



## **Cisco Container Platform 1.0 User Guide**

**First Published:** 2018-05-22

### **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.

The Cisco implementation of TCP header compression is an adaptation of a program developed by the University of California, Berkeley (UCB) as part of UCB's public domain version of the UNIX operating system. All rights reserved. Copyright © 1981, Regents of the University of California.

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 NON-INFRINGEMENT 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.

Cisco and the Cisco logo are trademarks or registered trademarks of Cisco and/or its affiliates in the U.S. and other countries. To view a list of Cisco trademarks, go to this URL: <https://www.cisco.com/go/trademarks>. Third-party trademarks mentioned 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. (1721R)

© 2018 Cisco Systems, Inc. All rights reserved.



## CONTENTS

---

### CHAPTER 1

#### **Cisco Container Platform 1**

- Administrator Workflow 1
- User Workflow 2
- Accessing Cisco Container Platform Web Interface 3
- Setting Up Cisco Container Platform 3
- Configuring Cisco Smart Software Licensing 4
  - Workflow of Cisco Smart Software Licensing 5
  - Generating Registration Token 5
  - Configuring Transport Settings 6
  - Registering Cisco Container Platform License 7
  - Renewing Authorization 7
  - Reregistering Cisco Container Platform License 8
  - Deregistering Registration 8

---

### CHAPTER 2

#### **Managing Cisco Container Platform Infrastructure Configuration 9**

- Managing Provider Profile 9
  - Adding Provider Profile 9
  - Modifying Provider Profile 10
  - Deleting Provider Profile 10
- Managing ACI-CNI Profile 10
  - Adding ACI-CNI Profile 10
  - Modifying ACI-CNI Profile 11
  - Deleting ACI-CNI Profile 11
- Managing VIP Pool 11
  - Adding VIP Pool 11
  - Modifying VIP Pool 12

Deleting VIP Pool 12

---

**CHAPTER 3**

**Administering Kubernetes Clusters 13**

- Creating Kubernetes Clusters 13
- Upgrading Kubernetes Clusters 14
- Scaling Kubernetes Clusters 14
- Deleting Kubernetes Clusters 15
- Managing Users and RBAC 15
  - Configuring Local Users 15
  - Changing Login Password 15
  - Configuring AD Servers 16
  - Configuring AD Groups 16
- Monitoring Health of Cluster Deployments 16
- Monitoring Logs from Cluster Deployments 17
  - Viewing EFK Logs Using Kibana (Tenant Cluster) 18
  - Viewing EFK Logs Using Kibana (Control Plane Cluster) 18
  - Forwarding Logs to External Elasticsearch Server 19

---

**CHAPTER 4**

**Managing Kubernetes Clusters 21**

- Setting up Kubernetes Dashboard 21

---

**CHAPTER 5**

**Services and Networking 23**

- Load Balancing Kubernetes Services using NGINX 23
  - Types of Ingress 23
- Network Policies 26

---

**CHAPTER 6**

**Deploying Applications on Kubernetes Clusters 27**

- Workflow of Deploying Applications 27
- Downloading Kubeconfig File 27
- Sample Scenarios 28
  - Deploying a Pod with Persistent Volume 28
  - Deploying Cafe Application with Ingress 29

---

**APPENDIX A**

**User Privileges on vSphere 33**

User Privileges on vSphere 33





# 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.

Cisco Container Platform provides authentication and authorization, security, high availability, networking, load balancing, and operational capabilities to effectively operate and manage Kubernetes clusters. Cisco Container Platform also provides a validated configuration of Kubernetes and can integrate with underlying infrastructure components such as [Cisco HyperFlex](#) and [Cisco ACI](#). The infrastructure provider for Cisco Container Platform is Hyperflex.

Using the Cisco Container Platform web interface, you can create Kubernetes clusters on which you can deploy containerized applications. The clusters are created on the infrastructure provider platform.

The two user personas in Cisco Container Platform are as follows:

- The **Administrator** persona, which is associated with the **Administrator** role.
- The **User** persona, which is associated with the **User** role.

This chapter contains the following topics:

- [Administrator Workflow, on page 1](#)
- [User Workflow, on page 2](#)
- [Accessing Cisco Container Platform Web Interface, on page 3](#)
- [Setting Up Cisco Container Platform, on page 3](#)
- [Configuring Cisco Smart Software Licensing, on page 4](#)

## Administrator Workflow

The following table lists the workflow for Cisco Container Platform administrators.

Task	Related Section
Access the Cisco Container Platform web interface with <i>Administrator</i> credentials.	<a href="#">Accessing Cisco Container Platform Web Interface, on page 3</a>
Set up the Cisco Container Platform infrastructure configuration.	<a href="#">Setting Up Cisco Container Platform, on page 3</a>

Task	Related Section
Configure Cisco Smart Software Licensing for your Cisco Container Platform instance.	<a href="#">Configuring Cisco Smart Software Licensing, on page 4</a>
Manage the Cisco Container Platform infrastructure configurations using which clusters are created.	<a href="#">Managing Cisco Container Platform Infrastructure Configuration, on page 9</a>
Create Kubernetes clusters.	<a href="#">Creating Kubernetes Clusters, on page 13</a>
Add users, assign appropriate roles, and associate the new users to the Kubernetes clusters that you have created.	<a href="#">Managing Users and RBAC, on page 15</a>
Monitor Kubernetes clusters.	<a href="#">Monitoring Health of Cluster Deployments, on page 16</a> <a href="#">Viewing EFK Logs Using Kibana (Tenant Cluster), on page 18</a>
Manage Kubernetes cluster using the Kubernetes Dashboard.	<a href="#">Managing Kubernetes Clusters, on page 21</a>
Manage the lifecycle of Kubernetes clusters by scaling or upgrading the clusters.	<a href="#">Scaling Kubernetes Clusters, on page 14</a> <a href="#">Upgrading Kubernetes Clusters, on page 14</a>

## User Workflow

The following table lists the workflow for developers assigned with the *User* role.

Task	Related Section
Access the Cisco Container Platform web interface with user credentials.	<a href="#">Accessing Cisco Container Platform Web Interface, on page 3</a>
Monitor Kubernetes clusters that are assigned to the user.	<a href="#">Monitoring Health of Cluster Deployments, on page 16</a> <a href="#">Viewing EFK Logs Using Kibana (Tenant Cluster), on page 18</a>
Manage the assigned Kubernetes clusters using the Kubernetes Dashboard or CLI.	<a href="#">Managing Kubernetes Clusters, on page 21</a>
Deploy applications on the assigned Kubernetes clusters.	<a href="#">Deploying Applications on Kubernetes Clusters, on page 27</a>



# Accessing Cisco Container Platform Web Interface

## Before you begin

Ensure that you have configured the prerequisites for integrating ACI with Cisco Container Platform.

