802.1X

This chapter contains the following sections:

802.1X Overview

802.1X defines a client-server based access control and authentication protocol that restricts unauthorized clients from connecting to a LAN through publicly accessible ports. The authentication server authenticates each client connected to a Cisco NX-OS device port.

Until the client is authenticated, 802.1X access control allows only Extensible Authentication Protocol over LAN (EAPOL) traffic through the port to which the client is connected. After authentication is successful, normal traffic can pass through the port.

The RADIUS distributed client/server system allows you to secure networks against unauthorized access. In the Cisco ACI implementation, RADIUS clients run on the ToRs and send authentication and accounting requests to a central RADIUS server that contains all user authentication and network service access information.

Host Support

The 802.1X feature can restrict traffic on a port with the following modes:

- **Single-host Mode**—Allows traffic from only one endpoint device on the 802.1X port. Once the endpoint device is authenticated, the APIC puts the port in the authorized state. When the endpoint device leaves the port, the APIC put the port back into the unauthorized state. A security violation in 802.1X is defined as a detection of frames sourced from any MAC address other than the single MAC address authorized as a result of successful authentication. In this case, the interface on which this security association violation is detected (EAPOL frame from the other MAC address) will be disabled. Single host mode is

applicable only for host-to-switch topology and when a single host is connected to the Layer 2 (Ethernet access port) or Layer 3 port (routed port) of the APIC.

- **Multi-host Mode**—Allows multiple hosts per port but only the first one gets authenticated. The port is moved to the authorized state after the successful authorization of the first host. Subsequent hosts are not required to be authorized to gain network access once the port is in the authorized state. If the port becomes unauthorized when reauthentication fails or an EAPOL logoff message is received, all attached hosts are denied access to the network. The capability of the interface to shut down upon security association violation is disabled in multiple host mode. This mode is applicable for both switch-to-switch and host-to-switch topologies
- **Multi-Auth Mode**—Allows multiple hosts and all hosts are authenticated separately.

Note Each host must have the same EPG/VLAN information.

- **Multi-Domain Mode**—For separate data and voice domain. For use with IP-Phones.

Authentication Modes

ACI 802.1X supports the following authentication modes:

- **EAP**—The authenticator then sends an EAP-request/identity frame to the supplicant to request its identity (typically, the authenticator sends an initial identity/request frame followed by one or more requests for authentication information). When the supplicant receives the frame, it responds with an EAP-response/identity frame.
- **MAB**—MAC Authentication Bypass (MAB) is supported as the fallback authentication mode. MAB enables port-based access control using the MAC address of the endpoint. A MAB-enabled port can be dynamically enabled or disabled based on the MAC address of the device that connects to it. Prior to MAB, the endpoint's identity is unknown and all traffic is blocked. The switch examines a single packet to learn and authenticate the source MAC address. After MAB succeeds, the endpoint's identity is known and all traffic from that endpoint is allowed. The switch performs source MAC address filtering to help ensure that only the MAB-authenticated endpoint is allowed to send traffic.

Guidelines and Limitations

802.1X port-based authentication has the following configuration guidelines and limitations:

- The Cisco ACI supports 802.1X authentication only on physical ports.
- The Cisco ACI does not support 802.1X authentication on port channels or subinterfaces.
- The Cisco ACI supports 802.1X authentication on member ports of a port channel but not on the port channel itself.
- Member ports with and without 802.1X configuration can coexist in a port channel. However, you must ensure the identical 802.1X configuration on all the member ports in order for channeling to operate with 802.1X

