RADIUS, TACACS+, LDAP, RSA, SAML, OAuth 2, and DUO

This chapter contains the following sections:

Overview

This article provides step by step instructions on how to enable RADIUS, TACACS+, LDAP, RSA, DUO, SAML, OAuth 2 users to access the APIC. It assumes the reader is thoroughly familiar with the *Cisco Application Centric Infrastructure Fundamentals* manual, especially the User Access, Authentication, and Accounting chapter.

Note In the case of a disaster scenario such as the loss of all but one APIC in the cluster, APIC disables remote authentication. In this scenario, only a local administrator account can log into the fabric devices.

Note Remote users for AAA Authentication with shell:domains=all/read-all/ will not be able to access Leaf switches and Spine switches in the fabric for security purposes. This pertains to all version up to 4.0(1h).

User IDs in the APIC Bash Shell

User IDs on the APIC for the Linux shell are generated within the APIC for local users. Users whose authentication credential is managed on external servers, the user ID for the Linux shell can be specified in the cisco-av-pair. Omitting the (16001) in the above cisco-av-pair is legal, in which case the remote user gets a default Linux user ID of 23999. Linux User IDs are used during bash sessions, allowing standard Linux permissions enforcement. Also, all managed objects created by a user are marked as created-by that user's Linux user ID.

The following is an example of a user ID as seen in the APIC Bash shell:

```
admin@ifav17-ifc1:~> touch myfile
admin@ifav17-ifc1:~> ls -l myfile
-rw-rw-r-- 1 admin admin 0 Apr 13 21:43 myfile
admin@ifav17-ifc1:~> ls -ln myfile
-rw-rw-r-- 1 15374 15374 0 Apr 13 21:43 myfile
admin@ifav17-ifc1:~> id
uid=15374(admin) gid=15374(admin) groups=15374(admin)
```

AV Pair on the External Authentication Server

The Cisco APIC requires that an administrator configure a Cisco AV Pair on an external authentication server. The Cisco AV pair specifies the APIC required RBAC roles and privileges for the user. The Cisco AV Pair format is the same for RADIUS, LDAP, or TACACS+.

To configure a Cisco AV Pair on an external authentication server, an administrator adds a Cisco AV pair to the existing user record. The Cisco AV pair format is as follows:

```
shell:domains =
domainA/writeRole1|writeRole2|writeRole3/readRole1|readRole2,
domainB/writeRole1|writeRole2|writeRole3/readRole1|readRole2
shell:domains =
domainA/writeRole1|writeRole2|writeRole3/readRole1|readRole2,
domainB/writeRole1|writeRole2|writeRole3/readRole1|readRole2(16003)
```

Starting with Cisco APIC release 2.1, if no UNIX ID is provided in AV Pair, the APIC allocates the unique UNIX user ID internally.

Note The APIC Cisco AV-pair format is compatible and can co-exist with other Cisco AV-pair formats. APIC will pick up the first matching AV-pair from all the AV-pairs.

Starting with release 3.1(x), the AV Pair shell:domains=all//admin allows you to assign Read-only privileges to users and provide them access to the switches and run commands.

The APIC supports the following regexes:

```
shell:domains\\s*[=:]\\s*((\\S+?/\\S*?/\\S*?)(,\\S+?/\\S*?/\\S*?){0,31})(\\(\\d+\\))$
shell:domains\\s*[=:]\\s*((\\S+?/\\S*?/\\S*?)(,\\S+?/\\S*?/\\S*?){0,31})$
```

Examples:

- Example 1: A Cisco AV Pair that contains a single Security domain with only writeRoles:

```
shell:domains=domainA/writeRole1|writeRole2/
```

- Example 2: A Cisco AV Pair that contains a single Security domain with only readRoles:

```
shell:domains=domainA//readRole1|readRole2
```

Note The "/" character is a separator between writeRoles and readRoles per Security domain and is required even if only one type of role is to be used.

The Cisco AVpair string is case sensitive. Although a fault may not be seen, using mismatching cases for the domain name or roles could lead to unexpected privileges being given.

An example configuration for an open RADIUS server (/etc/raddb/users) is as follows:

```
aaa-network-admin Cleartext-Password := "<password>"
Cisco-avpair = "shell:domains = all/aaa/read-all(16001)"
```

Best Practice for Assigning AV Pairs

As best practice, we recommend that you assign unique UNIX user IDs in the range of 16000 to 23999 for the AV pairs that are assigned to users when in the Bash shell (using SSH, Telnet, or serial/KVM consoles). If a situation arises when the Cisco AV pair does not provide a UNIX user ID, the user is assigned a user id of 23999 or similar number from the range that also enables the user's home directories, files, and processes accessible to remote users with a UNIX ID of 23999.

Note Beginning with the 5.3(1) release, telnet is not supported.

To ensure that your remote authentication server does not explicitly assign a UNIX ID in its cisco-av-pair response, open an SSH session to the Cisco Application Policy Infrastructure Controller (APIC) and log in as an administrator using a remote user account. After you have logged in, run the following commands (replace "*userid*" with the username with which you logged in):

```
admin@apic1:remoteuser-userid> cd /mit/uni/userext/remoteuser-userid
admin@apic1:remoteuser-userid> cat summary
```

The Cisco AV pair string is case sensitive. Although a fault may not be seen, using mismatching cases for the domain name or roles could lead to unexpected privileges being given.

Configuring an AV Pair on the External Authentication Server

The numerical value within the parentheses in the attribute/value (AV) pair string is used as the UNIX user ID of the user who is logged in using Secure Shell (SSH) or Telnet.

Note Beginning with the 5.3(1) release, telnet is not supported.

SUMMARY STEPS

1. Configure an AV pair on the external authentication server.

DETAILED STEPS

Procedure

Configure an AV pair on the external authentication server.

The Cisco AV pair definition is as follows (Cisco supports AV pairs with and without UNIX user IDs specified):

Example:

```
shell:domains = domainA/writeRole1|writeRole2|writeRole3/readRole1|readRole2,domainB/writeRole1|writeRole2|writeRole3/readRole1|readRole2
shell:domains =
domainA/writeRole1|writeRole2|writeRole3/readRole1|readRole2,domainB/writeRole1|writeRole2|writeRole3/readRole1|readRole2(8101)

These are the boost regexes supported by APIC:
uid_regex("shell:domains\\s*[=:]\\s*((\\S+?/\\S*?/\\S*?),\\S+?/\\S*?/\\S*?){0,31})(\\(\\d+\\))$");
regex("shell:domains\\s*[=:]\\s*((\\S+?/\\S*?/\\S*?),\\S+?/\\S*?/\\S*?){0,31})$");
```

The following is an example:

```
shell:domains = coke/tenant-admin/read-all,pepsi//read-all(16001)
```

Configuring a Remote User

Instead of configuring local users, you can point the APIC at the centralized enterprise credential datacenter. The APIC supports Lightweight Directory Access Protocol (LDAP), active directory, RADIUS, and TACACS+.

Note When an APIC is in minority (disconnected from the cluster), remote logins can fail because the ACI is a distributed system and the user information is distributed across APICS. Local logins, however, continue to work because they are local to the APIC.

Starting with the 3.1(1) release, **Server Monitoring** can be configured through RADIUS, TACACS+, LDAP, and RSA to determine whether the respective AAA servers are alive or not. Server monitoring feature uses the respective protocol login to check for server aliveness. For example, a LDAP server will use ldap login and a Radius server will use radius login with server monitoring to determine server aliveness.

To configure a remote user authenticated through an external authentication provider, you must meet the following prerequisites:

- The DNS configuration should have already been resolved with the hostname of the RADIUS server.

- You must configure the management subnet.

Configuring a Remote User Using the NX-OS Style CLI

Instead of configuring local users, you can point the APIC at the centralized enterprise credential datacenter. The APIC supports Lightweight Directory Access Protocol (LDAP), active directory, RADIUS, and TACACS+.

To configure a remote user authenticated through an external authentication provider, you must meet the following prerequisites:

- The DNS configuration should have already been resolved with the hostname of the RADIUS server.
- You must configure the management subnet.

Changing the Default Behavior for Remote Users with Missing or Bad Cisco AV Pairs

Procedure

Step 1 On the menu bar, choose **Admin** > **Authentication** > **AAA** > **Policy** tab.

Step 2 From the **Remote user login policy** drop-down list, choose **Assign Default Role**.

The default value is **No Login**. The **Assign Default Role** option assigns the minimal read-only privileges to users that have missing or bad Cisco AV Pairs. Bad AV Pairs are those AV Pairs that fail the parsing rules.

Changing Default Behavior for Remote Users with Missing or Bad Cisco AV Pairs Using the NX-OS Style CLI

The Cisco APIC requires that an administrator configure a Cisco AV Pair on an external authentication server. To do so, an administrator adds a Cisco AV pair to the existing user record. The Cisco AV pair specifies the APIC required RBAC roles and privileges for the user. The Cisco AV Pair format is the same for RADIUS, LDAP, or TACACS+. One AV pair format contains a Cisco UNIX user ID and one does not. Both are correct if all remote users have the same role and mutual file access is acceptable. If the UNIX user ID is not specified, ID 23999 is applied by the APIC system, and more than one role/read privilege is specified to any AV Pair user. This can cause users to have higher or lower permissions than configured through the group settings. This topic explains how to change the bahavior if that is not acceptable.