For more information, refer to the following documents:

- [ACI Integration Requirements](#) section of the *Cisco Container Platform 1.0 Installation Guide*
- [Planning](#) and [Prerequisites](#) section of the Cisco ACI and Kubernetes Integration page

Ensure that you have powered on the Control Plane VMs on the infrastructure provider platform.

---

**Step 1** Access the following URL using your web browser:

```
https://<Cisco Container Platform IP Address>
```

**Note** We recommend that you use the Chrome, Safari, or Firefox browser to access the URL.

**Step 2** Log in to the web interface as an admin user using the password given during the Cisco Container Platform installation.

---

# Setting Up Cisco Container Platform



---

**Note** This topic is applicable only for an ACI environment. In a non-ACI environment, the IP address range of the default VIP pool must be expanded to include the additional VIPs for tenant clusters. For more information, see [Managing VIP Pool, on page 11](#).

---

When you log in to Cisco Container Platform for the first time, you need to configure the Cisco Container Platform initial setup using the **Cisco Container Platform Setup** wizard.

---

**Step 1** On the **Welcome** page, click **START THE SETUP**.

**Step 2** In the **Infrastructure Provider** screen, specify information such as description, IP address, port, username, and password of the provider profile, and then click **CONNECT**.

**Step 3** In the **ACI CNI Credentials** screen, specify information such as IP address, username, and password of the APIC instance, and then click **NEXT**.

**Step 4** In the **ACI CNI Configuration** screen, perform these steps:

- In the **ACI VIRTUAL MACHINE MANAGER (VMM) DOMAIN NAME** field, type the Virtual Machine Manager Domain (VMMD) that you want to use.
- In the **ACI INFRASTRUCTURE VLAN ID** field, type the VLAN number for layer 2 networking.
- In the **ACI VIRTUAL ROUTING & FORWARDING (VRF) NAME** field, type the Virtual Routing and Forwarding (VRF) IP address.
- In the **L3OUT** field, type the ACI object for allowing external internet connectivity.

- e) In the **L3OUT EXTERNAL NETWORKS** field, type the external network that is reachable through the L3OUT object.
- f) In the **AAEP NAME** field, type an Attachable Access Entity Profile (AAEP) name to associate the VMM domain to an AAEP.
- g) In the **NAMESERVERS** field, enter the IP address of all the DNS servers that the ACI fabric can access.
- h) In the **NODE VLAN START ID** field, enter the starting IP address for the IP pool that is used to allocate IP addresses to the node VLAN.
- i) In the **NODE VLAN END ID** field, enter the ending IP address for the IP pool that is used to allocate IP addresses to the node VLAN.
- j) In the **STARTING SUBNET FOR PODS** field, enter the starting IP address for the IP pool that is used to allocate IP addresses to the pods.
- k) In the **STARTING SUBNET FOR SERVICE** field, enter the starting IP address for the IP pool that is used to allocate IP addresses to the service VLAN.
- l) In the **OPFLEX MULTICAST RANGE** field, enter a range for the Opflex multicast.
- m) In the **CONTROL PLANE CONTRACT NAME** field, enter the name of the contract that allows traffic from the Control Plane cluster to the tenant cluster.
- n) Click **CONNECT**.

**Step 5**

In the **Summary** screen, verify the configuration, and then click **FINISH**.

For more information on adding, modifying, or deleting the Cisco Container Platform infrastructure configuration parameters, see [Managing Cisco Container Platform Infrastructure Configuration, on page 9](#).

## Configuring Cisco Smart Software Licensing

You need to configure Cisco Smart Software Licensing to easily procure, deploy, and manage licenses for your Cisco Container Platform instance. The number of licenses required depends on the number of VMs necessary for your deployment scenario.

A Cisco Container Platform instance is available for a 90-day evaluation period after which, you need to register with Cisco Smart Software Manager (Cisco SSM). Cisco SSM enables you to manage your Cisco Smart Software Licenses from one centralized website. With Cisco SSM, you can organize and view your licenses in groups called virtual accounts. You can also use Cisco SSM to transfer the licenses between virtual accounts as needed.

You can access Cisco SSM from the Cisco Software Central homepage at [software.cisco.com](http://software.cisco.com), under the **Smart Licensing** area.

If you do not want to manage licenses using Cisco SSM, either for policy reasons or network availability reasons, you can choose to install Cisco SSM satellite at your premises. Cisco Container Platform registers and reports license consumption to the Cisco SSM satellite as it does to Cisco SSM.

**Note**

Ensure that you use Cisco SSM Satellite version 5.0 or later. For more information on installing and configuring Cisco SSM satellite, refer to <http://www.cisco.com/go/smartsatellite>.

## License Usage and Compliance

Once you register Cisco Container Platform with Cisco SSM, you will receive the **Cisco Container Platform License with Support** license.

Cisco SSM or Cisco SSM satellite totals the license requirements for all your Cisco Container Platform instances and compares the total license usage to the number of licenses purchased, on a daily basis. After the data synchronization, your Cisco Container Platform instance displays one of the following status indicators:

- **Authorized**, when the number of licenses purchased is sufficient
- **Out of Compliance**, when the number of licenses is insufficient
- **Authorization Expired**, when the product has not communicated with Cisco SSM or Cisco SSM satellite for a period of 90 days.

## Workflow of Cisco Smart Software Licensing

The following table describes the workflow of Cisco Smart Software Licensing:

Task	Related Section
Generate a product instance registration token in your virtual account	<a href="#">Generating Registration Token, on page 5</a>
Configure the transport settings using which Cisco Container Platform connects to Cisco SSM or Cisco SSM satellite	<a href="#">Configuring Transport Settings, on page 6</a>
Register the Cisco Container Platform instance with Cisco SSM or Cisco SSM satellite	<a href="#">Registering Cisco Container Platform License, on page 7</a>
Manage licenses	<a href="#">Renewing Authorization, on page 7</a> <a href="#">Reregistering Cisco Container Platform License, on page 8</a> <a href="#">Deregistering Registration, on page 8</a>

## Generating Registration Token

You need to generate a registration token from Cisco SSM or Cisco SSM satellite to register the Cisco Container Platform instance.

### Before you begin

Ensure that you have set up a Smart Account and a Virtual account on Cisco SSM or Cisco SSM satellite.

**Step 1** Log in to your Smart Account on [Cisco SSM](#) or Cisco SSM satellite.

**Step 2** Navigate to the Virtual account using which you want to register the Cisco Container Platform instance.

**Step 3** If you want to enable higher levels of encryption for the products registered using the registration token, check the **Allow export-controlled functionality on the products registered with this token** check box.

