



Upgrading the Cisco ACI CNI Plug-in

[New and Changed Information](#) 2

New and Changed Information

The following table provides an overview of the significant changes up to this current release. The table does not provide an exhaustive list of all changes or of the new features up to this release.

Cisco APIC Release Version	Feature	Description
5.1(x)	--	Adds instructions for using acc-provision, depending on the release version, in two sections: <ul style="list-style-type: none">• Generating an Updated Cisco ACI CNI Configuration, on page 3• Nondisruptively Upgrading the Cisco ACI CNI Plug-in, on page 4
3.0 and later.	--	This document was created.

Cisco ACI CNI Plug-in Upgrade

This document describes how to upgrade the Cisco Application Centric Infrastructure (ACI) Container Network Interface (CNI) Plug-in, including guidelines and disruptive and nondisruptive upgrade methods.

Guidelines for Upgrading the Cisco ACI CNI Plug-in

You need to have the following in order to use or upgrade the Cisco Application Centric Infrastructure (ACI) Plug-in:

- Cisco Application Policy Infrastructure Controller
- Switches from the EX family or later

See the [Cisco ACI and Cisco APIC documentation](#) on Cisco.com for details.

You may not need to upgrade both Cisco Application Centric Infrastructure (ACI) and Container Network Interface (CNI) Plug-in at the same time. A specific version of the Cisco ACI CNI Plug-in can be compatible with multiple instances of Cisco Application Policy Infrastructure Controller (APIC) and multiple versions of Cisco ACI leaf firmware.

For information about compatibility, see the [Cisco ACI Virtualization Compatibility Matrix](#) on Cisco.com.

Upgrading the Cisco ACI Plug-in

All the components of the Cisco Application Centric Infrastructure (ACI) Container Network Interface (CNI) Plug-in are containerized. Upgrading the Cisco ACI CNI component follows the same model of a normal Kubernetes or OpenShift application update.

The upgrade process upgrades both the control plane and the data plane of the Cisco ACI CNI Plug-in.

The control plane consists of the following pods:

- `aci-containers-controller`
- `aci-containers-host`
- `aci-containers-operator`

The data plane consists of the container `aci-containers-openvswitch`.

The following section includes two upgrade procedures:

- Nondisruptive upgrade: This workflow takes extra steps to minimize traffic disruption.
- Disruptive upgrade: This workflow is ideal for environments where traffic disruption is not a concern, such as labs, and reduces the required upgrade time.

Generating an Updated Cisco ACI CNI Configuration

Before you can upgrade the Cisco Application Centric Infrastructure (ACI) Container Network Interface (CNI) plug-in, you must generate an updated Cisco ACI CNI configuration.

Procedure

- Step 1** Download the new version of the Cisco ACI CNI plug-in that you need for your environment and install it on your master nodes by completing the following steps:
- a) Go to the [software download link](#).
 - b) Click **APIC OpenStack and Container Plugins**.
 - c) Download the version of the CNI plug-in that you want to upgrade to.

- Step 2** Copy the private key and certificate in the same folder of your ACI CNI JSON configuration file.
- Communications between Cisco Application Policy Infrastructure Controller (APIC) and `aci-containers-controller` uses certificate-based authentication. The certificates are generated by `acc-provision` when the cluster is set up the first time and are saved locally.

- Step 3** Ensure that in your current working directory you have the key and certificate files that were generated by `acc-provision` previously.
- If you don't have these files, you can use a new executable that is called `acc-retrieve-cert` to retrieve them. You can run the executable by using one of the two variants of the **`acc-retrieve-cert`** command:

```
acc-retrieve-cert  
or  
acc-retrieve-cert --kubeconfig path-to-kube-config
```

- Step 4** If the current cluster was provisioned with a Cisco ACI CNI version earlier than 5.0, you must add the following configuration in the `aci_cni_original_config.yaml` file:

```
aci_config:  
  use_legacy_kube_naming_convention: True
```

- Step 5** Generate the updated Cisco ACI CNI container configuration based on the original Cisco ACI CNI configuration file by executing the following commands:

When using an `acc-provision` release 5.1 or later:

```
acc-provision -f flavor -c aci_cni_original_config.yaml --upgrade -o aci_cni_new_config.yaml
```

Example:

```
acc-provision -f kubernetes-1.18 -c aci_cni_prod.yaml --upgrade -o aci_cni_prod_5_1_1_0.yaml
```

When using `acc-provision` release earlier than 5.1:

```
acc-provision -f flavor -c aci_cni_original_config.yaml -a -o aci_cni_new_config.yaml
```

Example:

```
acc-provision -f kubernetes-1.17 -c aci_cni_prod_4_2_2_2.yaml -a -o aci_cni_prod_5_0_1_0.yaml
```

What to do next

Upgrade the Cisco ACI CNI plug-in, using either the disruptive or nondisruptive method.

Nondisruptively Upgrading the Cisco ACI CNI Plug-in

The steps in this procedure upgrade all Cisco Application Centric Infrastructure (ACI) Container Network Interface (CNI) Plug-in plug in containers nondisruptively.

Before you begin

You must have followed the guidelines, fulfilled the prerequisites, and upgrading the Cisco ACI CNI configuration.

