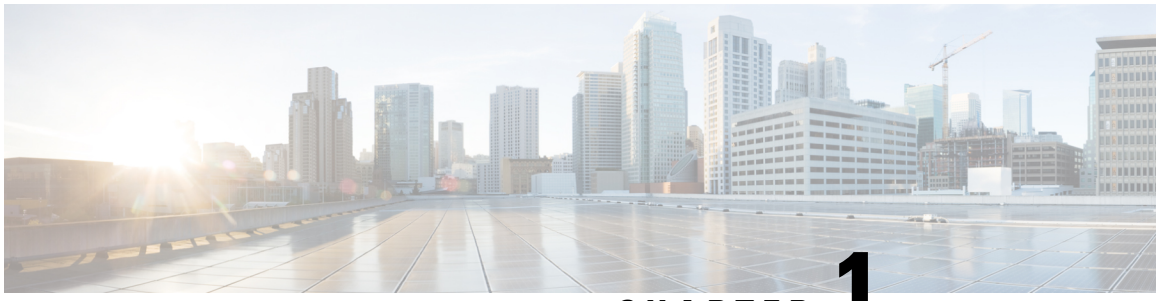
Configuring Trustpoints on Cisco Catalyst 9800 Series Controllers

First Published: 2021-04-26

Americas Headquarters
Cisco Systems, Inc.
170 West Tasman Drive
San Jose, CA 95134-1706
USA
http://www.cisco.com
Tel: 408 526-4000
800 553-NETS (6387)
Fax: 408 527-0883

CONTENTS



CHAPTER 1

Overview of Trustpoints on Catalyst 9800

Note The documentation set for this product strives to use bias-free language. For purposes of this documentation set, bias-free is defined as language that does not imply discrimination based on age, disability, gender, racial identity, ethnic identity, sexual orientation, socioeconomic status, and intersectionality. Exceptions may be present in the documentation due to language that is hardcoded in the user interfaces of the product software, language used based on RFP documentation, or language that is used by a referenced third-party product.

Certificate-based authentication is a method to identify a user, device or machine before they can be granted access to a network. A wireless network comprising of a Wireless LAN Controller (WLC), hereafter referred to as controller, Access Points (AP) and clients, commonly uses certificate-based authentication to validate the identities of peer devices when participating in services such as AP Join, Management Access and Web Authentication. Each service can use different sets of client and server certificates.

But how do devices get their digital identities?

To begin with, each participating device (controller, access point, and client) has its own device certificate and a Certificate Authority (CA) certificate that validates its authenticity. A closer look at the certificates available on the Catalyst 9800 controller shows the following types:

- Cisco installed Manufacturing Installed Certificate (MIC) - On physical appliances (Catalyst 9800-40, Catalyst 9800-80, Catalyst 9800-L), these are by default, factory installed, and widely known as the Cisco installed MIC or Secure Unique Device Identifier (SUDI) device certificate. In addition to this, controllers and access points have a Cisco Manufacturing Certificate Authority (CA) certificate that is used to sign and validate device certificates.
    - Wireless LAN Controller self-signed certificate for virtual controller - The Catalyst 9800-CL, (the virtual instance of the controller) does not come with any manufacturing certificate. In the absence of an identity certificate, it relies on the self-signed certificate that has to be generated by the Day 0 wizard or manually using a script and validated by the local Cisco IOS Certificate Authority. This acts as the Catalyst 9800-CL's local identity certificate and is used for AP Join, Mobility connection and Network Mobility Services Protocol-Connected Mobile Experience (NMSP-CMX) connections.
- IOS-XE device self-signed certificate- The default self-signed certificate is auto-generated during the controller's initial startup if any HTTPS, SSH or NETCONF service is configured on the controller.

The above default certificates provide an easy and out of the box method of early trust between peer devices. However, if you want to provide better security, then you can consider using

- Third-party validated certificates, including Locally Significant Certificates (LSC).