**Note** This option is available only if you are compliant with the Export-Controlled functionality.

**Step 4** Click **New Token** to generate a registration token.

**Step 5** Copy and save the token for using it when you register your Cisco Container Platform instance. For more information on registering your Cisco Container Platform instance, see [Registering Cisco Container Platform License, on page 7](#).

---

## Configuring Transport Settings

By default, Cisco Container Platform directly communicates with the Cisco SSM. You can modify the mode of communication by configuring the transport settings.

### Before you begin

Ensure that you have obtained the registration token for the Cisco Container Platform instance.

---

**Step 1** Log in to the Cisco Container Platform web interface.

**Step 2** From the left pane, click **Licensing**.

If you are running Cisco Container Platform in the Evaluation mode, a license notification is displayed on the **Smart Software Licensing** pane.

**Step 3** If a license notification is displayed, click the **edit the Smart Call Home Transport Settings** link. Alternatively, click the **Licensing Status** tab, and then click the **View/Edit** link that appears under **Transport Settings**.

**Step 4** In the **Transport Settings** dialog box, perform one of these steps:

- To configure Cisco Container Platform to send the license usage information to Cisco SSM using the Internet:
  1. Click the **DIRECT** radio button.
  2. Configure a DNS on Cisco Container Platform to resolve *tools.cisco.com*.

This is the default setting.

- To configure Cisco Container Platform to send the license usage information to Cisco SSM using the Cisco SSM satellite:
  1. Click the **TRANSPORT GATEWAY** radio button.
  2. Enter the URL of the Cisco SSM satellite.
- To configure Cisco Container Platform to send the license usage information to Cisco SSM using a proxy server. For example, an off-the-shelf proxy, such as Cisco Transport Gateway or Apache:
  1. Click the **HTTP/HTTPS PROXY** radio button.
  2. Enter the IP address and port number of the proxy server.

**Step 5** Click **SAVE**.

---

## Registering Cisco Container Platform License

You need to register your Cisco Container Platform instance with Cisco SSM or Cisco SSM satellite before the 90-day evaluation period expires.

### Before you begin

Ensure that you have configured the transport settings.

---

**Step 1** Log in to the Cisco Container Platform web interface.

**Step 2** From the left pane, click **Licensing**.

**Step 3** In the license notification, click **Register**.  
The **Smart Software Licensing Product Registration** dialog box appears.

**Step 4** In the **Product Instance Registration Token** field, copy and paste the registration token that you generated using the Cisco SSM or Cisco SSM satellite.

For more information on generating a registration token, see [Generating Registration Token, on page 5](#).

**Step 5** Click **REGISTER** to complete the registration process.  
Cisco Container Platform sends a request to Cisco SSM or Cisco SSM satellite to check the registration status and Cisco SSM or Cisco SSM satellite reports back the status to Cisco Container Platform, on a daily basis.

If registering the token fails, you can reregister the Cisco Container Platform instance using a new token.

For more information on reregistering Cisco Container Platform, see [Reregistering Cisco Container Platform License, on page 8](#).

---

## Renewing Authorization

By default, the authorization is automatically renewed every 30 days. However, Cisco Container Platform allows a user to manually initiate the authorization renew in case the automatic renewal process fails. The authorization expires if Cisco Container Platform is not connected to Cisco SSM or Cisco SSM satellite for 90 days and the licenses consumed by Cisco Container Platform are reclaimed and put back to the license pool.

### Before you begin

Ensure that the Cisco Container Platform instance is registered with Cisco SSM or Cisco SSM satellite.

---

**Step 1** Log in to the Cisco Container Platform web interface.

**Step 2** From the left pane, click **Licensing**.

**Step 3** From the **Actions** drop-down list, choose **Renew Authorization Now**.

**Step 4** Click **OK** in the **Renew Authorization** dialog box.

Cisco Container Platform synchronizes with Cisco SSM or Cisco SSM satellite to check the license authorization status and Cisco SSM or Cisco SSM satellite reports back the status to Cisco Container Platform, on a daily basis.

---

## Reregistering Cisco Container Platform License

You can reregister Cisco Container Platform with Cisco SSM or Cisco SSM satellite by deregistering it and registering it again, or by using a register force option.

### Before you begin

Ensure that you have obtained a new registration token from Cisco SSM or Cisco SSM satellite.

---

- Step 1** Log in to the Cisco Container Platform web interface.
  - Step 2** From the left pane, click **Licensing**.
  - Step 3** From the **Actions** drop-down list, choose **Reregister**.
  - Step 4** In the **Product Instance Registration Token** field of the **Smart Software Licensing Product Reregistration** dialog box, enter the registration token that you generated using Cisco SSM or Cisco SSM satellite.  
For more information on generating a registration token, see [Generating Registration Token, on page 5](#).
  - Step 5** Click **REGISTER** to complete the registration process.  
Cisco Container Platform sends a request to Cisco SSM or Cisco SSM satellite to check the registration status and Cisco SSM or Cisco SSM satellite reports back the status to Cisco Container Platform, on a daily basis.
- 

## Deregistering Registration

You can deregister the Cisco Container Platform instance from Cisco SSM or Cisco SSM satellite to release all the licenses from the current Virtual account and the licenses are available for use by other products in the virtual account. Deregistering disconnects Cisco Container Platform from Cisco SSM or Cisco SSM satellite.

### Before you begin

Ensure that the Cisco Container Platform instance is registered with Cisco SSM or Cisco SSM satellite.

---

- Step 1** Log in to the Cisco Container Platform web interface.
  - Step 2** From the left pane, click **Licensing**.
  - Step 3** From the **Actions** drop-down list, choose **Deregister**.
  - Step 4** Click **DEREGISTER** in the confirmation dialog box.  
Cisco Container Platform sends a request to Cisco SSM or Cisco SSM satellite to check the deregistration status and Cisco SSM or Cisco SSM satellite reports back the status to Cisco Container Platform, on a daily basis.
-



## CHAPTER 2

# Managing Cisco Container Platform Infrastructure Configuration

---

This chapter contains the following topics:

- [Managing Provider Profile, on page 9](#)
- [Managing ACI-CNI Profile, on page 10](#)
- [Managing VIP Pool, on page 11](#)

## Managing Provider Profile

Cisco Container Platform enables you to define the provider profile on which clusters can be created.

You can configure multiple provider profiles in an instance of Cisco Container Platform and use the same provider profile for multiple clusters.

## Adding Provider Profile

### Before you begin

Cisco Container Platform interacts with vSphere through the user that you configure when you add a provider profile. Hence, you need to ensure that this user has the necessary privileges.

For more information on the vSphere user privileges, see [User Privileges on vSphere, on page 33](#).

