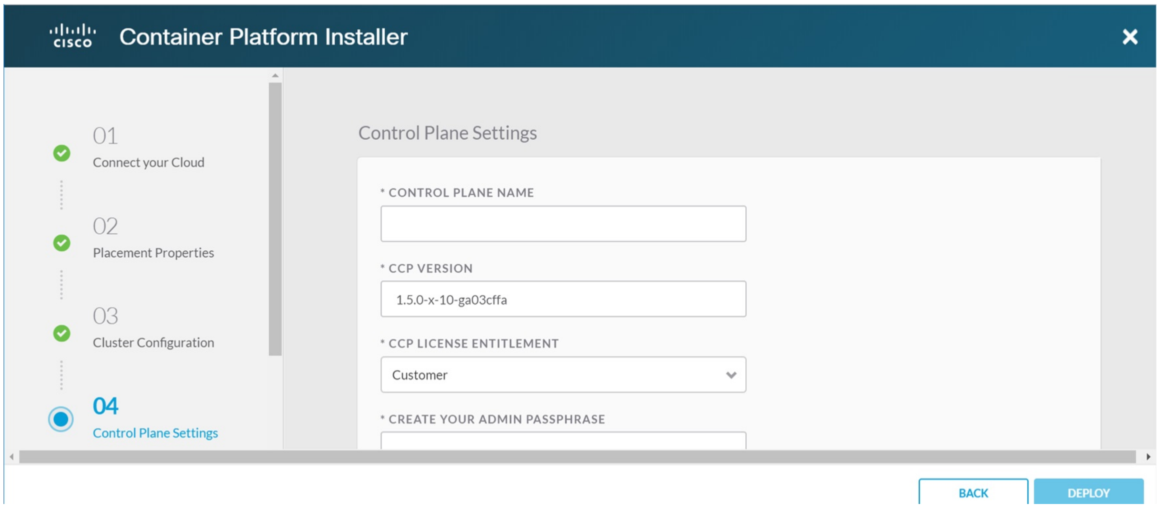
You can use this key to ssh to the Control Plane nodes.

Note
- Ensure that you enter the public key in a single line.
- If you do not have an SSH key pair, you can generate it using the **ssh-keygen** command.

f) Click **NEXT**.

The **Control Plane Settings** screen appears.

Figure 14: Control Plane Settings Screen



* CONTROL PLANE NAME

* CCP VERSION: 1.5.0-x-10-ga03cffa

* CCP LICENSE ENTITLEMENT: Customer

* CREATE YOUR ADMIN PASSPHRASE

Step 6 In the **Control Plane Settings** screen, enter the following information:

a) In the **CONTROL PLANE NAME** field, enter the name of the Cisco Container Platform cluster.

Note

- The cluster name must start with an alphanumeric character (a-z, A-Z, 0-9). It can contain a combination of hyphen (-) symbols and alphanumeric characters (a-z, A-Z, 0-9). The maximum length of the cluster name is 46 characters.
- Deployment of the installer VM fails if another Control Plane cluster with the same name already exists on the same datastore. You must ensure that you specify a unique name for the Control Plane cluster.

b) In the **CCP VERSION** field, enter the version of the Cisco Container Platform cluster.

c) From the **CCP LICENSE ENTITLEMENT** drop-down list, choose an entitlement option that indicates the type of Smart Licensing that you want to use.

Note The **Partner** option will only be used in conjunction with a **Not for Retail (NFR)** or **Trial** license.

d) In the **CREATE YOUR ADMIN PASSPHRASE** field, enter the passphrase you want to use for an **Administrator** user of the Cisco Container Platform Control Plane.

e) Expand **Advanced Settings**, in the **NTP SERVERS** field, enter the list of any NTP servers in your environment.

f) Click **DEPLOY** and then monitor the installation progress through the vCenter **Web console**.

Note You can use the ssh private key to access the Installer, control plane VMs, or the tenant cluster VMs. However, logging into these VMs using a username and password is not supported.