To change the default behavior for remote users with missing or bad Cisco AV pairs using the NX-OS CLI:

Procedure

Step 1 In the NX-OS CLI, start in Configuration mode.

Example:

```
apic1#
apic1# configure
```

Step 2 Configure the aaa user default role.

Example:

```
apic1(config)# aaa user default-role
 assign-default-role  assign-default-role
 no-login             no-login
```

Step 3 Configure the aaa authentication login methods.

Example:

```
apic1(config)# aaa authentication
 login  Configure methods for login

apic1(config)# aaa authentication login
 console  Configure console methods
 default  Configure default methods
 domain   Configure domain methods

apic1(config)# aaa authentication login console
 <CR>

apic1(config)# aaa authentication login domain
 WORD      Login domain name
 fallback
```

Login Domains

A login domain defines the authentication domain for a user. Login domains can be set to the Local, LDAP, RADIUS, TACACS+, DUO, SAML, RSA, or OAuth 2 authentication mechanisms. When accessing the system from REST, the CLI, or the GUI, the APIC enables the user to select the correct authentication domain.

For example, in the REST scenario, the username is prefixed with a string so that the full login username looks as follows:

`apic:<domain>\<username>`

If accessing the system from the GUI, the APIC offers a drop-down list of domains for the user to select. If no `apic: domain` is specified, the default authentication domain servers are used to look up the username.

Starting in ACI version 1.0(2x), the login domain fallback of the APIC defaults local. If the default authentication is set to a non-local method and the console authentication method is also set to a non-local method and both non-local methods do not automatically fall back to local authentication, the APIC can still be accessed via local authentication.

To access the APIC fallback local authentication, use the following strings:

- From the GUI, use *apic:fallback\\username*.
- From the REST API, use *apic#fallback\\username*.

Note Do not change the fallback login domain. Doing so could result in being locked out of the system.

Creating Login Domain Using the GUI

Before you begin

- The ACI fabric is installed, Application Policy Infrastructure Controllers (APICs) are online, and the APIC cluster is formed and healthy.
- The login domain name, realm, and remote server provider group are available to define the authentication domain for the user.

Procedure

Step 1 In the APIC, create Login Domain.

a) On the menu bar, choose **Admin** > **Authentication** > **AAA** > **Policy** tab.

b) In the **Properties** pane, click the **Actions** icon > **Create Login Domain**. You can also create a login domain by clicking **+** displayed against **Login Domains**.

c) In the **Create Login Domain** pane, specify the following:

- The user configured domain name.
- Description of the login domain.
- The realm to verify the identity of an entity (person or device) accessing the fabric devices.The options available in the **Realm** drop-down list are discussed here:

  1. For Release 4.2(x) and earlier, choose a security method from Local, LDAP, RADIUS, TACACS+, RSA, or SAML for processing authentication requests.
  2. For Release 5.0(x) and later, choose a security method from DUO Proxy LDAP, DUO Proxy Radius, LDAP, RADIUS, TACACS+, RSA, SAML, OAuth 2, or Local for processing authentication requests.

  Note
  If LDAP, RADIUS, or TACACS+ is specified as the default security method and the associated provider group specified in this dialog is not available to provide authentication during a user login, fallback local authentication is not executed by the APIC server unless is specifically configured to do so.

- A RADIUS provider group for a group of remote servers supporting the RADIUS protocol for authentication.
- A TACACS+ provider group for a group of remote servers supporting the TACACS+ protocol for authentication.
- An LDAP provider group for a group of remote servers supporting the LDAP protocol for authentication.
- A RSA provider group for a group of remote servers supporting the RSA protocol for authentication.
- A SAML provider group for a group of remote servers supporting the SAML protocol for authentication.
- An OAuth 2 provider group for a group of remote servers supporting the OAuth 2 protocol for authentication.

Step 2 Click **Submit**.

RADIUS Authentication

Remote Authentication Dial-In User Service (RADIUS) is a networking protocol that provides centralized authentication, authorization, and accounting (AAA) management for users who connect and use a network service.

To configure users on RADIUS servers, the APIC administrator must configure the required attributes (`shell:domains`) using the `cisco-av-pair` attribute. The default user role is network-operator.

The SNMPv3 authentication protocol options are SHA and MD5. The privacy protocol options are AES-128 and DES. If these options are not specified in the `cisco-av-pair` attribute, MD5 and DES are the default authentication protocols.

For example, SNMPv3 authentication and privacy protocol attributes can be specified as follows:

```
snmpv3:auth=SHA priv=AES-128
```

Similarly, the list of domains would be as follows:

```
shell:domains="domainA domainB …"
```

Configuring APIC for RADIUS Access

Before you begin

- The ACI fabric is installed, Application Policy Infrastructure Controllers (APICs) are online, and the APIC cluster is formed and healthy.
- The RADIUS server host name or IP address, port, authorization protocol, and key are available.
- The APIC management endpoint group is available.

Procedure

Step 1 In the APIC, create the RADIUS provider.

a) On the menu bar, choose **Admin** > **Authentication** > **RADIUS** tab.
b) Click the **Actions** icon > **Create RADIUS Provider**.
c) In the displayed pop-up window, specify the RADIUS host name (or IP address), description, port, protocol, key, timeout, retries, management endpoint group, and select if server monitoring needs to be enabled or not. .

Note
If the APIC is configured for in-band management connectivity, out-of-band management does not work for authentication. With the APIC release2.1(1x), you can set a global toggle between In-band and out-of-band as the default management connectivity between the APIC server and other external management devices.

For toggling in-band or out-of-band management in the APIC GUI:

- Prior to Release 2.2(1x): In the **Navigation** pane, choose **Fabric** > **Fabric Policies** > **Global Policies** > **Connectivity Preferences**. In the **Work Pane** select either **inband** or **ooband**.
- For Release 2.2(x) and 2.3(x): In the **Navigation** pane, choose **Fabric** > **Fabric Policies** > **Global Policies** > **APIC Connectivity Preferences**. In the **Work Pane** select either **inband** or **ooband**.
- For Release 3.0(1x) or later: In the **Navigation** pane, choose **System** > **System Settings** > **APIC Connectivity Preferences**. In the **Work Pane** select either **inband** or **ooband**.

Step 2 Create the **Login Domain** for TACACS+.

For the detailed procedure, see Creating Login Domain Using the GUI, on page 7.

What to do next

This completes the APIC RADIUS configuration steps. Next, configure the RADIUS server.

Configuring Radius in APIC Using REST API

```
HTTP POST to https://{{apichost}}/api/node/mo/.xml
<aaaRadiusProvider authPort="1812" authProtocol="pap" descr="myradius"
    monitorServer="disabled"
    name="server.radius.local" key="mykey"
    retries="1" timeout="5"/>
```

To configure a login domain for RADIUS using REST API:

```
HTTP POST to https://{{apichost}}/api/node/mo/.xml
<aaaUserEp  descr="" dn="uni/userext"  name=""  pwdStrengthCheck="yes" rn=""
status="modified">
    <aaaLoginDomain descr="" name="RadDom" rn="logindomain-RadDom" status="created">
        <aaaDomainAuth name="" providerGroup="RadDom" realm="radius" rn="domainauth"
status="created"/>
    </aaaLoginDomain>
   <aaaRadiusEp descr="" name="" retries="1" rn="radiusext" status="modified" timeout="5">

        <aaaRadiusProviderGroup descr=""  name="RadDom"  rn="radiusprovidergroup-RadDom"
status="created">
            <aaaProviderRef descr="acs" name="radius1.server.com" order="1"
                rn="providerref-radius.server.com" status="created" />
            <aaaProviderRef descr="acs" name="radius2.server.com" order="2"
                rn="providerref-radius2.server.com" status="created" />
        </aaaRadiusProviderGroup>
    </aaaRadiusEp>
</aaaUserEp>
```

TACACS+ Authentication

Terminal Access Controller Access Control System Plus (TACACS+) is another remote AAA protocol that is supported by Cisco devices. TACACS+ has the following advantages over RADIUS authentication:

- Provides independent AAA facilities. For example, the Cisco Application Policy Infrastructure Controller (APIC) can authorize access without authenticating.
- Uses TCP to send data between the AAA client and server, enabling reliable transfers with a connection-oriented protocol.
- Encrypts the entire protocol payload between the switch and the AAA server to ensure higher data confidentiality. RADIUS encrypts passwords only.
- Uses the av-pairs that are syntactically and configurationally different than RADIUS but the Cisco APIC supports `shell:domains`.