- 
- Step 1** From the left pane, click **Infrastructure Providers**.
- Step 2** Click **NEW PROVIDER** and specify information such as name and description of provider, IP address, port, username and password of the provider profile.
- Step 3** Click **SUBMIT**.
-

## Modifying Provider Profile

---

- Step 1** From the left pane, click **Infrastructure Providers**.
- Step 2** From the drop-down list displayed under the **ACTIONS** column, choose **Edit** corresponding to the provider profile that you want to modify.
- Step 3** Change the provider details as necessary and click **SUBMIT**.
- 

## Deleting Provider Profile

---

- Step 1** From the left pane, click **Infrastructure Providers**.
- Step 2** From the drop-down list displayed under the **ACTIONS** column, choose **Delete** corresponding to the provider profile that you want to delete.
- Step 3** Click **DELETE** in the confirmation dialog box.
- 

## Managing ACI-CNI Profile

Cisco Container Platform enables you to define ACI-CNI profiles using which tenant clusters can be created.

You can define multiple ACI-CNI profiles and use the same profile for multiple clusters.

## Adding ACI-CNI Profile

---

- Step 1** From the left pane, click **ACI-CNI Profiles**.
- Step 2** Click **NEW ACI-CNI PROFILE** and perform these steps:
- a) Specify information such as name, ACI host, username and password of the ACI instance.
 

**Note** If there is more than one host, use a comma-separated host list.
  - b) In the **ACI VMM DOMAINNAME** field, enter the Virtual Machine Manager Domain (VMMD) that you want to use.
  - c) In the **ACI INFRA VLAN ID** field, enter the VLAN number for layer 2 networking.
  - d) In the **VRF NAME** field, enter the Virtual Routing and Forwarding (VRF) IP address.
  - e) In the **L3 OUTSIDE POLICY NAME** field, enter the ACI object for allowing external internet connectivity.
  - f) In the **AAEP NAME** field, enter an Attachable Access Entity Profile (AAEP) name to associate the VMM domain to an AAEP.
  - g) In the **NAMESERVERS** field, enter the IP address of all the DNS servers that the ACI fabric can access.
  - h) In the **NODE VLAN START ID** field, enter the starting IP address for the IP pool that is used to allocate IP addresses to the node VLAN.
  - i) In the **NODE VLAN END ID** field, enter the ending IP address for the IP pool that is used to allocate IP addresses to the node VLAN.



- j) In the **STARTING SUBNET FOR PODS** field, enter the starting IP address for the IP pool that is used to allocate IP addresses to the pods.
- k) In the **STARTING SUBNET FOR SERVICE** field, enter the starting IP address for the IP pool that is used to allocate IP addresses to the service VLAN.
- l) In the **OPFLEX MULTICAST RANGE** field, enter a range for the Opflex multicast.
- m) In the **CONTROL PLANE CONTRACT NAME** field, enter the name of the contract that allows traffic from the Control Plane cluster to the tenant cluster.

**Step 3** Click **SUBMIT**.

---

## Modifying ACI-CNI Profile

---

**Step 1** From the left pane, click **ACI-CNI Configuration**.

**Step 2** From the drop-down list displayed under the **ACTIONS** column, choose **Edit** for the ACI-CNI profile that you want to modify.

**Step 3** Change the ACI-CNI profile details as necessary and click **SUBMIT**.

---

## Deleting ACI-CNI Profile

---

**Step 1** From the left pane, click **ACI-CNI Configuration**.

**Step 2** From the drop-down list displayed under the **ACTIONS** column, choose **Delete** for the ACI-CNI profile that you want to delete.

**Step 3** Click **DELETE** in the confirmation dialog box.

---

## Managing VIP Pool

Cisco Container Platform Virtual IP Address (VIP) pools are reserved ranges of IP addresses that are assigned as virtual IP addresses within the Cisco Container Platform clusters. For example, the master VIPs of tenant clusters or the external IP addresses of Ingress controllers are assigned from the VIP pool. The range of IP addresses in the VIP pools must be outside of the IP addresses that are assigned by DHCP.

## Adding VIP Pool

---

**Step 1** From the left pane, click **VIP Pools**.

**Step 2** Click **NEW VIP POOL** and perform these steps:

- a) In the **POOL NAME** field, enter a name for the VIP pool.
- b) In the **SUBNET VRID RANGES** field, enter a range of virtual router IDs, a list of comma-separated ranges, or a number. Virtual router IDs can be a value in the range 1 to 254.

For example: 1-10, 16-100, 15

- c) In the **SUBNET CIDR** field, specify a subnet from which virtual IP addresses can be allocated.  
For example: 10.0.0.0/24
- d) In the **IP ADDRESS RANGE** field, enter the start and end IP address for the range of addresses from which virtual IP addresses can be allocated. The IP address range must be within the subnet CIDR specified in Step (c).  
For example: Start: 10.0.0.50, End: 10.0.0.100

**Step 3** Click **SUBMIT**.

---

## Modifying VIP Pool

---

- Step 1** From the left pane, click **VIP Pools**.
  - Step 2** From the drop-down list displayed under the **ACTIONS** column, choose **Edit** for the VIP pool that you want to modify.
  - Step 3** Change the subnet CIDR and the IP address as necessary and click **SUBMIT**.
- 

## Deleting VIP Pool

---

- Step 1** From the left pane, click **VIP Pools**.
  - Step 2** From the drop-down list displayed under the **ACTIONS** column, choose **Delete** for the VIP pool that you want to delete.
  - Step 3** Click **DELETE** in the confirmation dialog box.
-



## CHAPTER 3

# Administering Kubernetes Clusters

---

You can create, modify, or delete Kubernetes clusters using the Cisco Container Platform web interface.

This chapter contains the following topics:

- [Creating Kubernetes Clusters, on page 13](#)
- [Upgrading Kubernetes Clusters, on page 14](#)
- [Scaling Kubernetes Clusters, on page 14](#)
- [Deleting Kubernetes Clusters, on page 15](#)
- [Managing Users and RBAC, on page 15](#)
- [Monitoring Health of Cluster Deployments, on page 16](#)
- [Monitoring Logs from Cluster Deployments, on page 17](#)

## Creating Kubernetes Clusters

---

**Step 1** From the left pane, click **Clusters**, and then click **NEW CLUSTER**.

**Step 2** In the **Basic Information** screen, specify the following information, and then click **NEXT**:

- The infrastructure provider where the cluster needs to be created.  
For more information, see [Adding Provider Profile, on page 9](#).
- The name, version of Kubernetes, SSH public key, and description to be used for creating the cluster.
- If you are using ACI CNI, specify the ACI CNI profile, see [Adding ACI-CNI Profile, on page 10](#).

**Step 3** In the **Provider Settings** screen, specify the data center, cluster, resource pool, network, HyperFlex local network, datastore, and VM template that you have configured on vSphere, and then click **NEXT**.