Upgrading Cisco Container Platform

Upgrading Cisco Container Platform and upgrading tenant clusters are independent operations. You must upgrade the Cisco Container Platform to allow tenant clusters to upgrade. Specifically, tenant clusters cannot be upgraded to a higher version than the Control Plane. For example, if the Control Plane is at version 1.10, the tenant cluster cannot be upgraded to the 1.11 version.

Upgrading Cisco Container Platform is a three-step process:

- Upgrading Cisco Container Platform Tenant Base VM, on page 24
- Deploying Upgrade VM, on page 25
- Upgrading Cisco Container Platform, on page 25

Note Taking a snapshot of the VMs managed by Cisco Container Platform is currently unsupported and results in failures during upgrades.

Upgrading Cisco Container Platform Tenant Base VM

You can follow the instructions in the Installing Cisco Container Platform > Importing Cisco Container Platform Tenant Base VM section.

Note The older tenant images are no longer required, you can delete them from your vCenter instance.

Deploying Upgrade VM

Follow the instructions in the Installing Cisco Container Platform > Deploying Installer VM section to deploy the latest VM.

It may take a few minutes for the deployment of the VM to complete. You can view the status of the upgrade task using the Web console of vCenter.

Note Depending on CNI usage, the port used to access Cisco Container Platform may change as part of the upgrade.

Upgrading Cisco Container Platform

The Cisco Container Platform Control Plane is upgraded using an installer UI. After the installer VM is switched on, the URL of the installer appears on the vCenter **Web console**.

Step 1 Obtain the URL from the vCenter **Web console** and use a browser to open the installer UI.

Step 2 Click **Upgrade**.

Step 3 In the **Connect your Cloud** screen, enter the following information:

a) In the **VCENTER HOSTNAME OR IP ADDRESS** field, enter the IP address of the vCenter instance that you want to use.

b) In the **PORT** field, enter the port of the vCenter instance that you want to use.

c) In the **VCENTER USERNAME** field, enter the username of the user with administrator access to the vCenter instance.

d) In the **VCENTER PASSPHRASE** field, enter the passphrase of the vCenter user.

e) Click **CONNECT**.

Step 4 In the **Placement Properties** screen, enter the following information:

a) In the **CISCO CONTAINER PLATFORM (CCP) URL** field, enter the URL for accessing Cisco Container Platform in the following format:

https://<*CCP_IP_Address*>:<*Port*>

b) From the **VSPHERE DATACENTER** drop-down list, choose the datacenter.

c) From the **BASE VM IMAGE** drop-down list, choose the Cisco Container Platform tenant base VM name.

d) Click **NEXT**.

Step 5 In the **Cluster Configuration** screen, enter the following information:

a) In the **CIDR FOR KUBERNETES POD NETWORK** field, **192.168.0.0/16** is displayed as the default pod network CIDR of the Kubernetes cluster for the installer. If the CIDR IP addresses conflict with the tenant cluster VM network or the vCenter network, you need to set a different value for the CIDR.

This CIDR is the single large CIDR from which smaller CIDRs are automatically allocated to each node for allocating IP addresses to the pods in the Kubernetes cluster. For more information, refer to https://kubernetes.io/docs/setup/scratch/#network-connectivity.

b) In the **USERNAME FOR NODE ACCESS** field, enter the username of the user who can ssh into the Cisco Container Platform Control Plane nodes.

c) In the **SSH PUBLIC KEY FOR INSTALLER NODE ACCESS** field, enter an ssh public key.

You can use this key to ssh to the Control Plane nodes.

Note
- Ensure that you enter the public key in a single line.
- You can use the private key to securely connect to the Cisco Container Platform Control Plane VMs through SSH, after installation.
- If you do not have an SSH key pair, you can generate it using the **ssh-keygen** command.

d) Click **NEXT**.

Step 6 In the **Control Plane Settings** screen, enter the following information:

a) In the **CONTROL PLANE NAME** field, enter the name of the Cisco Container Platform cluster.

Note You need to enter the same cluster name that you used during installation.

b) From the **CCP LICENSE ENTITLEMENT** drop-down list, choose an entitlement option that indicates the type of Smart Licensing that you want to use.