The following XML example configures the Cisco Application Centric Infrastructure (ACI) fabric to work with a TACACS+ provider at IP address 10.193.208.9:

```
<aaaTacacsPlusProvider name="10.193.208.9"
           key="test123"
           authProtocol="pap"/>
```

Note While the examples provided here use IPv4 addresses, IPv6 addresses could also be used.

The following guidelines and limitations apply when using TACACS+:

- The TACACS server and TACACs ports must be reachable by ping.
- The TACACS server with the highest priority is considered first to be the primary server.

Configuring APIC for TACACS+ Access

Before you begin

- The Cisco Application Centric Infrastructure (ACI) fabric is installed, Application Policy Infrastructure Controllers (APICs) are online, and the APIC cluster is formed and healthy.
- The TACACS+ server host name or IP address, port, and key are available.
- The APIC management endpoint group is available.

Procedure

Step 1 In the APIC, create the **TACACS+ Provider**.

a) On the menu bar, choose **Admin** > **Authentication** > **TACACS** tab.
b) Click the **Actions** icon > **Create TACACS+ Provider**.
c) In the displayed pop-up window, specify the TACACS+ host name (or IP address), description, port, authorization protocol, key, timeout, retries, management endpoint group, and select if server monitoring needs to be enabled or not.

Note

If the APIC is configured for in-band management connectivity, out-of-band management does not work for authentication. With the APIC release2.1(1x), you can set a global toggle between In-band and out-of-band as the default management connectivity between the APIC server and other external management devices.

For toggling in-band or out-of-band management in the APIC GUI:

- Prior to Release 2.2(1x): In the **Navigation** pane, choose **Fabric** > **Fabric Policies** > **Global Policies** > **Connectivity Preferences**. In the **Work Pane** select either **inband** or **ooband**.
- For Release 2.2(x) and 2.3(x): In the **Navigation** pane, choose **Fabric** > **Fabric Policies** > **Global Policies** > **APIC Connectivity Preferences**. In the **Work Pane** select either **inband** or **ooband**.
- For Release 3.0(1x) or later: In the **Navigation** pane, choose **System** > **System Settings** > **APIC Connectivity Preferences**. In the **Work Pane** select either **inband** or **ooband**.

Step 2 Create the **Login Domain** for TACACS+.

For the detailed procedure, see Creating Login Domain Using the GUI, on page 7.

What to do next

This completes the APIC TACACS+ configuration steps. Next, if a RAIDUS server will also be used, configure the APIC for RADIUS. If only a TACACS+ server will be used, go to the ACS server configuration topic below.

Configuring TACACS in APIC Using the REST API

Make sure that you configure aaaTacacsPlusProviderGroup with the same name as the name of the TACACS login domain.

```
HTTP POST to https://{{apichost}}/api/node/mo/.xml
<aaaTacacsPlusProvider name="server.tacacs.local"
    authProtocol="pap"
    monitorServer="enabled" monitoringUser="user1" monitoringPassword="mypwd"
    port="49" retries="1" key="mykey" timeout="15" />
```

To configure a login domain for TACACS using the REST API:

```
HTTP POST to https://{{apichost}}/api/node/mo/.xml
<aaaUserEp  descr="" dn="uni/userext"  name=""  pwdStrengthCheck="yes" rn=""
status="modified">
    <aaaLoginDomain  descr=""  name="Tacacs" nameAlias=""  rn="logindomain-Tacacs"
status="created,modified">
        <aaaDomainAuth  descr="" name="" nameAlias=""  providerGroup="Tacacs"
            realm="tacacs" rn="domainauth" status="created,modified"/>
    </aaaLoginDomain>
    <aaaTacacsPlusEp  descr="" name="" nameAlias=""  retries="1" rn="tacacsext"
status="created,modified" timeout="5">
        <aaaTacacsPlusProviderGroup  descr="" name="Tacacs" nameAlias=""
            rn="tacacsplusprovidergroup-Tacacs" status="created,modified">
           <aaaProviderRef descr="testing" name="tacacs.server.com" nameAlias="" order="1"

                rn="providerref-tacacs.server.com"  status="created,modified" />
          <aaaProviderRef descr="testing" name="tacacs2.server.com" nameAlias="" order="2"
```

```
                    rn="providerref-tacacs2.server.com"  status="created,modified" />
            </aaaTacacsPlusProviderGroup>
        </aaaTacacsPlusEp>
    </aaaUserEp>
```

Configuring a Cisco Secure Access Control Server for RADIUS and TACACS+ Access to the APIC

Before you begin

- The Cisco Secure Access Control Server (ACS) version 5.5 is installed and online.

Note ACS v5.5 was used to document these steps. Other versions of ACS might support this task but the GUI procedures might vary accordingly.

- The Cisco Application Policy Infrastructure Controller (Cisco APIC) RADIUS or TACACS+ keys are available (or keys for both if both will be configured).
- The APICs are installed and online; the APIC cluster is formed and healthy.
- The RADIUS or TACACS+ port, authorization protocol, and key are available.

Procedure

Step 1 Log in to the ACS server to configure the APIC as a client.

a) Navigate to **Network Resources** > **Network Devices Groups** > **Network Devices and AAA Clients**.

b) Specify the client name, the APIC in-band IP address, select the TACACS+ or RADIUS (or both) authentication options.

Note
If the only RADIUS or TACACS+ authentication is needed, select only the needed option.

c) Specify the authentication details such as Shared Secret (key), and port as appropriate for the authentication option(s).

Note
The **Shared Secret**(s) must match the APIC **Provider** key(s).

Step 2 Create the Identity Group.

a) Navigate to **Users and Identity Stores** > **Internal Groups** option.

b) Specify the **Name**, and **Parent Group** as appropriate.

Step 3 Map users to the Identity Group.

a) In the **Navigation** pane, click the **Users and Identity Stores** > **Internal Identity Stores** > **Users** option.

b) Specify the user **Name**, and **Identity Group** as appropriate.

Step 4 Create the Policy Element.

a) Navigate to the **Policy Elements** option.
b) For RADIUS, specify the Authorization and Permissions > Network Access > Authorization Profiles **Name**. For TACACS+, specify the Authorization and Permissions > Device Administration > Shell Profile **Name** as appropriate.
c) For RADIUS, specify the **Attribute** as `cisco-av-pair`, **Type** as string, and the **Value** as **`shell:domains = <domain>/<role>/,<domain>// role`** as appropriate. For TACACS+, specify the **Attribute** as `cisco-av-pair`, **Requirement** as Mandatory, and the **Value** as **`shell:domains = <domain>/<role>/,<domain>// role`** as appropriate.

   The syntax of the **Value** field determines whether write privileges are granted:

   - For read/write privileges, the syntax is `shell:domains = <domain>/<role>/`.
   - For read-only privileges, the syntax is `shell:domains = <domain>// <role>`.

   For example, if the *cisco-av-pair* has a value of `shell:domains = solar/admin/,common// read-all`, then `solar` is the security domain, `admin` is the role that gives write privileges to this user in the security domain called `solar`, `common` is the tenant common, and `read-all` is the role with read privileges that gives this user read privileges to all of the tenant common.

Step 5 Create a service selection rule.

a) For RADIUS, create a service selection rule to associate the Identity Group with the Policy Element by navigating to **Access Policies** > **Default Device Network Access Identity** > **Authorization** and specifying the rule **Name**, **Status**, and **Conditions** as appropriate, and **Add** the `Internal Users:UserIdentityGroup in ALL Groups:<identity group name>`.
b) For TACACS+, create a service selection rule to associate the Identity Group with the Shell Profile by navigating to **Access Policies** > **Default Device Admin Identity** > **Authorization**. Specify the rule **Name**, **Conditions**, and **Select** the **Shell Profile** as appropriate.

What to do next

Use the newly created RADIUS and TACACS+ users to log in to the APIC. Verify that the users have access to the correct APIC security domain according to the assigned RBAC roles and privileges. The users should not have access to items that have not been explicitly permitted. Read and write access rights should match those configured for that user.

LDAP/Active Directory Authentication

Similar to RADIUS and TACACS+, LDAP allows a network element to retrieve AAA credentials that can be used to authenticate and then authorize the user to perform certain actions. An added certificate authority configuration can be performed by an administrator to enable LDAPS (LDAP over SSL) trust and prevent man-in-the-middle attacks.

The XML example below configures the ACI fabric to work with an LDAP provider at IP address 10.30.12.128.

Note While the examples provided here use IPv4 addresses, IPv6 addresses could also be used.

```
<aaaLdapProvider name="10.30.12.128"
          rootdn="CN=Manager,DC=ifc,DC=com"
```

```
     basedn="DC=ifc,DC=com"
     SSLValidationLevel="strict"
     attribute="CiscoAVPair"
     enableSSL="yes"
key="myldappwd"
     filter="cn=$userid"
     port="636" />
```

Note For LDAP configurations, best practice is to use **`CiscoAVPair`** as the attribute string. If customer faces the issue using Object ID 1.3.6.1.4.1.9.22.1, an additional Object ID 1.3.6.1.4.1.9.2742.1-5 can also be used in the LDAP server.