- When you enable 802.1X authentication, supplicants are authenticated before any other Layer 2 or Layer 3 features are enabled on an Ethernet interface.
- 802.1X is supported only on a leaf chassis that is EX or FX type.
- 802.1X is only supported Fabric Access Ports. 802.1X is not supported on Port-Channels, or Virtual-Port-Channels.
- IPv6 is not supported for dot1x clients in the 3.2(1) release.
- While downgrading to earlier releases especially in cases where certain interface config (host mode and auth type) is unsupported in that release, dot1x authentication type defaults to none. Host-mode would need to be manually re-configured to either single host/multi host depending on whatever is desired. This is to ensure that the user configures only the supported modes/auth-types in that release and doesn't run into unsupported scenarios.
- Multi-Auth supports 1 voice client and multiple data clients (all belonging to same data vlan/epg).
- Fail-epg/vlan under 802.1X node authentication policy is a mandatory configuration.
- Multi-domain more than 1 voice and 1 data client puts the port in security disabled state.
- The following platforms are not supported for 802.1X:
  - N9K-C9396PX
  - N9K-M12PQ
  - N9K-C93128TX
  - N9K-M12PQ
- When you use 802.1x in an ACI fabric with strong encryption enabled, the IP packet containing the certificate may exceed 1500 bytes. If you configure reachability to the authenticator over the out-of-band (OOB) interface, packets are automatically fragmented. However, the ACI fabric does not support packet fragmentation. To allow forwarding of packets larger than 1500 bytes when in-band management is used, use the following two options:
  - **Control Plane MTU Change:** Adjust the control plane MTU settings. For detailed instructions, please refer to the Cisco APIC System Management Configuration Guide.

Note This is a global fabric-wide value used for all other protocols.

  - **Ensure Jumbo MTU Support:** Verify that all devices along the path can forward jumbo MTU packets.

Configuration Overview

The 802.1X and RADIUS processes are started only when enabled by APIC. Internally, this means dot1x process is started when 802.1X Inst MO is created and radius process is created when radius entity is created.

Dot1x based authentication must be enabled on each interface for authenticating users connected on that interface otherwise the behavior is unchanged.

RADIUS server configuration is done separately from dot1x configuration. RADIUS configuration defines a list of RADIUS servers and a way to reach them. Dot1x configuration contains a reference to RADIUS group (or default group) to use for authentication.

Both 802.1X and RADIUS configuration must be done for successful authentication. Order of configuration is not important but if there is no RADIUS configuration then 802.1X authentication cannot be successful.

Configuring 802.1X Port Authentication Using the APIC GUI

Before you begin

Configure a RADIUS Provider policy.

Procedure

Step 1 On the menu bar, click **Fabric** > **External Access Policies** > **Policies** > **Interface** > **802.1X Port Authentication** and perform the following actions:

a) Right click on **802.1X Port Authentication**, to open **Create 802.1X Port Authentication Policy**.
b) In the **Name** field, enter a name for the policy.
c) In the **Host Mode** field, select the policy mode. The modes are:

- **Multi Auth**—For allowing multiple hosts and all hosts are authenticated separately.

  Note Each host must have the same EPG/VLAN information.

- **Multi Domain**—For separate data and voice domain. For use with IP-Phones.
- **Multi Host**—For allowing multiple hosts per port but only the first one gets authenticated.
- **Single Host**—For allowing only one host per port.

d) If your device does not support 802.1X then in the **MAC Auth** field, select **EAP_FALLBACK_MAB** and click **Submit**.

Step 2 To associate the **802.1X Port Authentication Policy** to a Fabric Access Group, navigate to **Fabric** > **External Access Policies** > **Interfaces** > **Leaf Interfaces** > **Policy Groups** > **Leaf Access Port** and perform the following actions:

a) Right click on **Leaf Access Port**, to open **Create Leaf Access Port Policy Group**.
b) In the **Name** field, enter a name for the policy.
c) In the **802.1X Port Authentication Policy** field, select the policy previously created and click **Submit**.

Configuring 802.1X Node Authentication Using the APIC GUI

Before you begin

Configure a RADIUS Provider policy.