Third-party certificates require a PKI framework that enables encryption of public keys and digital certificates. Along with different authentication protocols, the PKI model works with Certificate Authorities, Root Certificates and asymmetric key encryption to ensure that the digital certificates are securely exchanged over encrypted tunnels during a client and server exchange.

On Catalyst 9800 controllers, these digital certificates are configured and held in containers called trustpoints and used when the devices initiate a secure communication with the other network devices. Trustpoint is one of the most important configuration entities for a PKI client. A trustpoint includes the identity certificate of the CA that signed the device certificate, CA-specific trustpoint configuration parameters, and an association with an enrolled identity certificate.

Trustpoints provide a mapping between the identity certificate and the application/service that needs the certificate. For example, for the SSL/HTTPS server functionality, the *ip http secure-trustpoint <trustpoint name>* tells the controller what identity certificate to present to an SSL client. Depending on your requirement, you can configure many trustpoints.

A Case for Trustpoints

Identity validation using certificates spans across a range of functions and protocols in the Catalyst 9800 wireless environment. Certificates are primarily used for authentication when an Access Point joins the controller using with CAPWAP with DTLS, for Web Administration and Web Authentication using HTTP with TLS and for Local EAP Authentication. Certificates are also used when the controller communicates with Cisco Connected Mobile Experience (CMX), Cisco Digital Network Architecture Center (DNA Center) and Digital Network Architecture Spaces (DNA Spaces). Some of these exchanges require additional configuration whereas others do not require any action from your side.

These scenarios are outlined in the following sections along with their default behaviour and recommended actions.

Use a Trustpoint to Secure Web Administration on Catalyst 9800

The admin interface of the Catalyst 9800 web user interface (WebUI) is usually accessed securely over HTTPS from a remote workstation over a web browser for web administration purposes.

Default behaviour

When you enable a secure HTTP connection, the controller automatically picks one of its certificates, to the best of its judgement, even if none is configured for the HTTP secure server. This can be a self-signed certificate that may be used for future SSL handshakes between the remote workstation (client) and the HTTPS server. However, a self-certified (self-signed) certificate does not provide adequate security, and the connecting client

generates a notification that the certificate is self-certified, and the user has the opportunity to accept or reject the connection.

If you want a dedicated third-party certificate to be used for future SSL handshakes between the controller and the connecting client, it is best to configure a trustpoint to use this certificate. Once an identity certificate is configured and found, it always takes precedence over a self-signed certificate.

We recommend that you point/map the HTTPS application to use controller's trusted third-party certificate for increased security.

Use a Trustpoint to Secure Web Authentication on Catalyst 9800

Web authentication is a Layer 3 authentication method available on the Catalyst 9800 controller and widely used for guest access configuration. Web authentication allows users to get authenticated through a web browser on a wireless client, with minimal configuration on the client side.

Depending on the web policy configured on the controller, your guest user completes the authentication either with a username and password, or by entering the email address or by getting redirected to a particular web page before getting access to the guest portal. This portal is hosted on the controller internally as a predefined page, or hosted on the controller as a customized page, or hosted on an external server.

Many a times these guest WLANs are open with no Layer 2 authentication, hence we need to provide a means to protect the user credentials between the controller and the guest client.

Default behaviour

When guest users try to access the web policy page on a Windows system using the browser, they receive a security warning because of the standard self-signed certificate that is installed on the controller by default. In some other cases, systems that use the automatic web portal detection system, display a pop-up with the login page, once they detect that a user is connected to a web authentication network.

To avoid this warning for guest users, we recommend you deploy a third-party certificate signed by a trusted certificate authority because disabling the https encryption might compromise the security.

Additionally, for clients to trust the web authentication certificate, we recommend that you define a hostname that matches the Common Name (CN) in the certificate. This is possible both from the controller's CLI or the GUI.

You can configure it through the global parameter setting for webauth, where you can define the hostname for the Virtual IP address being used for web authentication. For instructions on how to configure it, refer List item. in *Recommendations and Limitations.*

Use a Trustpoint to Secure AP Join and Configure Mobility Tunnel on Catalyst 9800

Access Point (AP) Join

The controller and access points use CAPWAP and DTLS protocols to manage and encrypt data exchange with one another. The management interface on the controller handles the communication between these two entities.