- Note**
- Ensure that DRA and HA are enabled on the cluster that you choose in this step. For more information on enabling DRS and HA on clusters, refer to the *Cisco Container Platform 1.0 Installation Guide*.
  - Ensure that the datastore that you choose in this step is accessible to the hosts in the cluster.

**Step 4** In the **Node Configuration** screen, specify the following information, and then click **NEXT**:

- The number of worker and master nodes, and their VCPU and memory configurations.

- The VM username that you want to use as the login for the VM.
- The VIP pool that you want to use for this cluster.
- The IP addresses in CIDR notation that you want to use as the pod subnet.

**Step 5** In the **Summary** screen, verify the configuration, and then click **FINISH**.

The cluster deployment takes few minutes to complete. The newly created cluster is displayed on the **Clusters** page.

For more information on deploying applications on clusters, see [Deploying Applications on Kubernetes Clusters, on page 27](#).

## Upgrading Kubernetes Clusters

### Before you begin

Ensure that you have imported the latest tenant cluster OVA to the vSphere environment.

For more information on importing the tenant cluster OVA, refer to the *Installation Guide*.

**Step 1** From the left pane, click **Clusters**.

**Note** When an upgrade is available for a Kubernetes cluster, a **Refresh** icon is displayed under the **Kubernetes Version** column.

**Step 2** Click the **Refresh** icon corresponding to the cluster that you want to upgrade.

**Step 3** In the **Kubernetes Update** dialog box, choose a template for the VM, and then click **Update Now**. It may take a few minutes for the Kubernetes cluster upgrade to complete.

## Scaling Kubernetes Clusters

You can scale clusters by adding or removing nodes to them based on the demands of the workloads you want to run.

**Step 1** From the left pane, click **Clusters**.

**Step 2** From the drop-down list displayed under the **ACTIONS** column, choose **Edit** and modify the number of worker nodes, and then click **UPDATE**.

Alternatively, follow these steps to scale the cluster:

- Click the name of the cluster that you want to scale.
- Click the **Nodes** tab.

- c) From the right pane, click **EDIT**, modify the number of worker nodes, and then click **UPDATE**.
- 

## Deleting Kubernetes Clusters

### Before you begin

Ensure that the cluster you want to delete is not currently in use, as deleting a cluster removes the containers and data associated with it.

---

- Step 1** From the left pane, click **Clusters**.
- Step 2** From the drop-down list displayed under the **ACTIONS** column, choose **Delete** for the cluster that you want to delete.
- Step 3** Click **DELETE** in the confirmation dialog box.
- 

## Managing Users and RBAC

Cisco Container Platform provides Role-based Access Control (RBAC) through built-in static roles, namely the *Administrator* and *User* roles. Role-based access allows you to use local accounts and LDAP for authentication and authorization.

### Configuring Local Users

Cisco Container Platform allows you to manage local users. An administrator can add a user, and assign an appropriate role and cluster(s) to the user.

---

- Step 1** From the left pane, click **User Management**, and then click the **Users** tab.
- Step 2** Click **NEW USER**.
- Step 3** Specify information such as first name, last name, username, password, and role for the user.
- Step 4** Click **SUBMIT**.  
The new user is displayed on the **User Management** page.
- Note** You can edit or delete a user by using the options available under the **ACTIONS** column.
- 

### Changing Login Password

- Step 1** From the left pane, click **User Management**, and then click the **Users** tab.
- Step 2** From the drop-down list displayed under the **ACTIONS** column, choose **Edit** corresponding to your name.
- Note** Administrators can change password and role for other users as well.

**Step 3** Change the password and role assigned as necessary, and click **SUBMIT**.

---

## Configuring AD Servers

LDAP authentication is performed using a service account that can access the LDAP database and query for user accounts. You will need to configure the AD server and service account in Cisco Container Platform.

---

- Step 1** From the left pane, click **User Management**, click the **Active Directory** tab, and then click **EDIT**.
  - Step 2** In the **SERVER IP ADDRESS** field, type the IP address of the AD server.
  - Step 3** In the **PORT** field, type the port number for the AD server.
  - Step 4** For improved security, we recommend that you check **STARTTLS**.
  - Step 5** In the **BASE DN** field, specify the domain name of the AD server for all the accounts that you have.
  - Step 6** In the **ACCOUNT USERNAME** field, specify the service account name that is used for accessing the LDAP server.
  - Step 7** In the **PASSWORD** field, type the password of the AD account.
  - Step 8** Click **SUBMIT**.
- 

## Configuring AD Groups

Cisco Container Platform allows you to manage users using AD groups. An administrator can add users to AD groups, and then assign appropriate roles and clusters to the groups.

### Before you begin

Ensure that you have configured the AD server that you want to use.

For more information on configuring AD servers, see [Configuring AD Servers, on page 16](#).

---

- Step 1** From the left pane, click **User Management**, and then click the **Groups** tab.
  - Step 2** Click **ADD GROUP**.
  - Step 3** Specify information such as the name of the AD group and the role you want to assign to the group.
    - Note** If the AD group is associated with the *Administrator* role, by default, access is provided to all clusters. But, if the AD group is associated with the *User* role, you need to assign a cluster.
  - Step 4** From the **CLUSTERS** drop-down list, choose the names of the cluster that you want to assign to the AD group.
  - Step 5** Click **SUBMIT**.
- 

## Monitoring Health of Cluster Deployments

It is recommended to continuously monitor the health of your cluster deployment to improve the probability of early detection of failures and avoid any significant impact from a cluster failure.

Cisco Container Platform is deployed with Prometheus and Grafana configured to start monitoring and logging services automatically when a Kubernetes cluster is created.

[Prometheus](#) is an open-source systems monitoring and alerting toolkit and [Grafana](#) is an open source metric analytics and visualization suite.

Prometheus collects the data from the cluster deployment, and Grafana provides a general purpose dashboard for displaying the collected data. Grafana offers a highly customizable and user-friendly dashboard for monitoring purposes.




---

**Note** A user with *Administrator* role can view all the cluster deployments, but a user with *User* role can view only those clusters for which the user has permission to view.

---

**Step 1** Access the Kubernetes cluster master node using ssh.

```
ssh -l <username> <IP address of master node>
```

**Note** Once you create a Kubernetes cluster, it may take a few minutes for the necessary services to start. If ssh to a cluster fails, we recommend that you try again after a few minutes.

**Step 2** Obtain the password for Grafana, which is stored as a Kubernetes secret.

```
kubectl get secrets ccp-addons-grafana -o yaml | grep grafana-admin-password | awk '{print $2}' | base64 --decode
```

**Step 3** Access the Grafana UI using a web browser.

```
<IP address of worker node>:<Node port>/login
```

**Step 4** Log in to the Grafana UI of your Kubernetes cluster using your username, and the password that you obtained in Step 2.