Note The **Partner** option will only be used in conjunction with a **Not for Retail (NFR)** or **Trial** license.

c) In the **ENTER CURRENT ADMIN PASSPHRASE** field, enter the current passphrase for an **Administrator** user of the Cisco Container Platform Control Plane.

d) Click **UPGRADE**.

Uninstalling Cisco Container Platform

Uninstalling Cisco Container Platform removes all containers and services associated with it. You will no longer be able to create or manage tenant clusters on this Cisco Container Platform instance.

Step 1 Open the Cisco Container Platform web interface, log in to the Control Plane cluster using its VIP address, and then delete all the Kubernetes tenant clusters that belong to the Cisco Container Platform instance.

For more information on deleting Kubernetes clusters, refer to the *Cisco Container Platform User Guide*.

Step 2 Follow these steps to delete the Control Plane and installer node VMs:

a) In the vSphere web client, right-click the VM, choose **Power** > **Power off**, and then click **Yes** in the confirmation dialog box.

b) Right-click each VM and choose **Delete from Disk**.

Step 3 Follow these steps to delete the Control Plane cluster data disks:

a) In the vSphere web client, choose **Home** > **Storage**.

b) From the left pane, choose the datastore that is used to install the Control Plane VMs. This is the same as the datastore to which the installer VM is imported to unless you have changed it in the installer UI.

c) If you have installed the Control Plane using the default name, right-click the folder name with the prefix **ccpcontrol** or if you have provided a different name to the Control Plane in the installer UI, right-click the folder with that name.

d) Choose **Delete File**.

Backing Up and Restoring Cisco Container Platform

This chapter contains the following topics:

Backing Up Cisco Container Platform

You can back up the Cisco Container Platform application data that pertains to the following components:

- Application users
- Virtualization providers
- Tenant clusters

Note The logging or monitoring data from Prometheus, Grafana, and the EFK stack is not included in the backup archive.

Backing Up Cisco Container Platform v1.5.0+

Step 1 Log in to the console of the master node of Cisco Container Platform Control Plane.

Step 2 Run the following command.

```
/ccp_related_files/percona_backup.sh ./backup.tar
```

Step 3 Copy the `backup.tar` backup archive to a secure location.

Note You must ensure that the backup archive is maintained securely as anyone with access to it has administrative capabilities on all tenant clusters.

Backing Up Cisco Container Platform v1.1.0-1.4.x

Step 1 Log in to the console of the master node of Cisco Container Platform Control Plane.

Step 2 Run the following commands.

```
    kubectl exec mysql-0 -- mkdir -p /tmp/backup
    kubectl exec -t mysql-0 -- bash -c "rm -Rf /tmp/backup/* && xtrabackup --backup
--target-dir=/tmp/backup  -p$(kubectl get secret mysql -o jsonpath='{.data.mysql-root-password}'|base64
 -d)    --alsologtostderr=true"
    kubectl exec mysql-0 -- tar -cvf /tmp/backup.tar /tmp/backup
    kubectl cp mysql-0:/tmp/backup.tar ./backup.tar
```

Note Depending on the memory usage of the database, any of the preceding commands may fail with an **ExitCode:137** error code. It is safe to run these commands multiple times until they succeed.

Step 3 Copy the `backup.tar` backup archive to a secure location.

Note You must ensure that the backup archive is maintained securely as anyone with access to it has administrative capabilities on all tenant clusters.

Restoring Cisco Container Platform

You can restore a valid backup to a new Cisco Container Platform Control Plane that has control over all existing Cisco Container Platform settings and tenant clusters.

Restoring Cisco Container Platform Control Plane data is slightly different from traditional restore methods. The data can be restored into a version of Cisco Container Platform newer than the version from which the backup was made.

For example, you may back up the data from a Cisco Container Platform Control Plane v1.4 installation, and then, as part of a restore or recovery process, restore that data into a new Cisco Container Platform Control Plane v1.5 installation.

You can restore data into any version of Cisco Container Platform v1.5 or later. For example, because upgrades are supported from v1.4 to v1.5, it is possible to restore a v1.4 backup into a new Cisco Container Platform v1.5 install.