Procedure

Step 1	Complete one of the following actions:	
	Option	Description
	If your version of acc-provision is	Then
	5.1 or later	Skip step 1 and go to step 2.
	Earlier than 5.1	<p>Edit the <code>aci_cni_new_config.yaml</code> file by modifying the <code>updateStrategy</code> from <code>RollingUpdate</code> to <code>OnDelete</code>, as shown in the following example:</p> <pre># sed -i -e 's/RollingUpdate/OnDelete/g' aci_cni_new_config.yaml</pre> <p>Note When acc-provision 5.1 or later is invoked with the <code>--upgrade</code> option, the <code>updateStrategy</code> is updated to <code>OnDelete</code>, so this step does not need to be performed when using earlier versions.</p>

Step 2 Apply the new Cisco ACI CNI configuration.

This step immediately upgrades the `aci-containers-controller` and the `aci-containers-operators` containers. However no upgrade should take place for the `aci-containers-host` and `aci-containers-openvswitch` containers.

Example:

Kubernetes syntax:

```
# kubectl apply -f aci_cni_new_config.yaml
```

Example:

OpenShift syntax:

```
# oc apply -f aci_cni_new_config.yaml
```

Step 3 Disable scheduling and drain the first node to undergo the upgrade.

Caution The following commands to drain a node are only an example. Before executing them, verify if they are correct and safe for your cluster. If the commands are executed incorrectly, an outage can result.

Example:

Kubernetes syntax:

```
# kubectl drain node --ignore-daemonsets --delete-local-data
```

Example:

OpenShift syntax:

```
# oc adm manage-node node --schedulable=false
# oc adm drain node --force --delete-local-data --ignore-daemonsets
```

Step 4 Get the list of pods that are currently running on the node that has been evacuated or drained.

Only DaemonSet should be present.

Example:

Kubernetes syntax for Cisco ACI versions 5.0 and later:

```
kubectl -n aci-containers-system get pod -o wide | grep node
```

Example:

Kubernetes syntax for Cisco ACI versions earlier than 5.0:

```
kubectl -n kube-system get pod -o wide | grep node node
```

Example:

OpenShift syntax:

```
# oc -n aci-containers-system get pod -o wide | grep node
```

Step 5 Restart the Cisco ACI CNI DaemonSets.

Example:

Kubernetes syntax Cisco ACI versions 5.0 and later:

```
# kubectl -n aci-containers-system delete pod \
  aci-containers-host-ID aci-containers-openvswitch-ID
```

Example:

Kubernetes syntax Cisco ACI versions earlier than 5.0:

```
# kubectl -n kube-system delete pod \
  aci-containers-host-ID aci-containers-openvswitch-ID
```

Example:

OpenShift syntax:

```
# oc -n aci-containers-system delete pod \
  aci-containers-host-ID aci-containers-openvswitch-ID
```

Step 6 Wait until `aci-containers-openvswitch` and `aci-containers-host` containers are restarted on the drained node.

Step 7 Uncordon the node.

Example:

Kubernetes syntax:

```
kubectl uncordon node
```

Example:

OpenShift syntax:

```
oc adm manage-node node --schedulable
```

Step 8 Repeat Step 3 through Step 7 for all the remaining nodes.

Disruptively Upgrading the Cisco ACI CNI Plug-in

The steps in this procedure upgrade all Cisco Application Centric Infrastructure (ACI) Container Network Interface (CNI) Plug-in containers in a rolling upgrade. This method upgrades the Cisco ACI CNI Plug-in more quickly than the nondisruptive method—but *is disruptive*.

Before you begin

You must have followed the guidelines, fulfilled the prerequisites, and upgrading the Cisco ACI CNI Plug-in configuration.

Procedure

Step 1 Apply the new Cisco ACI CNI Plug-in configuration.

This step starts the rolling upgrade.

Example:

Kubernetes syntax:

```
# kubectl apply -f aci_cni_new_config.yaml
```

Example:

OpenShift syntax:

```
# oc apply -f aci_cni_new_config.yaml
```

Step 2 Wait a few minutes, and then verify that all the daemons are set to UP-TO-DATE and AVAILABLE.

Example:

Kubernetes syntax for Cisco ACI CNI versions 5.0 and later:

```
kubectl -n aci-containers-system get daemonset
NAME                                DESIRED  CURRENT  READY  UP-TO-DATE  AVAILABLE  NODE  SELECTOR  AGE
aci-containers-host                5         5        5      5            5          <none>    17d
aci-containers-openvswitch         5         5        5      5            5          <none>    17d
```

Example:

Kubernetes syntax for Cisco ACI CNI versions earlier than 5.0:

```
kubectl -n kube-system get daemonset
NAME                                DESIRED  CURRENT  READY  UP-TO-DATE  AVAILABLE  NODE  SELECTOR  AGE
aci-containers-host                5         5        5      5            5          <none>    17d
aci-containers-openvswitch         5         5        5      5            5          <none>    17d
```

Example:

OpenShift syntax:

```
oc -n aci-containers-system get daemonset
NAME                                DESIRED  CURRENT  READY  UP-TO-DATE  AVAILABLE  NODE SELECTOR  AGE
aci-containers-host                6        6        6      6           6          <none>         172d
aci-containers-openvswitch 6        6        6      6           6          <none>         172d
```



Americas Headquarters
Cisco Systems, Inc.
San Jose, CA 95134-1706
USA

Asia Pacific Headquarters
CiscoSystems(USA)Pte.Ltd.
Singapore

Europe Headquarters
CiscoSystemsInternationalBV
Amsterdam,TheNetherlands

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