**Note** It is important to either change or retain the original login credentials since the secret that was used to initialize the Grafana login may be lost or changed with future upgrades.

**Step 5** Add Prometheus as the data source and configure the Grafana dashboard to monitor the health of your cluster deployments.

---

## Monitoring Logs from Cluster Deployments

The Elasticsearch, Fluentd, and Kibana (EFK) stack enables you to collect and monitor log data from containerized applications for troubleshooting or compliance purposes. These components are automatically installed when you install Cisco Container Platform.

Fluentd is an open source data collector. It works at the backend to collect and forward the log data to Elasticsearch.

Kibana is an open source analytics and visualization platform designed to work with Elasticsearch. It allows you to create rich visualizations and dashboards with the aggregated data.




---

**Note** A user with the *Administrator* role can view all logs, but a user with *User* role can view logs for only those clusters for which the user has permission to view.

---

## Viewing EFK Logs Using Kibana (Tenant Cluster)

---

- Step 1** Download the Kubeconfig file of the cluster whose logs you want to view, see [Downloading Kubeconfig File, on page 27](#).
- Step 2** Copy the contents of the downloaded Kubeconfig file to:
- Your local host `~/ .kube/config`
  - A local file and export `KUBECONFIG=<Downloaded Kubeconfig file>`
- Step 3** Create a port-forward using `kubectl` to access Kibana from outside a cluster.
- a) Determine the pod.
 

```
kubectl -n kube-system get pods
```

For example, `kibana-logging-7db596d7f6-g9pxv`
  - b) Open a port-forward.
 

```
kubectl port-forward -n kube-system
```

For example, `kibana-logging-7db596d7f6-g9pxv 5601:5601`
- Step 4** Access the Kibana UI and view the data from the target tenant cluster using a web browser.  
`http://localhost:5601/app/kibana`  
 For more information on customizing the Kibana UI, refer to the [latest Kibana documentation](#).
- 

## Viewing EFK Logs Using Kibana (Control Plane Cluster)

---

- Step 1** Access the Kubernetes cluster master node using `ssh`.
- ```
ssh ccpuser@control plane master node
sudo cat /etc/kubernetes/admin.conf
```
- Step 2** Copy the contents of the downloaded Kubeconfig file to:
- Your local host `~/ .kube/config`
  - A local file and export `KUBECONFIG=<Full path of the Kubeconfig local file>`
- For more information on setting Kubeconfig, see [Configure Access to Multiple Clusters](#).
- Step 3** Create a port-forward using `kubectl` to access Kibana from outside a cluster.
- a) Determine the pod.



```
kubectl -n kube-system get pods
```

For example, kibana-logging-7db596d7f6-g9pxv

- b) Open a port-forward.

```
kubectl port-forward -n kube-system
```

For example, kibana-logging-7db596d7f6-g9pxv 5601:5601

- Step 4** Access the Kibana UI and view the data from the target tenant cluster using a web browser.

<http://localhost:5601/app/kibana>

For more information on customizing the Kibana UI, refer to the [latest Kibana documentation](#).

- Step 5**

### What to do next

## Forwarding Logs to External Elasticsearch Server

Use the following Curl commands to configure forwarding of logs to an external Elasticsearch server:

- Step 1** Open a terminal that has a curl client installed.

- Step 2** Configure Cisco Container Platform login credentials.

```
export MGMT_HOST=https://<Cisco Container Platform IP address>:<Port>
export CCP_USER=<Username>
export CCP_PASSWORD=<Password>
```

- Step 3** Login to Cisco Container Platform and save the session cookie for future requests into the cookies.txt local file.

```
curl -k -j -c cookies.txt -X POST -H "Content-Type:application/x-www-form-urlencoded" -d
"username=$CCP_USER&password=$CCP_PASSWORD" $MGMT_HOST/2/system/login/
```

- Step 4** Get the list of cluster names.

```
curl -s -k -b cookies.txt -H "Content-Type: application/json"
$MGMT_HOST/2/clusters/ | jq -r '.[].name'
```

- Step 5** Set the CLUSTER\_NAME environment variable to the cluster that you are working on.

```
export CLUSTER_NAME="<A cluster name from Step 2>"
```

- Step 6** Configure the cluster UUID.

```
export CLUSTER_UUID=$(curl -s -k -b cookies.txt -H "Content-Type: application/json"
$MGMT_HOST/2/clusters/$CLUSTER_NAME | jq -r '.uuid')
```

- Step 7** Configure the Elasticsearch server IP address and port number.

```
export EFK_SERVER=<IP address of Elasticsearch server>
export EFK_PORT=<Port number of Elasticsearch server>
```

- Step 8** Install the helm chart to configure the custom Elasticsearch server.

```
curl -s -k -b cookies.txt -X POST --header 'Content-Type: application/json' --header 'Accept:
application/json' -d '{"chart_url": "/opt/ccp/charts/ccp-agent.tgz", "name": "ccpagent", "options":
```

```
"cp-efk.localLogForwarding.enabled=False,cp-efk.localLogForwarding.elasticsearchHost='EFK_SERVER',cp-efk.localLogForwarding.elasticsearchPort='EFK_PORT')'
$MGMT_HOST/2/clusters/$CLUSTER_UUID/helmcharts
```

---



## CHAPTER 4

# Managing Kubernetes Clusters

---

The Cisco Container Platform web interface allows you to manage Kubernetes clusters by using the **Kubernetes Dashboard**. Once you set up the **Kubernetes Dashboard**, you can deploy applications on the authorized Kubernetes clusters, and manage the application and the cluster itself.

This chapter contains the following topic:

- [Setting up Kubernetes Dashboard, on page 21](#)

## Setting up Kubernetes Dashboard

---

- Step 1** From the left pane, click **Clusters**.
- Step 2** From the drop-down list displayed under the **ACTIONS** column, choose **Kubernetes Dashboard** for the cluster that you want to access using the Kubernetes Dashboard.
- Step 3** Click the **Download** icon of the cluster environment to get the Kubernetes configuration (Kubeconfig) file.
- Step 4** Use the Kubeconfig file from Step 3 to login to the Kubernetes Dashboard.
-





## CHAPTER 5

# Services and Networking

---

This chapter contains the following topics:

- [Load Balancing Kubernetes Services using NGINX, on page 23](#)
- [Network Policies, on page 26](#)

## Load Balancing Kubernetes Services using NGINX

Cisco Container Platform uses NGINX to offer advanced layer 7 load balancing solutions. NGINX can handle a large number of requests and at the same time, it can be run on Kubernetes containers.

The NGINX load balancer is automatically provisioned as part of Kubernetes cluster creation. Each Kubernetes cluster is provisioned with a single L7 NGINX load balancer. You can access the load balancer using its virtual IP address, which can be found by running the command `kubectl get svc`.