Step 1 Power off the VMs that belong to the previous Control Plane instance.

Step 2 Install a new Cisco Container Platform Control Plane v1.5.0+.

Step 3 Copy the backup from the secure location to Control Plane master.

```
scp ./backup.tar <control_plane_master>:/tmp/backup.tar
```

Step 4 Log in to the console of the master node of Cisco Container Platform Control Plane.

Step 5 Run the following command.

```
/ccp_related_files/percona_restore.sh /tmp/backup.tar
```

APPENDIX A

Troubleshooting Cisco Container Platform

This appendix describes the problems that may occur during the installation and operation of Cisco Container Platform and the possible ways of resolving these problems.

It contains the following topics:

Unable to Deploy NGINX Ingress Controller Using Helm

Description	Error Message	Recommended Solution
Deploying the NGINX Ingress controller using Helm fails as RBAC is not configured in Helm.	It seems the cluster it is running with Authorization enabled (like RBAC) and there is no permissions for the ingress controller. Please check the configuration	As Cisco Container Platform uses RBAC for authentication, Helm also needs to be configured to use RBAC. Enable the RBAC parameter in Helm using the following command: `--set rbac.create=true`

Unable to Start NGINX Ingress Controller Pod

Description	Error Message	Recommended Solution
When kube-proxy is used, setting both the `controller.service.externalIPs` and `controller.hostNetwork` variables to **`true`** for the NGINX-Ingress chart results in an invalid configuration. Both kube-proxy and NGINX uses port 80 for communication, causing a port conflict, and the NGINX Ingress controller pod is set to the `CrashLoopBackOff` state.	Port 80 is already in use. Please check the flag --http-port	Ensure that both the `controller.service.externalIPs` and `controller.hostNetwork` variables are not set to **`true`** at the same time.

Unable to Power on Worker VMs after a Shutdown

Description	Error Message	Recommended Solution
Worker VMs may fail to power on after a shutdown.	File system specific implementation of LookupAndOpen[file] failed.	Follow these steps to resolve the problem: 1. In the left pane, click the VM that you want to power on. 2. In the right pane, from the **Actions** drop-down list, choose **Edit Settings**. The **Edit Settings** window displays the multiple hard disks of the VM. 3. Except for the primary hard disk (Hard disk 1), click each hard disk, and then click the **Remove** icon. **Note** Ensure that the **Delete files from datastore** check box is not checked. 4. Click **OK**.

Application Pods Crash When Using Contiv CNI in Tenant Clusters

When you use Contiv as the CNI for a tenant cluster, you need to ensure that the application pods that need HugePages must have the following section in the pod manifest. Otherwise, the pods may crash.

```
resources:
   limits:
     hugepages-2Mi: 512Mi
     memory: 512Mi
```

The preceeding section in the pod manifest limits 512 MB in memory for HugePages for the pod. It allocates 256 HugePages, with each HugePage having 2MB size.

HugePages are allocated to the pods only if you have enabled HugePages on the host. Otherwise, the HugePage allocation in the pod manifest is ignored by Kubernetes. The following table shows the Cisco Container Platform CNIs that use HugePages.

Cisco Container Platform CNI	Use HugePages
Contiv	Yes
ACI	No
Calico	No

Example of Allocating HugePages for Applications

Step 1 Check the total and free HugePages on the worker nodes. Each HugePage is 2048 KB in size.

```
$ grep -i huge /proc/meminfo
AnonHugePages: 0 kB
ShmemHugePages: 0 kB
HugePages_Total: 1024
HugePages_Free: 972
HugePages_Rsvd: 0
HugePages_Surp: 0
Hugepagesize: 2048 kB

$ sudo sysctl -a | grep -i huge
vm.hugepages_treat_as_movable = 0
vm.hugetlb_shm_group = 0
vm.nr_hugepages = 1024
vm.nr_hugepages_mempolicy = 1024
vm.nr_overcommit_hugepages = 0
```

Step 2 If the host has less HugePages, increase the HugePages allocation.

```
sudo su
echo 2048 > /proc/sys/vm/nr_hugepages

# Check the increased number of HugePages
cat /proc/sys/vm/nr_hugepages
grep -i huge /proc/meminfo
sudo sysctl -a | grep -i huge
```

Note You need to perform these steps on all the hosts.