Instead of configuring the Cisco AVPair, you have the option to create LDAP group maps in the APIC.

Configuring LDAP

There are two options for LDAP configurations: you can configure a Cisco AVPair or configure LDAP group maps in the APIC. This section contains instructions for both configuration options.

Configuring Windows Server 2012 LDAP for APIC Access with Cisco AVPair

Before you begin

- First, configure the LDAP server, then configure the Cisco Application Policy Infrastructure Controller (Cisco APIC) for LDAP access.
- The Microsoft Windows Server 2012 is installed and online.
- The Microsoft Windows Server 2012 Server Manager ADSI Edit tool is installed. To install ADSI Edit, follow the instructions in the Windows Server 2012 Server Manager help.
- `CiscoAVPair` attribute specifications: Common Name = **`CiscoAVPair`**, LDAP Display Name = **`CiscoAVPair`**, Unique X500 Object ID = 1.3.6.1.4.1.9.22.1, Description = **`CiscoAVPair`**, Syntax = **`Case Sensitive String`**.

Note For LDAP configurations, best practice is to use **`CiscoAVPair`** as the attribute string. If customer faces the issue using Object ID 1.3.6.1.4.1.9.22.1, an additional Object ID 1.3.6.1.4.1.9.2742.1-5 can also be used in the LDAP server.

- A Microsoft Windows Server 2012 user account is available that will enable the following:
  - Running ADSI Edit to add the `CiscoAVPair` attribute to the Active Directory (AD) Schema.
  - Configuring an Active Directory LDAP user to have `CiscoAVPair` attribute permissions.
- Port 636 is required for configuring LDAP integration with SSL/TLS.

Procedure

Step 1 Log in to an Active Directory (AD) server as a domain administrator.

Step 2 Add the `CiscoAVPair` attribute to the AD schema.

a) Navigate to **Start** > **Run**, type **`mmc`** and press **Enter**.
The Microsoft Management Console (MMC) opens.

b) Navigate to **File** > **Add/Remove Sanp-in** > **Add**.

c) In the **Add Standalonee Snap-in** dialog box, select the **Active Directory Schema** and click **Add**.
The MMC **Console** opens.

d) Right-click the **Attributes** folder, select the **Create Attribute** option.
The **Create New Attribute** dialog box opens.

e) Enter **`CiscoAVPair`** for the **Common Name** , **`CiscoAVPair`** for the **LDAP Display Name**, **`1.3.6.1.4.1.9.22.1`** for the **Unique X500 Object ID**, and select **`Case Sensitive String`** for the **Syntax**.

f) Click **OK** to save the attribute.

Step 3 Update the **User Properties** class to include the **CiscoAVPair** attribute.

a) In the MMC **Console**, expand the **Classes** folder, right-click the **user** class, and choose **Properties**.
The **user Properties** dialog box opens.

b) Click the **Attributes** tab, and click **Add** to open the **Select Schema Object** window.

c) In the **Select a schema object:** list, choose **`CiscoAVPair`**, and click **Apply**.

d) In the MMC **Console**, right-click the **Active Directory Schema**, and select **Reload the Schema**.

Step 4 Configure the **`CiscoAVPair`** attribute permissions.

Now that the LDAP includes the **`CiscoAVPair`** attributes, LDAP users need to be granted Cisco APIC permission by assigning them Cisco APIC RBAC roles.

a) In the ADSI Edit dialog box, locate a user who needs access to the Cisco APIC.

b) Right-click on the user name, and choose **Properties**.
The **<user> Properties** dialog box opens.

c) Click the **Attribute Editor** tab, select the *CiscoAVPair* attribute, and enter the *Value* as **`shell:domains = <domain>/<role>/,<domain>// role`**.

For example, if the *CiscoAVPair* has a value of `shell:domains = solar/admin/,common// read-all(16001)`, then `solar` is the security domain, `admin` is the role for this user that gives write privileges to this user in the security domain called `solar`, `common` is the Cisco Application Centric Infrastructure (Cisco ACI) tenant common, and `read-all(16001)` is the role with read privileges that gives this user read privileges to all of the Cisco ACI tenant common.

d) Click **OK** to save the changes and close the **<user> Properties** dialog box.

The LDAP server is configured to access the Cisco APIC.

What to do next

Configure the Cisco APIC for LDAP access.

Configuring APIC for LDAP Access

Before you begin

- The Cisco Application Centric Infrastructure (ACI) fabric is installed, Application Policy Infrastructure Controllers (APICs) are online, and the APIC cluster is formed and healthy.
- The LDAP server host name or IP address, port, bind DN, Base DN, and password are available.
- The APIC management endpoint group is available.

Procedure

Step 1 In the APIC, configure the LDAP Provider.

a) On the menu bar, choose **Admin** > **Authentication** > **LDAP** > **Providers** tab.
b) Click the **Actions** icon > **Create LDAP Provider**.
c) In the displayed pop-up window, specify the LDAP host name (or IP address), port, bind DN, base DN, password, attribute, retries, timeout, SSL certificate validation level, filter type, management endpoint group, and select if server monitoring needs to be enabled or not. .

Note

- The bind DN is the string that the APIC uses to log in to the LDAP server. The APIC uses this account to validate the remote user attempting to log in. The base DN is the container name and path in the LDAP server where the APIC searches for the remote user account. This is where the password is validated. Filter is used to locate the attribute that the APIC requests to use for the *cisco-av-pair*. This contains the user authorization and assigned RBAC roles for use on the APIC. The APIC requests the attribute from the LDAP server.
- **Attribute** field—Enter one of the following:
  - For LDAP server configurations with a Cisco AVPair, enter **`CiscoAVPair`**.
  - For LDAP server configurations with an LDAP group map, enter **`memberOf`**.
- If the APIC is configured for in-band management connectivity, choosing an out-of-band management endpoint group for LDAP access does not take effect. Alternatively, an out-of-band over an in-band management endpoint group can connect a LDAP server, but requires configuring a static route for the LDAP server. The sample configuration procedures in this document use an APIC in-band management endpoint group.

Step 2 Create the **Login Domain** for LDAP.

For the detailed procedure, see Creating Login Domain Using the GUI, on page 7.

What to do next

This completes the APIC LDAP configuration steps. Next, test the APIC LDAP login access.

Configuring LDAP Group Map Rules on the Cisco APIC

Configuring an LDAP group map on the Cisco APIC requires first creating LDAP group map rules. This section explains how to create LDAP group map rules.

Before you begin

An LDAP server is running with a configured group mapping.

Procedure

Step 1 On the menu bar, choose **Admin** > **Authentication** > **LDAP** > **LDAP Group Map Rules** tab.

Step 2 Click the **Actions** icon > **Create LDAP Group Map Rule**.

Step 3 In the displayed pop-up window, specify the map rule name, description (optional), group DN, and security domain in the appropriate fields then click **Next**. The **Create LDAP Group Map Rule: Roles** dialog appears with security domain options.

Step 4 Click the + to access the Role Name and Role Privilege Type fields.

Step 5 Click the **Role Name** drop-down arrow to choose a role name.

Step 6 Click the **Role Privilege Type** drop-down arrow to choose a role privilege type (**Read** or **Write**) .

Repeat Step 4 to 6 to add additional roles to the LDAP group map rule.

Step 7 When finished, click **Finished**.

What to do next

After specifying the LDAP group map rules, create an LDAP group map.

Configuring an LDAP Group Map on the Cisco APIC

Configuring an LDAP group map on the Cisco APIC requires first creating LDAP group map rules. This section explains how to create an LDAP group map.

Before you begin

- A running LDAP server is configured with group mapping.
- LDAP group map rules have been configured.

Procedure

Step 1 On the menu bar, choose **Admin** > **Authentication** > **LDAP** > **LDAP Group Maps** tab.

Step 2 Click the **Actions** icon > **Create LDAP Group Map** .

Step 3 In the displayed pop-up window, specify the map name, description (optional) and rule.

a) In the **Rules** field, click (+), then click the Name drop-down arrow to choose a specified LDAP group map rule. Click **Update**.

Repeat this step to add additional rules to the LDAP group map.

Step 4 When finished, click **Submit**.

Multi-factor Authentication with DUO

Cisco APIC supports multi-factor authentication with Duo security. Duo security itself does not act as repository for user identities. It offers second factor (2F) authentication on top of an organization's existing authentication, which could be on-premesis or cloud-based. Second factor authentication with Duo occurs once the user has finished the authentication with the organization's primary authentication source.

Duo supports three types of 2F authentication methods after you complete authentication with the primary authentication source:

- Notification push on mobile using the Duo mobile app on smartphones.
- Phone call on your registered phone or mobile numbers.
- Passcode that is generated on the Duo mobile app.

The user is authenticated using the following servers:

- The Duo proxy RADIUS server uses the multi-factor authentication in Cisco APIC to authenticate a distributed client/server system using RADIUS PAP primary authentication method.
- The Duo proxy LDAP server uses the multi-factor authentication in Cisco APIC to authenticate a remote server using Cisco AVPair or Group Maps authentication method.