To use the NGINX load balancer, you must create an Ingress resource. Ingress is a Kubernetes object that allows you to define HTTP load balancing rules to allow inbound connections to reach the cluster services. You can configure Ingress to create external URLs for services, load balance traffic, terminate SSL, offer name-based virtual hosting, and so on.

## Types of Ingress

Cisco Container Platform supports the following types of Ingresses:

- **Simple fanout**

It enables you to access the website using http.

For example:

```
cafe.test.com -> 10.1.1.1 -> /tea    tea-svc:80
                               /coffee coffee-svc:80
```

For this type of Ingress, you need to create a yaml file that defines the Ingress rules.

**Figure 1: Sample yaml file**

```

apiVersion: extensions/v1beta1
kind: Ingress
metadata:
  name: cafe-ingress
spec:
  rules:
  - host: cafe.test.com
    http:
      paths:
      - path: /tea
        backend:
          serviceName: tea-svc
          servicePort: 80
      - path: /coffee
        backend:
          serviceName: coffee-svc
          servicePort: 80

```

- **Simple fanout with SSL termination**

It enables you to access the website using https.

For example:

```

https://cafe.test.com -> 10.1.1.1 -> /tea    tea-svc:80
                             /coffee  coffee-svc:80

```

For this type of Ingress, you need to create the following yaml files:

- A yaml file that defines the Secret

**Figure 2: Sample yaml file**

```

apiVersion: v1
kind: Secret
metadata:
  name: cafe-secret
type: Opaque
data:
  tls.crt: base64 encoded cert
  tls.key: base64 encoded key

```

- A yaml file that defines the Ingress rules

Figure 3: Sample yaml file

```

apiVersion: extensions/v1beta1
kind: Ingress
metadata:
  name: cafe-ingress
spec:
  tls:
  - hosts:
    - cafe.test.com
    secretName: cafe-secret
  rules:
  - host: cafe.example.com
    http:
      paths:
      - path: /tea
        backend:
          serviceName: tea-svc
          servicePort: 80
      - path: /coffee
        backend:
          serviceName: coffee-svc
          servicePort: 80

```

- **Name based virtual hosting**

It enables you to access the website using multiple host names.

For example:

```

tea.test.com  --|          |-> tea.test.com  s1:80
               | 10.1.1.1 |
coffee.test.com --|      |-> coffee.test.com s2:80

```

For this type of Ingress, you need to create a yaml file that defines the Ingress rules.

Figure 4: Sample yaml file

```

apiVersion: extensions/v1beta1
kind: Ingress
metadata:
  name: cafe-ingress
spec:
  rules:
  - host: tea.test.com
    http:
      paths:
      - path: /tea
        backend:
          serviceName: tea-svc
          servicePort: 80
  - host: coffee.test.com
    http:
      paths:
      - path: /coffee
        backend:
          serviceName: coffee-svc
          servicePort: 80

```



---

**Note** You can download the yaml files that are shown in this topic from the following link:

<https://github.com/nginxinc/kubernetes-ingress/tree/master/examples/complete-example>

---

For more information on a sample scenario of implementing Ingress, see [Deploying Cafe Application with Ingress, on page 29](#).

## Network Policies

Cisco Container Platform supports [Kubernetes NetworkPolicies](#). The NetworkPolicies are independent of the underlying container network plugin—ACI CNI, Contiv VPP, or Calico.





## CHAPTER 6

# Deploying Applications on Kubernetes Clusters

Once you have created Kubernetes cluster using the Cisco Container Platform web interface, you can deploy containerized applications on top of it.

This chapter contains the following topics:

- [Workflow of Deploying Applications, on page 27](#)
- [Downloading Kubeconfig File, on page 27](#)
- [Sample Scenarios, on page 28](#)

## Workflow of Deploying Applications

| Task                                                                                                                 | Related Section                                          |
|----------------------------------------------------------------------------------------------------------------------|----------------------------------------------------------|
| Create Kubernetes clusters using the Cisco Container Platform web interface.                                         | <a href="#">Creating Kubernetes Clusters, on page 13</a> |
| Download the kubeconfig file that contains the cluster information and the certificates required to access clusters. | <a href="#">Downloading Kubeconfig File, on page 27</a>  |
| Use the kubectl utility to deploy the application and test the scenario.                                             | <a href="#">Sample Scenarios, on page 28</a>             |

## Downloading Kubeconfig File

You must download the cluster environment to access the Kubernetes clusters using command line tools such as `kubectl` or using APIs.

---

**Step 1** From the left pane, click **Clusters**.

**Step 2** Click the **Download** icon corresponding to the cluster environment that you want to download.

The `kubeconfig` file that contains the cluster information and the certificates required to access clusters is downloaded to your local system.

---

# Sample Scenarios

This topic contains a few sample scenarios of deploying applications.

## Deploying a Pod with Persistent Volume

This scenario describes deploying and configuring a pod with persistent volume.

**Step 1** Go to the following URL:

<https://github.com/kubernetes/examples/tree/master/staging/volumes/vsphere>

**Step 2** Download the following yaml files:

- vsphere-volume-sc-fast.yaml
- vsphere-volume-pvcsc.yaml
- vsphere-volume-pvcscpod.yaml

**Step 3** Open the **kubect** utility.

**Step 4** Configure the Kubernetes cluster.

```
export KUBECONFIG=<Path to kubeconfig file>
```

**Step 5** Create the storage class.

```
$ kubectl create -f vsphere-volume-sc-fast.yaml
```

**Step 6** Verify if the storage cluster is created.

```
$ kubectl describe storageclass fast

Name: fast
IsDefaultClass: No
Annotations: <none>
Provisioner: kubernetes.io/vsphere-volume
Parameters: diskformat=zeroedthick,fstype=ext3
No events.```
```

**Step 7** Create the persistent volume claim to request for storage.

```
$ kubectl create -f vsphere-volume-pvcsc.yaml
```

**Step 8** Verify if the persistent volume claim (pvc) is created.

```
$ kubectl describe pvc pvcsc001

Name: pvcsc001
Namespace: default
StorageClass: fast
Status: Bound
Volume: pvc-83295256-f8e0-11e6-8263-005056b2349c
Labels: <none>
Capacity: 2Gi
Access Modes: RWO
Events:````
FirstSeen LastSeen Count From              SubObjectPath Type Reason      Message
1m         1m         1    persistentvolume Normal Provisioning Successfully provisioned
```

```
-controller                               Succeeded   volume pvc-83295256-f8e0
   -11e6-8263-005056b2349c
   using
```

```
kubernetes.io/vsphere-volume
```

Persistent Volume is automatically created and is bounded to this pvc.

