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

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.

- The name, version of Kubernetes, and description to be used for creating the cluster.
- If you are using ACI, specify the ACI profile, see Adding ACI Profile.

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 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 SSH public key that you want to use for creating the cluster.
- The VM username that you want to use as the login for the VM.
- The subnet that you want to use for this cluster.
- The number of load balancer IP addresses for this cluster.

  For more information, see Load Balancer Services.
- The IP addresses in CIDR notation that you want to use as the pod subnet.
- Whether or not you want to enable Istio
- A root CA certificate to allow tenant clusters to securely connect to additional services

Step 5 In the **Harbor Registry** screen, specify if you want to enable Harbor. If no, click **NEXT**. If yes, you must specify the following information, and then click **NEXT**:

a) Ensure the switch to enable Harbor is activated
b) A password for Harbor server admin
c) The immutable registry size in gigabits

Step 6 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.

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 *Cisco Container Platform Installation Guide*.

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

Step 2 From the drop-down list displayed under the **ACTIONS** column, choose **Upgrade**.

Step 3 In the **Upgrade Cluster** dialog box, enter a Kubernetes version, choose a new template for the VM, and then click **Submit**. It may take a few minutes for the Kuberenetes 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**, modify the number of worker nodes or load balancer IP addresses, and then click **UPDATE**.

Alternatively, follow these steps to scale the cluster:

a) Click the name of the cluster that you want to scale.
b) Click the **Nodes** tab.
c) From the right pane, click **EDIT**, modify the number of worker nodes or load balancer IP addresses, 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, passphrase, 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 Passphrase

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 passphrase and role for other users as well.

Step 3 Change the passphrase 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 **PASSPHRASE** field, type the passphrase 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 4.

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

Note When you run the command on a Control Plane cluster, you can skip **-n ccp** in the command as these services run in the default namespace.

Step 3 Access the Grafana UI using a web browser.

```
https://<VIP>/grafana
```

Where <*VIP*> is the Virtual IP address of the control or tenant cluster as the case may be. In case of a tenant cluster, <*VIP*> is the Virtual IP address that is used by the cluster Ingress as described in Services and Networking.

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.

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 ccp get pods
```

For example, `kibana-logging-7db596d7f6-g9pxv`

b) Open a port-forward.

```
kubectl port-forward -n ccp
```

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 get pods
```

For example, `kibana-logging-7db596d7f6-g9pxv`

b) Open a port-forward.

```
kubectl port-forward
```

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_PASSPHRASE=<Passphrase>
```

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":

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