Procedure

Step 1 On the menu bar, click **Fabric** > **External Access Policies** > **Policies** > **Switch** > **802.1X Node Authentication** and perform the following actions:

a) Right click on **802.1X Node Authentication**, to open **Create 802.1X Node Authentication Policy**.
b) In the **Name** field, enter a name for the policy.
c) In the **Failed-auth EPG** field, select the tenant, application profile, and EPG to deploy to in the case of failed authentication.
d) In the **Failed-auth VLAN**. select the VLAN to deploy to in the case of failed authentication.

Step 2 To associate the **802.1X Node Authentication Policy** to a Leaf Switch Policy Group, navigate to **Fabric** > **External Access Policies** > **Switches** > **Leaf Switches** > **Policy Groups** and perform the following actions:

a) Right click on **Policy Groups**, to open **Create Access Switch Policy Group**.
b) In the **Name** field, enter a name for the policy.
c) In the **802.1X Node Authentication Policy** field, select the policy previously created and click **Submit**.

Step 3 To associate the **802.1X Node Authentication Policy** to a Leaf Interface Profile, navigate to **Fabric** > **External Access Policies** > **Interfaces** > **Leaf Interfaces** > **Profiles** and perform the following actions:

a) Right click on **Profiles**, to open **Create Leaf Interface Profile**.
b) In the **Name** field, enter a name for the policy.
c) Expand the **Interface Selectors** table, to open the **Create Access Port Selector** dialog box and enter the **Name** and **Interface IDs** information.
d) In the **Interface Policy Group** field, select the policy previously created and click **OK** and **Submit**.

Configuring 802.1X Port Authentication Using the NX-OS Style CLI

Procedure

Step 1 Configure a Policy Group:

Example:

```
apic1# configure
apic1(config)#
apic1(config)# template policy-group mypol
apic1(config-pol-grp-if)# switchport port-authentication mydot1x
apic1(config-port-authentication)# host-mode multi-host
apic1(config-port-authentication)# no shutdown
apic1(config-port-authentication)# exit
apic1(config-pol-grp-if)# exit
```

Step 2 Configure the leaf interface profile:

Example:

```
apic1(config)#
apic1(config)#leaf-interface-profile myprofile
apic1(config-leaf-if-profile)#leaf-interface-group mygroup
```

```
apic1(config-leaf-if-group)# interface ethernet 1/10-12
apic1(config-leaf-if-group)# policy-group mypol
apic1(config-leaf-if-group)# exit
apic1(config-leaf-if-profile)# exit
```

Step 3 Configure the leaf profile:

Example:

```
apic1(config)#
apic1(config)# leaf-profile myleafprofile
apic1(config-leaf-profile)# leaf-group myleafgrp
apic1(config-leaf-group)# leaf 101
apic1(config-leaf-group)# exit
```

Step 4 Apply an interface policy on the leaf switch profile:

Example:

```
apic1(config-leaf-profile)# leaf-interface-profile myprofile
apic1(config-leaf-group)# exit
```

Configuring 802.1X Node Authentication Using NX-OS Style CLI

Procedure

Step 1 Configure the radius authentication group:

Example:

```
apic1# configure
apic1(config)#
apic1(config)# aaa group server radius myradiusgrp
apic1(config-radius)#server 192.168.0.100 priority 1
apic1(config-radius)#exit
```

Step 2 Configure node level port authentication policy:

Example:

```
apic1(config)# policy-map type port-authentication mydot1x
apic1(config-pmap-port-authentication)#radius-provider-group myradiusgrp
apic1(config-pmap-port-authentication)#fail-auth-vlan 2001
apic1(config-pmap-port-authentication)#fail-auth-epg tenant tn1 application ap1 epg epg256
apic1(config)# exit
```

Step 3 Configure policy group and specify port authentication policy in the group:

Example:

```
apic1(config)#template leaf-policy-group lpg2
apic1(config-leaf-policy-group)# port-authentication mydot1x
apic1(config-leaf-policy-group)#exit
```

Step 4 Configure the leaf switch profile:

Example:

```
apic1(config)# leaf-profile mylp2
apic1(config-leaf-profile)#leaf-group mylg2
apic1(config-leaf-group)#  leaf-policy-group lpg2
apic1(config-leaf-group)#exit
```

Configuring 802.1X Port Authentication Using the REST API

Procedure

Create a 802.1X port authentication policy:

Example:

```
<polUni>
<infraInfra>
  <l2PortAuthPol adminSt="enabled" annotation="" descr="" dn="uni/infra/portauthpol-test21"
hostMode="multi-auth" name="test21" nameAlias="" ownerKey="" ownerTag="">
    <l2PortAuthCfgPol annotation="" macAuth="bypass" maxReauthReq="2" maxReq="2"   reAuthPeriod="3600"
 serverTimeout="30" suppTimeout="30" txPeriod="30"/>
  </l2PortAuthPol>
</infraInfra>
</polUni>

Modify:
<polUni>
<infraInfra>
  <l2PortAuthPol adminSt="enabled" annotation="" descr="" dn="uni/infra/portauthpol-test21"
hostMode="multi-domain" name="test21" nameAlias="" ownerKey="" ownerTag="" >
    <l2PortAuthCfgPol annotation="" macAuth="eap" maxReauthReq="2" maxReq="2"   reAuthPeriod="3600"
 serverTimeout="30" suppTimeout="30" txPeriod="30"/>
  </l2PortAuthPol>
</infraInfra>
</polUni>

Delete:
<polUni>
<infraInfra>
  <l2PortAuthPol adminSt="enabled" annotation="" descr="" dn="uni/infra/portauthpol-test21"
hostMode="multi-host" name="test21" nameAlias="" ownerKey="" ownerTag="" status="deleted">
    <l2PortAuthCfgPol annotation="" macAuth="bypass" maxReauthReq="2" maxReq="2"   reAuthPeriod="3600"
 serverTimeout="30" suppTimeout="30" txPeriod="30" status="deleted"/>
  </l2PortAuthPol>
</infraInfra>
</polUni>
```

Configuring 802.1X Node Authentication Using the REST API

Procedure

Configure a 802.1X node authentication policy:

Example:

```
<polUni>
<infraInfra>
  <l2NodeAuthPol annotation="" descr="" dn="uni/infra/nodeauthpol-802-node-2"
failAuthEpg="tn-t2,ap-ap,epg-epg1" failAuthVlan="vlan-2078" name="802-node-2" nameAlias="" ownerKey=""
 ownerTag="">
<l2RsAaaRadiusProviderGroup annotation="" tDn="uni/userext/radiusext/radiusprovidergroup-radius-grp"/>
</l2NodeAuthPol>
</infraInfra>
</polUni>

Modify:
<polUni>
<infraInfra>
  <l2NodeAuthPol annotation="" descr="" dn="uni/infra/nodeauthpol-802-node-2"
failAuthEpg="tn-t2,ap-ap,epg-epg1" failAuthVlan="vlan-2066" name="802-node-2" nameAlias="" ownerKey=""
 ownerTag="" status="deleted">
<l2RsAaaRadiusProviderGroup annotation="" tDn="uni/userext/radiusext/radiusprovidergroup-radius-grp"/>
</l2NodeAuthPol>
</infraInfra>
</polUni>

Delete:
<polUni>
<infraInfra>
  <l2NodeAuthPol annotation="" descr="" dn="uni/infra/nodeauthpol-802-node-2"
failAuthEpg="tn-t2,ap-ap,epg-epg1" failAuthVlan="vlan-2078" name="802-node-2" nameAlias="" ownerKey=""
 ownerTag="" status="deleted">
<l2RsAaaRadiusProviderGroup annotation="" tDn="uni/userext/radiusext/radiusprovidergroup-radius-grp"
 status="deleted"/>
</l2NodeAuthPol>
</infraInfra>
</polUni>
```