Configuring DUO RADIUS Proxy Provider

DUO RADIUS Proxy acts as a proxy RADIUS server that forwards the incoming RADIUS authentication request to the external RADIUS server, waits for response from that server and then if the authentication is successful with the external RADIUS server, it initiates the second factor authentication on the user's secondary device.

Before you begin

- The ACI fabric is installed, Application Policy Infrastructure Controllers (APICs) are online, and the APIC cluster is formed and healthy.

Procedure

Step 1 In the APIC, configure the DUO RADIUS proxy provider.

a) On the menu bar, choose **Admin** > **Authentication** > **DUO** > **Radius** tab.
b) Click the **Actions** icon > **Create DUO Radius Proxy Provider** from the menu.
c) In the displayed pop-up window, specify the following:

- The hostname or IP address of the DUO RADIUS proxy provider.

- The description of the DUO RADIUS proxy provider.
- The authentication port number for the RADIUS service. The range is from 1 to 65535. The default is 1812.
- Key is the secret text string shared between the device and a specific DUO RADIUS proxy server.
- The timeout for communication with a DUO RADIUS proxy provider server. The range is from 1 to 60 seconds. The default is 30 seconds. If set to 0, the AAA provider timeout is used.
- The number of retries when contacting the RADIUS endpoint. The range is from 0 to 5 retries. The default is 1.
- The out-of-band management EPG used to communicate with the DUO RADIUS service.
- Enabling Server Monitoring allows the connectivity of the remote AAA servers to be tested.

Step 2 Click **Submit**.

Configuring DUO LDAP Proxy Provider

Create DUO LDAP proxy providers, DUO LDAP proxy provider groups, and configure the default DUO LDAP proxy authentication settings. Create the global security management properties for DUO LDAP endpoints and DUO LDAP proxy provider groups.

Before you begin

- The ACI fabric is installed, Application Policy Infrastructure Controllers (APICs) are online, and the APIC cluster is formed and healthy.

Procedure

Step 1 In the APIC, configure the DUO LDAP proxy provider.

a) On the menu bar, choose **Admin** > **Authentication** > **DUO** > **LDAP** > **Providers** tab.
b) Click the **Actions** icon > **Create DUO LDAP Proxy Provider**.
c) In the displayed pop-up window, specify the following:

- The hostname or IP address of the DUO LDAP proxy provider.
- The description of the DUO LDAP proxy provider.
- The service port number for the LDAP service. The range is from 1 to 65535. The default is 389.
- The Bind DN is the string that the APIC uses to log in to the DUO LDAP proxy server.
- The DUO LDAP base DN to be used in a user search.
- The password for the DUO LDAP database account specified in the **Bind** DN field.
- The timeout for communication with an DUO LDAP proxy provider server. The range is from 0 to 60 seconds. The default is 30 seconds. If set to 0, the AAA provider timeout is used.

- The number of retries when contacting the DUO LDAP proxy endpoint.
- Enables an SSL connection with the DUO LDAP proxy provider. The default is disabled.
- The attribute to be downloaded that contains user role and domain information.
  1. For DUO LDAP proxy provider server configurations with a Cisco AVPair, enter `CiscoAVPair`.
  2. For DUO LDAP proxy provider server configurations with a DUO LDAP proxy group map, enter `memberOf`.

     Note
     For DUO LDAP configurations, best practice is to use `AciCiscoAVPair` as the attribute. This avoids problems related to the limitation DUO LDAP proxy servers not allowing overlapping object identifiers (OID); that is, the `ciscoAVPair` OID is already in use.

- The out-of-band management EPG used to communicate with the DUO LDAP proxy server.
- The DUO LDAP proxy provider server SSL Certificate validation level. The value can be:
  1. Permissive—A debugging knob to help diagnose DUO LDAP SSL Certificate issues.
  2. Strict—A level that should be used when in production.
- The DUO LDAP filter to be used in a user search.
- Enabling Server Monitoring allows the connectivity of the remote AAA servers to be tested.

Step 2 Click **Submit**.

Configuring DUO Proxy Using the REST API

```
The URL for all XML data :
POST  https://{{apichost}}/api/node/mo/.xml
```

The following are example configurations for Duo with proxy RADIUS and proxy LDAP servers.

RADIUS Configuration

- Add DUO RADIUS proxy provider:

```
<aaaRadiusProvider authPort="1812" authProtocol="pap" descr="duoradius"
       dn="uni/userext/duoext/radiusprovider-duoproxy.host.com"
       monitorServer="disabled" monitoringUser=""
       name="duoproxy.host.com"  key="mypasswd"
       retries="1" status="created" timeout="30"/>
```

- Add Login Domain with DUO RADIUS proxy provider:

```
<aaaUserEp  descr="" dn="uni/userext"  name="" pwdStrengthCheck="yes" rn=""
status="modified">
   <aaaLoginDomain descr="" name="DuoRadDom" rn="logindomain-DuoRadDom" status="created">

        <aaaDomainAuth descr="" name="" providerGroup="DuoRadDom" realm="radius"
realmSubType="duo" rn="domainauth" status="created"/>
    </aaaLoginDomain>
    <aaaDuoEp descr="" name="" retries="1" rn="duoext" status="modified" timeout="40">

        <aaaDuoProviderGroup name="DuoRadDom" providerType="radius"
```

```
secFacAuthMethods="auto,push"
             rn="duoprovidergroup-DuoRadDom" status="created">
            <aaaProviderRef descr="duoradproxy" name="duoproxy.host.com" order="1"
                rn="providerref-duoproxy.host.com" status="created" />
        </aaaDuoProviderGroup>
    </aaaDuoEp>
</aaaUserEp>
```

LDAP Configuration

- Add DUO LDAP proxy provider with the attribute `Cisco AVPair`:

```
<aaaLdapProvider name="duoproxy.host.com"
    SSLValidationLevel="strict"
    attribute="CiscoAvPair"
    basedn="CN=Users,DC=host,DC=com"
    dn="uni/userext/duoext/ldapprovider-duoproxy.host.com" enableSSL="no"
filter="cn=$userid"
    monitorServer="disabled"
    port="389" retries="1"
    rootdn="CN=admin,CN=Users,DC=host,DC=com"
    timeout="60"
    key="12345"/>
```

- Add DUO LDAP proxy provider with the attribute `memberOf`:

```
<aaaLdapProvider name="duoproxy.host.com"
    SSLValidationLevel="strict"
    attribute="memberOf"
    basedn="CN=Users,DC=host,DC=com"
    dn="uni/userext/duoext/ldapprovider-duoproxy.host.com" enableSSL="no"
filter="cn=$userid"
    monitorServer="disabled"
    port="389" retries="1"
    rootdn="CN=admin,CN=Users,DC=host,DC=com"
    timeout="60"
    key="12345"/>
```

- Add LDAP GroupMap rule:

```
<aaaLdapGroupMapRule name="DuoEmpRule" dn="uni/userext/duoext/ldapgroupmaprule-DuoEmpRule"

       groupdn="CN=Employee,CN=Users,DC=host,DC=com" status="created">
       <aaaUserDomain name="all" rn="userdomain-all" status="created,modified">
          <aaaUserRole name="fabric-admin" privType="writePriv" rn="role-fabric-admin"
 status="created,modified"/>
       </aaaUserDomain>
</aaaLdapGroupMapRule>
```

- Add LDAP GroupMap:

```
<aaaLdapGroupMap name="DuoEmpGroupMap" dn="uni/userext/duoext/ldapgroupmap-DuoEmpGroupMap"
  status="created">
    <aaaLdapGroupMapRuleRef name="DuoEmpRule" rn="ldapgroupmapruleref-DuoEmpRule"
status="created"/>
</aaaLdapGroupMap>
```

- Add DUO LDAP Login Domain using GroupMap:

```
<polUni>
   <aaaUserEp dn="uni/userext" name="" pwdStrengthCheck="yes" rn="" status="modified">

        <aaaDuoEp attribute="memberOf" basedn="" filter="sAMAccountName=$userid"
            name="" retries="1" rn="duoext" status="modified" timeout="30">
            <aaaDuoProviderGroup name="DuoLdapDom" authChoice="LdapGroupMap"
providerType="ldap"
```

```
                    rn="duoprovidergroup-DuoLdapDom" ldapGroupMapRef="DuoEmpGroupMap"
    secFacAuthMethods="auto,push" status="modified">
                   <aaaProviderRef name="duoproxy.host.com" order="1"
                       rn="providerref-duoproxy.host.com" status="modified"/>
               </aaaDuoProviderGroup>
           </aaaDuoEp>
          <aaaLoginDomain name="DuoLdapDom" rn="logindomain-DuoLdapDom" status="modified">

               <aaaDomainAuth name="" providerGroup="DuoLdapDom" realm="ldap"
    realmSubType="duo" rn="domainauth" status="modified"/>
           </aaaLoginDomain>
       </aaaUserEp>
    </polUni>
```

Get Login Domain for GUI

The GET URL for login domains:

```
GET https://apic.host.com/api/aaaListDomains.json
```

```
{   "totalCount": "5",
    "imdata": [{
            "name": "DuoRadDom",
            "type": "DUO",
            "secAuths": "auto,push"
        }, {
            "name": "DuoLdapDom",
            "type": "DUO",
            "secAuths": "auto,push"
        }, {
            "name": "RadDom",
            "type": "OTHER"
        }, {
            "name": "LdapDom",
            "type": "OTHER"
        }, {
            "name": "DefaultAuth",
            "guiBanner": "",
            "type": "OTHER"
        }
    ] }
```

RSA Secure ID Authentication

RSA Authentication provides a token which can be used in combination with a fixed key in many different ways to create the password. It supports both hardware and software tokens.