Step 3 Create the `bookinfo.yaml` file that allocates HugePages to the `reviews-v1` pod.

```
apiVersion: extensions/v1beta1
kind: Deployment
metadata:
name: reviews-v1
spec:
template:
    metadata:
    labels:
        app: reviews
        version: v1
    spec:
    containers:
    - name: reviews
        image: istio/examples-bookinfo-reviews-v1:1.5.0
        imagePullPolicy: IfNotPresent
        resources:
        limits:
            hugepages-2Mi: 512Mi
            memory: 512Mi
        ports:
        - containerPort: 9080
```

Step 4 Deploy `bookinfo.yaml` and check usage of HugePages.

```
$ kubectl create -f istio-$ISTIO_VERSION/samples/bookinfo/kube/bookinfo.yaml
deployment.extensions "reviews-v1" created

$ kubectl get pods | grep reviews
reviews-v1-6f56455f68-t6phs                                 1/1       Running   0          3m

# Check usage of HugePages by the pods
$ kubectl describe pod reviews-v1-6f56455f68-t6phs | grep -i '^Name:\|Image:\|huge\|mem'
Name:           reviews-v1-6f56455f68-t6phs
    Image:          istio/examples-bookinfo-reviews-v1:1.5.0
    hugepages-2Mi:  512Mi
    memory:         512Mi
    hugepages-2Mi:  512Mi
    memory:         512Mi

# Check usage of HugePages on each host
$ grep -i huge /proc/meminfo
AnonHugePages:         0 kB
ShmemHugePages:        0 kB
HugePages_Total:    1024
HugePages_Free:      972
HugePages_Rsvd:        0
HugePages_Surp:        0
Hugepagesize:       2048 kB

$ sudo sysctl -a | grep -i huge
vm.hugepages_treat_as_movable = 0
vm.hugetlb_shm_group = 0
vm.nr_hugepages = 1024
vm.nr_hugepages_mempolicy = 1024
vm.nr_overcommit_hugepages = 0
```

Step 5 Check the decrease of the `HugePages_Free` field in the output when the `reviews-v1` pod is using HugePages.

```
grep  -i huge /proc/meminfo
```

APPENDIX B

Version Mapping Table

Cisco Container Platform Version	Kubernetes Version	Image Names
1.0.0	1.10	Control Plane Installer - kcp-vm-1.0.0.ova Tenant Image - ccp-tenant-image-1.10.1-1.0.0.ova
1.0.1	1.10	Control Plane Installer - kcp-vm-1.0.1.ova Tenant Image - ccp-tenant-image-1.10.1-1.0.1.ova
1.1.0	1.10	Control Plane Installer - kcp-vm-1.1.0.ova Tenant Image - ccp-tenant-image-1.10.1-1.1.0.ova
1.4.0	1.10	Control Plane Installer - kcp-vm-1.4.0.ova Tenant Image - ccp-tenant-image-1.10.1-1.4.0.ova
1.5.0	1.10	Control Plane Installer - kcp-vm-1.5.0.ova Tenant Image - ccp-tenant-image-1.10.1-ubuntu16-1.5.0.ova
2.0.1	1.10 1.11	Control Plane Installer - kcp-vm-2.0.1.ova Tenant Image (Kubernetes 1.10) - ccp-tenant-image-1.10.1-ubuntu16-2.0.0.ova Tenant Image (Kubernetes 1.11) - ccp-tenant-image-1.11.3-ubuntu18-2.0.0.ova
2.1.0	1.10 1.11	Control Plane Installer - kcp-vm-2.1.0.ova Tenant Image (Kubernetes 1.10) - ccp-tenant-image-1.10.1-ubuntu16-2.1.0.ova Tenant Image (Kubernetes 1.11) - ccp-tenant-image-1.11.3-ubuntu18-2.1.0.ova

Note We recommend that you use the latest Kubernetes version ova for the installation.