**Step 9** Verify if the persistent volume claim is created:

```
$ kubectl describe pv pvc-83295256-f8e0-11e6-8263-005056b2349c

Name: pvc-83295256-f8e0-11e6-8263-005056b2349c
Labels: <none>
StorageClass: fast
Status: Bound
Claim: default/pvcsc001
Reclaim Policy: Delete
Access Modes: RWO
Capacity: 2Gi
Message:
Source:
Type: vSphereVolume (a Persistent Disk resource in vSphere)
VolumePath: [datastore1] kubevols/kubernetes-dynamic-pvc-83295256-f8e0-11e6-8263-005056b2349c.vmdk
FSType: ext3
No events.
```

**Note** VMDK is created inside the *kubevols* folder in the datastore, which is specified in the `vsphere cloudprovider config` file that is created during the setup of Kubernetes cluster on vSphere.

**Step 10** Create a pod that uses persistent volume claim with storage class.

```
$ kubectl create -f vsphere-volume-pvcscpod.yaml
```

**Step 11** Verify if the pod is up and running.

```
$ kubectl get pod pvpod

NAME      READY   STATUS    RESTARTS   AGE
pvpod    1/1     Running   0           48m
```

**Step 12** While the pod is starting, access vCenter and view the dynamically provisioned VMDKs of the pod.

## Deploying Cafe Application with Ingress

This scenario describes deploying and configuring the *Cafe application* with Ingress rules to manage incoming HTTP requests. It uses a **Simple fanout with SSL termination Ingress**.

For more information on Ingress, see [Load Balancing Kubernetes Services using NGINX, on page 23](#).

**Step 1** Go to the following URL:

<https://github.com/nginxinc/kubernetes-ingress/tree/master/examples/complete-example>

**Step 2** Download the following yaml files:

- `tea-rc.yaml`
- `tea-svc.yaml`

- coffee-rc.yaml
- coffee-svc.yaml
- cafe-secret.yaml
- cafe-ingress.yaml

**Step 3** Open the **kubectl** utility.

**Step 4** Obtain the IP address of the L7 NGINX load balancer that Cisco Container Platform automatically installs:

```
$ kubectl get pods --all-namespaces -l app=ingress-nginx -o wide
NAMESPACE      NAME                                READY  STATUS  RESTARTS  AGE  IP              NODE
ingressnginx   nginx-                              1/1    Running  0          3d   10.10.45.235   test-clusterwc5729f9ce2
                ingresscontroller
                -66974b775-jnmpl
```

**Step 5** Deploy the Cafe application.

a) Create the coffee and the tea services and replication controllers:

```
kubectl create -f tea-rc.yaml<br>
kubectl create -f tea-svc.yaml<br>
kubectl create -f coffee-rc.yaml<br>
kubectl create -f coffee-svc.yaml
```

**Step 6** Configure load balancing.

a) Create a Secret with an SSL certificate and a key:

```
kubectl create -f cafe-secret.yaml
```

b) Create an Ingress Resource:

```
kubectl create -f cafe-ingress.yaml
```

**Step 7** Verify that the Cafe application is deployed.

```
$ kubectl get pods -o wide
NAMESPACE      READY  STATUS    RESTARTS  AGE  IP              NODE
coffee-rc-jb9sx 1/1    Running   0          3d   192.168.151.134 test-cluster-wb3d42afeff
coffee-rc-tjwgj 1/1    Running   0          3d   192.168.44.133  test-cluster-wc5729f9ce2
tea-rc-6qmvm    1/1    Running   0          3d   192.168.44.132  test-cluster-wc5729f9ce2
tea-rc-ms46j    1/1    Running   0          3d   192.168.151.132 test-cluster-wb3d42afeff
tea-rc-tnftv    1/1    Running   0          3d   192.168.151.133 test-cluster-wb3d42afeff
```

**Step 8** Verify if the coffee and tea services are deployed.

```
$ kubectl get svc
NAME          TYPE        CLUSTER-IP    EXTERNAL-IP    PORT(S)    AGE
coffee-svc   ClusterIP   10.105.139.1  80/TCP         3d
kubernetes    ClusterIP   10.96.0.1     443/TCP        4d
tea-svc       ClusterIP   10.109.34.129 80/TCP         3d
```

**Step 9** Verify if the Ingress is deployed.

```
$ kubectl describe ing
Name: cafe-ingress
Namespace: default
Address:
Default backend: default-http-backend:80 (<none>)
```

```
TLS: cafe-secret terminates cafe.example.com
Rules:

Host          Path      Backends
cafe.example.com
              /tea      tea-svc:80 (<none>)
              /coffee   coffee-svc:80 (<none>)

Annotations:
Events: <none>
```

**Step 10**

Test the application.

- a) Access the load balancer IP address 10.10.45.235, which is obtained in Step 2.
- b) Test if the Ingress controller is load balancing as expected.

```
$ curl --resolve cafe.example.com:443:10.10.45.235 https://cafe.example.com/coffee --insecure
<!DOCTYPE html>
...
<p><span>Server&nbsp;address:</span> <span>192.168.151.134:80</span></p>
...
$ curl --resolve cafe.example.com:443:10.10.45.235 https://cafe.example.com/coffee --insecure
<!DOCTYPE html>
...
<p><span>Server&nbsp;address:</span> <span>192.168.44.133:80</span></p>
...
```

---





# APPENDIX **A**

## User Privileges on vSphere

This appendix contains the following topic:

- [User Privileges on vSphere, on page 33](#)

### User Privileges on vSphere

The following table provides the minimal set of privileges that are required by the vSphere user to execute the relevant operations in vCenter.

Roles	Privileges	Entities	Propagate to Children
manage-k8s-node-vm	Resource.AssignVMToPool System.Anonymous System.Read System.View VirtualMachine.Config.AddExistingDisk VirtualMachine.Config.AddNewDisk VirtualMachine.Config.AddRemoveDevice VirtualMachine.Config.RemoveDisk VirtualMachine.Inventory.Create VirtualMachine.Inventory.Delete	Cluster, Hosts, VM Folder	Yes
manage-k8s-volumes	Datastore.AllocateSpace Datastore.FileManagement System.Anonymous System.Read System.View	Datastore	No

<b>Roles</b>	<b>Privileges</b>	<b>Entities</b>	<b>Propagate to Children</b>
k8s-system-read-and-spbmprofile-view	StorageProfile.View System.Anonymous System.Read System.View	vCenter	No
ReadOnly	System.Anonymous System.Read System.View	Datacenter, Datastore Cluster, Datastore Storage Folder	Yes
ccp-register-extension	Extension.Register Extension.Unregister Extension.Update	vCenter	No

For more information on adding a provider profile, see [Adding Provider Profile, on page 9](#).