Configuring APIC for RSA Access Using the GUI

Before you begin

- The ACI fabric is installed, Application Policy Infrastructure Controllers (APICs) are online, and the APIC cluster is formed and healthy.
- The RSA server host name or IP address, port, authorization protocol, and key are available.
- The APIC management endpoint group is available.

Procedure

Step 1 In the APIC, create the RSA provider.

a) On the menu bar, choose **Admin** > **Authentication** > **RSA** tab.
b) Click the **Actions** icon > **Create RSA Provider** .
c) In the displayed pop-up window, specify the RSA host name (or IP address), port, protocol, and management endpoint group.

Step 2 Create the login domain for RSA.

Step 3 Create the **Login Domain** for RSA.

For the detailed procedure, see Creating Login Domain Using the GUI, on page 7.

What to do next

This completes the APIC RSA configuration steps. Next, configure the RSA server.

SAML Authentication

SAML is an XML-based open standard data format that enables administrators to access a defined set of Cisco collaboration applications seamlessly after signing into one of those applications. SAML describes the exchange of security related information between trusted business partners. It is an authentication protocol used by service providers to authenticate a user. SAML enables exchange of security authentication information between an Identity Provider (IdP) and a service provider.

SAML SSO uses the SAML 2.0 protocol to offer cross-domain and cross-product single sign-on for Cisco collaboration solutions. SAML 2.0 enables SSO across Cisco applications and enables federation between Cisco applications and an IdP. SAML 2.0 allows Cisco administrative users to access secure web domains to exchange user authentication and authorization data, between an IdP and a Service Provider while maintaining high security levels. The feature provides secure mechanisms to use common credentials and relevant information across various applications.

The authorization for SAML SSO Admin access is based on Role-Based Access Control (RBAC) configured locally on Cisco collaboration applications.

SAML SSO establishes a Circle of Trust (CoT) by exchanging metadata and certificates as part of the provisioning process between the IdP and the Service Provider. The Service Provider trusts the IdP's user information to provide access to the various services or applications.

Note Service providers are no longer involved in authentication. SAML 2.0 delegates authentication away from the service providers and to the IdPs.

The client authenticates against the IdP, and the IdP grants an Assertion to the client. The client presents the Assertion to the Service Provider. Since there is a CoT established, the Service Provider trusts the Assertion and grants access to the client.

Enabling SAML SSO results in several advantages:

- It reduces password fatigue by removing the need for entering different user name and password combinations.
- It transfers the authentication from your system that hosts the applications to a third party system. UsingSAML SSO, you can create a circle of trust between an IdP and a service provider. The service provider trusts and relies on the IdP to authenticate the users.
- It protects and secures authentication information. It provides encryption functions to protect authentication information passed between the IdP, service provider, and user. SAML SSO can also hide authentication messages passed between the IdP and the service provider from any external user.
- It improves productivity because you spend less time re-entering credentials for the same identity.
- It reduces costs as fewer help desk calls are made for password reset, thereby leading to more savings.

Basic Elements of SAML

- Client (the user's client): This is a browser-based client or a client that can leverage a browser instance for authentication. For example, a system administrator's browser.
- Service provider: This is the application or service that the client is trying to access.
- An Identity Provider (IdP) server: This is the entity that authenticates user credentials and issues SAML Assertions.
- Lightweight Directory Access Protocol (LDAP) users: These users are integrated with an LDAP directory, for example Microsoft Active Directory or OpenLDAP. Non-LDAP users reside locally on the Unified Communications server.
- SAML Assertion: It consists of pieces of security information that are transferred from IdPs to the service provider for user authentication. An assertion is an XML document that contains trusted statements about a subject including, for example, a username and privileges. SAML assertions are usually digitally signed to ensure their authenticity.
- SAML Request: This is an authentication request that is generated by a Unified Communications application. To authenticate the LDAP user, Unified Communications application delegates an authentication request to the IdP.
- Circle of Trust (CoT): It consists of the various service providers that share and authenticate against one IdP in common.
- Metadata: This is an XML file generated by an ACI application as well as an IdP. The exchange of SAML metadata builds a trust relationship between the IdP and the service provider.
- Assertion Consumer Service (ACS) URL: This URL instructs the IdPs where to post assertions. The ACS URL tells the IdP to post the final SAML response to a particular URL.

Note All in-scope services requiring authentication use SAML 2.0 as the SSO mechanism.

Supported IdPs and SAML Components

Supported IdPs

Identity Provider (IdP) is an authentication module that creates, maintains, and manages identity information for users, systems, or services and also provides authentication to other applications and service providers within a distributed network.

With SAML SSO, IdPs provide authentication options based on the user role or log in options for each of the Cisco collaboration applications. The IdPs store and validate the user credentials and generate a SAML response that allows the user to access the service provider protected resources.

Note You must be familiar with your IdP service, and ensure that it is currently installed and operational.

The APIC SAML SSO feature has been tested with following IdPs:

- https://technet.microsoft.com/en-us/library/cc772128(WS.10).aspx
- PingFederate: https://docs.pingidentity.com/pingfederate/latest/pf_pf_landing_page.html

SAML Components

A SAML SSO solution is based on a particular combination of assertions, protocols, bindings, and profiles. The various assertions are exchanged among applications and sites using the protocols and bindings, and those assertions authenticate the users among sites. The SAML components are as follows:

- SAML Assertion: It defines the structure and content of the information that is transferred from IdPs to service providers. It consists of packets of security information and contains statements that service providers use for various levels of access-control decisions.SAML SSO provides the following types of statements:
  - Authentication statements- These statements assert to the service provider about the method of authentication that occurs between the IdP and the browser at a particular time.
  - Attribute statements- These statements assert about certain attributes (name-value pairs) that are associated with the user. The attribute assertions contain specific information about the user. The service providers use attributes to make access-control decisions.
- SAML protocol: A SAML protocol defines how the SAML requests for and gets assertions. This protocol is responsible for the SAML request and response elements that consist of certain SAML elements or assertions. The SAML 2.0 contains the following protocols:
  - Assertion Query and Request Protocol
  - Authentication Request Protocol
- SAML binding: A SAML binding specifies the mapping of SAML assertion and/or protocol message exchanges with standard messaging formats or communication protocols like SOAP exchanges. ACI supports the following SAML 2.0 bindings:
  - HTTP Redirect (GET) Binding
  - HTTP POST Binding

- SAML profile: A SAML profile provides a detailed description of the combination of SAML assertions, protocols, and bindings to support well-defined use cases.

NTP Setup

In SAML SSO, Network Time Protocol (NTP) enables clock synchronization between the APIC and IdP. SAML is a time sensitive protocol and the IdP determines the time-based validity of a SAML assertion. If the IdP and the APIC clocks are not synchronized, the assertion becomes invalid and stops the SAML SSO feature. The maximum allowed time difference between the IdP and the APIC is 3 seconds.

Note For SAML SSO to work, you must install the correct NTP setup and make sure that the time difference between the IdP and the APIC does not exceed 3 seconds. If IdP and APIC clocks are not synchronized, the user will be redirected back to the APIC's login page even after successful authentication on IdP.

DNS Setup

Domain Name System (DNS) enables the mapping of host names and network services to IP addresses within a network or networks. DNS server(s) deployed within a network provide a database that maps network services to hostnames and, in turn, hostnames to IP addresses. Devices on the network can query the DNS server and receive IP addresses for other devices in the network, thereby facilitating communication between network devices.

In summary, APIC and Idp should be able to resolve each other's fully qualified domain names to IP addresses and should be resolvable by the client.

Certificate Authority

Cisco recommends using server certificates that are signed by one of the following types of Certificate Authority (CA):

- **Public CA**—A third-party company verifies the server identity and issues a trusted certificate.
- **Private CA**—You create and manage a local CA and issue trusted certificates.

The signing process varies for each product and can vary between server versions. It is beyond the scope of this document to provide detailed steps for every version of each server. Refer the appropriate server documentation for detailed instructions on how to get certificates signed by a CA.

If you get server certificates signed by a public CA, the public CA should already have a root certificate present in the trust store on the client computer. In this case, you do not need to import root certificates on the client computers. You should import root certificates if the certificates are signed by a CA that does not already exist in the trust store, such as a private CA. In SAML SSO, the IdP and service providers must have CA signed certificates with the correct domains in the CN or SAN. If the correct CA certificates are not validated, the browser issues a pop up warning.

If the APIC's trust store does not include the root certificate of the IdP, a new certificate authority should be created. This Certificate Authority should be used later while configuring the SAML Provider on APIC.

Configuring APIC for SAML Access

Note SAML-based authentication is only for the Cisco APIC GUI and not for the CLI or REST API. Also, SAML is not applicable for leaf switches and spine switches. You cannot configure SAML configuration using the Cisco APIC CLI.

Before you begin

- The Cisco Application Centric Infrastructure (ACI) fabric is installed, Application Policy Infrastructure Controllers (APICs) are online, and the Cisco APIC cluster is formed and healthy.
- The SAML server host name or IP address, and the IdP's metadata URL are available.
- The Cisco APIC management endpoint group is available.
- Set up the following:
  - Time synchronization and NTP
  - A DNS service policy to connect with the DNS providers
  - A custom certificate for Cisco ACI HTTPS access

For more information, see the *Cisco APIC Basic Configuration Guide*.

Procedure

Step 1 In the Cisco APIC GUI, create the **SAML Provider**.

a) On the menu bar, choose **Admin** > **AAA** .
b) In the Navigation pane, choose **Authentication** .
c) In the Work pane, choose **SAML** > **Providers**.
d) Choose **Actions** > **Create SAML Provider**.
e) Enter the SAML host name (or IP address), and description (optional).
f) Choose the **Identity Provider**.
g) Enter the **IdP Entity ID** for the SAML-based service.
h) Enter the **SP Entity ID**, which is the service provider entity ID. You can get the ID from the service provider. The format is as follows:

```
https://apic-id/api/aaaLoginSSO.json?name=domain-name
```

i) Enter the **Metadata URL provided by IDP**.
   - For ADFS, the IdP Metadata URL format is as follows:

```
https://FQDN-of-ADFS/FederationMetadata/2007-06/FederationMetadata.xml
```

j) Configure the **HTTPS Proxy** if it is needed to access the IdP metadata URL.
k) Choose the **Certificate Authority** if IdP is signed by a Private CA.
l) Choose the **Timeout** (in seconds) value.

m) Choose the **Signature Algorithm for Requests** authentication type for the user requests from the drop-down list.
n) Put a check in the **Sign SAML Auth Requests** check box to enable signing SAML authentication requests.
o) Put a check in the **Sign SAML Response Message** check box to enable signing SAML response messages.
p) Put a check in the **Sign Assertions in SAML Response** check box to enable signing assertions in SAML responses.
q) Put a check in the **Encrypt SAML Assertions** check box to enable encryption in SAML assertions.

Step 2 Create the **Login Domain** for SAML.

For the detailed procedure, see Creating Login Domain Using the GUI, on page 7.

Configuring SAML in APIC Using REST API

```
HTTP POST to https://{{apichost}}/api/node/mo/.xml
<aaaSamlProvider dn="uni/userext/samlext/samlprovider-auth.pingone.asia"
  entityId="https://192.168.32.1/api/aaaLoginSSO.json"
  spEntityId="https://apic.host.com"
  guiBannerMessage="" httpsProxy="proxy.server.com" idP="ping identity"
  metadataUrl="https://auth.pingone.com/c5f09515-6ce4-4776-a770-3d2ad98f078e/
    saml20/metadata/9a0cd2a5-daf6-40dd-9004-c562221fc6e2"
  monitorServer="disabled" name="auth.pingone.asia" retries="1"
  sigAlg="SIG_RSA_SHA256" status="created,modified" timeout="5" tp="pingonecert"
  wantAssertionsEncrypted="no" wantAssertionsSigned="yes" wantRequestsSigned="yes"
  wantResponseSigned="yes"/>
```

Note The metadataUrl value has a line break for readability. However, do not include a line break in the actual value.

Setting Up a Relying Party Trust in AD FS

Add relying party trust in AD FS Management Console:

Procedure

Step 1 Add relying party trust:

a) Login to AD FS Management Console on your AD FS server, Navigate to **ADFS** > **Trust Relationships** > **Relying Party Trusts** and right-click on **Add Relying Party Trust** and click **Start**.
b) Choose **Enter data about the relying party manually** or **Import data about relying party from a file (skip the steps d, e, f and g)** by importing the metadata file generated using the **Download SAML Metadata** option available on the corresponding login domain setup in APIC.
c) Enter your preferred **Display Name** for the relying party trust and click **Next**.
d) Choose AD FS Profile and click **Next**.
e) Click **Next** again.
f) Select **Enable support for the SAML 2.0 Web SSO Protocol** and enter **Relying party SAML2.0 SSO service UR** as *https://<APIC_hostname>/api/aaaLoginSSO.json?name=<Login_domain_name>* and click **Next**.
g) Enter the **Relying party trust identifier** – *https://<APIC_hostname>/api/aaaLoginSSO.json*

h) Choose **I do not want to configure multi-factor authentication settings for this relying party trust at this time** and click **Next**.
i) Choose **Permit all users to access this relying party** and click **Next**.
j) Select **Open the Edit Claim rules** dialog for this relying party trust when the wizard closes and click **Close**.

Step 2 Add the following **Claim** rules:

a) Send LDAP Attributes as claims:

- In the **Edit Claim Rules** window, click **Add Rule**.
- Select the **Claim Rule Template** as Send LDAP attributes as **Claims** and click **Next**.
- Enter a **Rule_*Name*** and select **Active Directory** as the Attribute Store.
- Select the reserved User Attribute for storing CiscoAvpair (For Ex: **Department**) as LDAP attribute type and map it to Outgoing Claim Manually Type as **CiscoAvpair**.
- Select **E-Mail-Addresses** on LDAP Attribute and map it to the Outgoing Claim Type **E-mail Address** and click **Finish**.

b) Transform an Incoming Claim:

- Click **Add Rule** again in the **Edit Claim Rules** window, and select **Transform an Incoming Claim as Claim Rule Template** and click **Next**.
- Select **E-Mail Address** as the Incoming claim type.
- Select **Name ID** as Outgoing claim type.
- Select **Transient Identifier** as Outgoing name ID format.

Step 3 To add a cluster of APICs, one can either setup multiple **Relying Party Trusts** or setup single **Relying Party Trust** and add multiple **Relying Party Identifiers** and **SAML Assertion Consumer Endpoints** to it.

a) Adding other APICs in a cluster with same relying party trusts created above.

1. Navigate to **ADFS Management Console** > **ADFS** > **Trust Relationships** > **Relying Party Trusts** and right-click on **CiscoAPIC** > **Properties**.
2. Click on Identifiers tab and add other APICs in cluster as: *https://<APIC2_hostname>/api/aaaLoginSSO.json*, *https://<APIC3_hostname>/api/aaaLoginSSO.json*
3. Click on **Endpoints** tab and Other two APICs by clicking on **Add SAML**. **Add SAML Post Binding**, Index as 1 and Enter trusted URL as: *https://<APIC2_hostname>/api/aaaLoginSSO.json?name=<Login_domain_name>*, and **Add SAML Post Binding** as: *https://<APIC3_hostname>/api/aaaLoginSSO.json?name=<Login_domain_name>*.

Step 4 Message and Assertion need to be signed in ADFS from powershell in ADFS server. For Signing Message and Assertion in ADFS Server:

a) Open Windows Powershell (should be run as Administrator) and execute the below command:
b) Set-AdfsRelyingPartyTrust -TargetName **RelyingpartytrustnameOfCiscoAPIC** -SamlResponseSignature **MessageAndAssertion**.

OAuth 2 Authorization

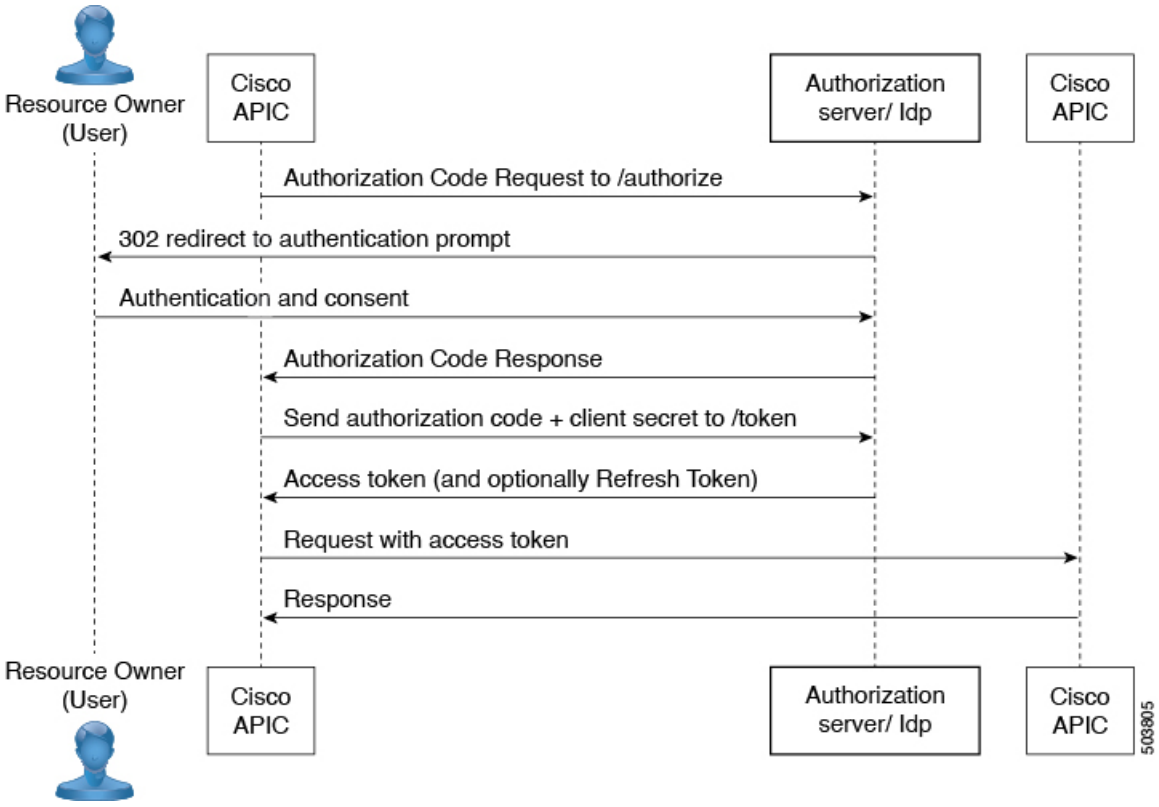
Open Authorization (OAuth) 2.0 is an open-standard authorization protocol. OAuth 2.0 allows you to access an application (Service Provider or SP) that is trusted or approved by an Identity Provider (IdP). OAuth 2.0 uses authorization tokens to provide identity and authorization claims to the consumer application.

For more details about OAuth 2.0, see RFC 6749.

OAuth 2.0 has been designed to support a variety of different client types, which consume REST APIs from service provider applications. This includes both browser applications accessing web services within the enterprise, and applications running on customer mobile devices. OAuth protocol defines multiple mechanisms for getting an authorization token where different mechanisms acknowledge the client type constraints. A simple OAuth example is - when you are trying to login to a website, say “https://service.example.com”, you could be asked to identify yourself using a social media platform login or your email login. If you are logged in to these identity providers, you need not login over and over again. You are authorized (using OAuth) to login to “https://service.example.com”, as soon as you choose one of the options.

OAuth 2.0 Authentication in Cisco ACI

Type of OAuth used in ACI is the *authorization grant flow*. In this method, Cisco APIC first requests an authorization grant by an authenticated user, and APIC then uses the authorization grant to obtain an access token that has the authorization information. The flow is depicted in the following diagram.



Elements of OAuth

- Resource owner(user)— data owner
- Web application— APIC (or Cloud APIC)
- Authorization server (AS) or Identity Provider (IdP) server— that authenticates and authorizes the user
- Resource Server— APIC

Note When the authorization server provides both, ID Token and access token, ID token is preferred over access token for username and CiscoAvpair claims. In case CiscoAvpair is not available in the ID token, both the username and CiscoAvpair claims are taken from the access token, if available. APIC does not combine username and CiscoAvpair claims from both the tokens i.e. it will not consider username from ID token and CiscoAvpair from access token or vice versa. If none of the tokens have CiscoAvpair claim, username from ID token is taken and tried for default authorization if configured.

Configuring OAuth in Cisco APIC

Use this procedure to configure OAuth in Cisco Application Policy Infrastructure Controller (APIC).

Prerequisites

Perform the following actions in an authorization server:

- Create an OAuth application for Cisco APIC. Note the client ID and secret.
- Ensure that authorization policies are setup to allow access to Cisco APIC.
- Note the *authorize* and *token* endpoints that would be used by Cisco Application Centric Infrastructure (ACI).
- Assign users to the application who would be using Cisco APIC.
- Ensure that the *CiscoAvpair* is set correctly for the users for authorization in Cisco ACI.
- Save the certificate chain for the Token URL.

For details about configuring OAuth 2.0 applications on Identity Providers, see the relevant documentation.

Creating a Provider

Use this procedure to create an OAuth 2 provider.

Procedure

Step 1 On the menu bar, choose **Admin** > **Authentication** > **OAuth 2** tab.

Step 2 Click the **Actions** icon > **Create OAuth 2 Provider**.

Step 3 In the **Create OAuth 2 Provider** window that is displayed, enter the following details:

- The host name or IP address for the OAuth 2 provider.
- Description for the provider.

- The contents of the GUI informational banner. The information is displayed before the user is redirected to the Identity Provider login page for authentication. On the GUI Redirect Banner, you must acknowledge the terms and conditions shown in the banner and continue to log on for further access.
- Client identifier of the APIC application on IdP.
- Client secret for the APIC application.
- Confirm the client secret for the APIC application.
- Username attribute in the token. Example: email, sub.
- List of OAuth 2 scopes. Example: "openid profile".
- IdP endpoint authorization URL.
- IdP endpoint token URL.
- The proxy HTTPS server configured here is local to this MO for accessing the metadata url and will not be configured at global level.
- Certificate authority used to contact token and authorization URLs.

  Select a Certificate Authority from the drop-down list. If you do not have one, select Creating a Certificate Authority.
- Timeout in seconds. The length of time the system should spend trying to contact the OAuth 2 endpoint before it times out. The range is from 5 to 30. The default is 5.
- The required management EPG. Example: inband, out-of-band.

Step 4 Click **Submit**.

What to do next

Link the created provider with a login domain. See Creating Login Domain Using the GUI, on page 7.

Creating a Certificate Authority

Use this procedure for creating certificate authorities using the certificate chain used for the token URL.

Procedure

Step 1 On the menu bar, choose **Admin** > **Security** > **Public Key Management** > **Certificate Authorities**.

Step 2 Click **Actions** > **Create Certificate Authority**.

A certificate authority can also be created while creating an OAuth 2 provider.

Step 3 Enter **Name**, **Description**, and **Certificate Chain**.

For obtaining the **Certificate Chain**, follow the procedure shown below.

a) Choose the Token URL from the authorization server.
b) In a browser window, enter the Token URL.
c) Right-click and choose **More Information**.

d) From the displayed pop-up window, click the **New Certificate** button.
e) The Certificate screen is displayed. Download the **PEM (chain)** certificate.
f) Choose a suitable program to open the file.
g) Choose the required certificate from the displayed chain of certificates.

Note
A maximum of eight certificate authorities can be created.

Step 4 Click **Save**.

User Login using OAuth

If you try to login to APIC using the created login domain for OAuth, you will be redirected to the login page of the authorization server (if not authenticated already). After the user authenticates, an authorization code is sent from the authorization server to APIC via the web browser. APIC will then exchange this code for an access token from IdP using the client ID and secret for the APIC application. Access token has the username and authorization details in the *CiscoAvpair*. You will then be logged-in to APIC. On APIC, the logged in user is indicated accordingly.

Configuring OAuth in APIC Using REST API

Use this procedure to configure OAuth in APIC using REST API.

Procedure

Step 1 Create OAuth Provider.

```
<aaaOauthProvider name="app.idpserver1.com" dn="uni/userext/oauthext/oauthprovider-app.idpserver1.com"

  status="created,modified"
  httpsProxy="https://proxy.foo.com:8080"  retries="1" timeout="5"
     usernameAttr="sub"
     scope="openid"
  authzEndpoint="https://app.idpserver1.com/oauth2/abc123/v1/authorize"
  tokenEndpoint="https://app.idpserver1.com/oauth2/abc123/v1/token"
  clientId="clientId1"
  secret="secret1"
  tp="idpcertchain" />
```

Step 2 Create OAuth Login Domain.

```
<aaaUserEp dn="uni/userext" status="created,modified">
  <aaaPingEp dn="uni/userext/pingext" pingCheck="false" retries="1" status="modified" timeout="5"/>
  <aaaLoginDomain dn="uni/userext/logindomain-TOAUTH" name="TOAUTH" status="created,modified">
    <aaaDomainAuth dn="uni/userext/logindomain-TOAUTH/domainauth" providerGroup="TOAUTH" realm="oauth"
 realmSubType="default" status="created,modified"/>
  </aaaLoginDomain>
  <aaaOauthEp rn="oauthext" status="modified">
    <aaaOauthProviderGroup dn="uni/userext/oauthext/oauthprovidergroup-TOAUTH" name="TOAUTH"
status="created,modified">
     <aaaProviderRef dn="uni/userext/oauthext/oauthprovidergroup-TOAUTH/providerref-app.idpserver1.com"
```

```
 name="app.idpserver1.com" order="1" status="created,modified"/>
    </aaaOauthProviderGroup>
  </aaaOauthEp>
</aaaUserEp>
```