Default behaviour

All appliance controller platforms (Catalyst 9800-40, 9800-80, 9800-L) and access points are shipped with a Cisco-installed Manufacturing Installed Certificate (MIC) device certificate. Additionally, controllers and access points have a Cisco Manufacturing Certificate Authority (CA) certificate that is used to sign and validate device certificates.

After an access point discovers a controller, it needs to join the controller. An AP can join a controller only after the controller and the AP verify each other's identity as part of the DTLS handshake.

A variation of the MIC on the Catalyst 9800-CL (the virtual controller) is a self-signed certificate, as the virtual controller does not have a MIC. The virtual controller relies on the self-signed certificate that has to be generated by the Day 0 wizard when you select it, once the wireless management interface has been enabled and the country configuration has been setup for AP Join. You can also automate this with a script. Refer to the Workflow to Configure a Trustpoint for a Self-signed Certificate on Catalyst 9800-CL, on page 49 to know how to set up a trustpoint on the Catalyst 9800-CL for AP Join.

Optionally, APs can use Locally Significant Certificates (LSC) to prove their identity. LSCs are created by an enterprise PKI managed by your company and are installed on the controller and the AP to provide more granular control. By default, LSC certificates are not installed on the controller and APs.

Mobility tunnel

Mobility Tunnel is a secure link between two controllers where data is encrypted and exchanged using CAPWAP and DTLS. When you configure a peer controller, the MIC certificate is used to create the tunnel. However, in case of a Catalyst 9800-CL, in the absence of a MIC, the self-signed certificate is used to configure a peer controller and will require you to add the self-signed certificate hash when configuring the mobility group.

Use a Trustpoint for Secure Connection between Catalyst 9800 and Cisco CMX

Cisco Connected Mobile Experiences (CMX) is a software solution that uses location and other intelligence from Cisco wireless infrastructure to generate analytics and deliver relevant services to customers on their mobile devices. Cisco CMX allows client authentication through the custom portal. This is similar to configuring web authentication where clients are redirected to the customized portal hosted on CMX.

Cisco CMX communicates with the Catalyst 9800 wireless controller using the Network Mobility Services Protocol (NMSP), which runs over a connection-oriented (TLS) transport. This transport provides a secure 2-way connectivity and is convenient when both the controller and CMX are on- premise and there is direct IP connectivity between them.

The controller verifies the peer and the host based on the certificate that is sent by the CMX when a connection is established. However, Root CAs are not preinstalled on the controller.

Use a Trustpoint to Secure Connection between Catalyst 9800 and Cisco DNA Center

The Cisco DNA Center can be used to deploy a wireless network using a mix of Catalyst 9800 and Catalyst 9800-CL devices.

When the DNA Center discovers the controller, a pre-defined trustpoint is pushed to the controller.

```
sdn-network-infra-iwan
```

This installs the DNA Center certificate on the controller and issues a certificate for Assurance. You can check the status using the command below.

```
show crypto pki certificates verbose sdn-network-infra-iwan
```

```
show crypto pki trustpoint sdn-network-infra-iwan status
```

Use a Trustpoint to Secure Connection between Catalyst 9800 and Cisco DNA Spaces

You can connect the Catalyst 9800 Series controller to Cisco DNA Spaces using the WLC Direct Connect option. However, the controller must have a third-party certificate that is used for identity validation when the controller tries to connect to Cisco DNA Spaces using the WLC Direct Connect option. Add the certificate using the following commands:

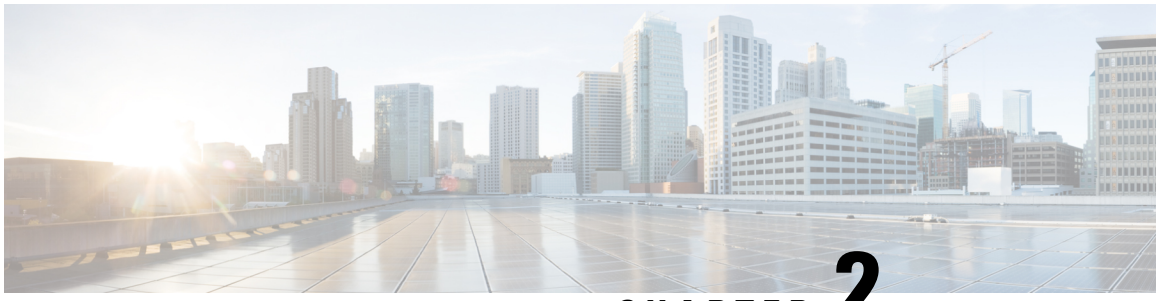
```
Device#conf t
Enter configuration commands, one per line. End with CNTL/Z.
Device(config)#ip name-server<DNS ip>
Device(config)#ip domain-lookup
Device(config)#crypto pki trustpool import url https://www.cisco.com/security/pki/trs/ios.p7b

Reading file from http://www.cisco.com/security/pki/trs/ios.p7b
Loading http://www.cisco.com/security/pki/trs/ios.p7b !!!
% PEM files import succeeded.
```

Use a Trustpoint for Local EAP Authentication on Catalyst 9800

Local EAP is an authentication method that allows users and wireless clients to be authenticated locally on the controller. It is designed for use in remote offices that want to maintain connectivity to wireless clients when the backend system becomes disrupted or the external authentication server goes down. When you enable local EAP, the controller serves as the authentication server and the local user database, so it removes dependence on an external authentication server. Local EAP retrieves user credentials from the local user database or the LDAP backend database to authenticate users. Local EAP supports LEAP, EAP-FAST, EAP-TLS, P EAPv0/MSCHAPv2, and PEAPv1/GTC authentication between the Catalyst 9800 controller and wireless clients.

Local EAP authentication requires the controller to set up a trustpoint as the controller needs to send the certificate for the client. Since clients do not trust the controller's default certificate, you will need to install a certificate trustpoint on the Catalyst 9800 controller that the client will trust (or import it manually in the client trust store).



CHAPTER 2

Supported Releases

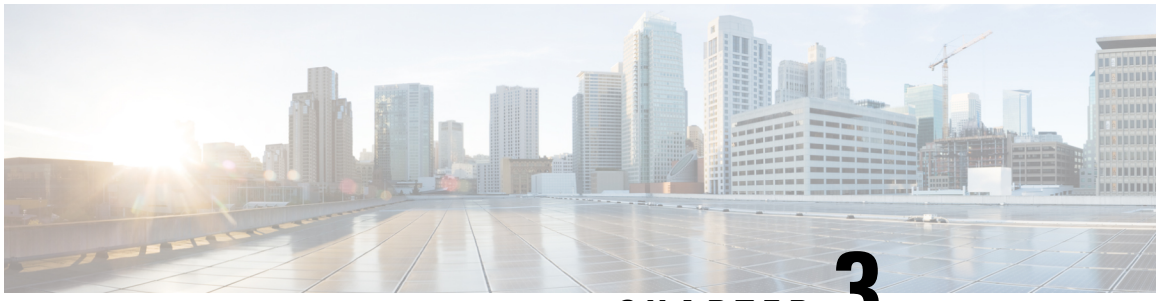
The information in this document is based on the following software versions. Unless specifically noted in the table below, the features and commands for Trustpoint configuration are available on the following software versions:

17.3.x, 17.4.x and 17.5.x

Table 1: Trustpoint/Certificate Features and Releases

Feature Name	Feature Description	Introductory Release
Enhanced Certificate Management Through GUI	The Public Key Infrastructure (PKI) Management page now displays the following tabs: **Trustpoints** tab: Used to add, create or enroll a new trustpoint. This also displays the current Trustpoints configured on the controller and other details of the trustpoint. You can also see if the trustpoint is in use for any of the features. **CA Server** tab: Used to enable or disable the Certificate Authority (CA) server functionality on the controller. The CA server functionality should be enabled for the controller to generate a Self-Signed Certificate (SSC). **Key Pair Generation** tab: Used to generate key pairs. **Certificate Management** tab: Used to generate and manage certificates, and perform all certificate related operations, on the controller.	Cisco IOS XE Amsterdam 17.3.1

Feature Name	Feature Description	Introductory Release
Enhanced Certificate Management Through GUI	New tab **Trustpool**	Cisco IOS XE Bengaluru 17.5.1
Intermediate CA support for AP authentication	The following commands have been introduced to import the complete certificate chain to the Trustpool in case the LSC certificate has been obtained from an intermediary CA. This is applicable for all other multi- tier certificates as well. **crypto pki trustpool import**	Cisco IOS XE Bengaluru 17.5.1
Support for both MIC and LSC APs to join the same C9800	The following commands have been introduced to configure AP authorization policy. • **ap auth-list ap-cert-policy allow-mic-ap** • **ap auth-list ap-cert-policy allow-mic-ap trustpoint <trustpoint-name>** • **ap auth-list ap-cert-policy{mac-address AP-EthenetMAC-address | serial number AP serial-number}policy-type mic** • **show ap auth-list ap-cert-policy**	Cisco IOS XE Bengaluru 17.5.1



CHAPTER 3

Recommendations and Limitations

This section outlines best practices, provides recommendations and also lists the limitations for certificate usage and trustpoint configuration.

General Guidelines

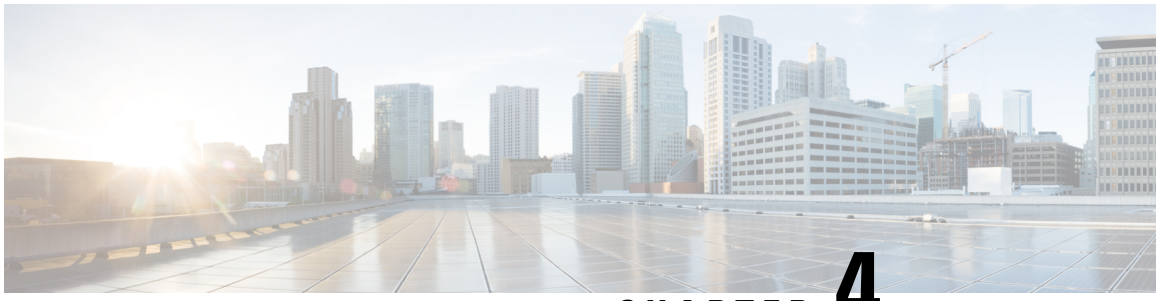
- For the C9800-CL in a public cloud, it is mandatory to use a Layer 3 port for wireless management. This interface is used to communicate with APs and other peer controllers to create Mobility Tunnels.
- Enable the HTTP server. This is required for secure web access and for Simple Certificate Enrollment Protocol (SCEP).
- Synchronize device clocks to a single NTP server for certificate validity.
- Assign trustpoints to the application. Use the WebUI to assign trustpoints as it is much simpler than using the CLI. Refer to Assign a Trustpoint for a Specific Service using the WebUI, on page 44 .
- Configure a hostname and a domain-name on the Cisco 9800. When you enroll for a certificate, the default subject-name is derived from the hostname.
- Catalyst 9800 supports using wildcard domain names in the CN field of the certificate.
- For clients to trust the web authentication certificate, define a hostname that matches the Common Name (CN) in the certificate. You can configure it through the global parameter setting for webauth, where you can define the hostname for the Virtual IP address being used for web authentication.

```
9800-L # configure terminal
9800-L (config)#parameter-map type webauth global
9800-L (config-params-parameter-map)#type webauth
9800-L (config-params-parameter-map)#timeout init-state sec 1234
9800-L (config-params-parameter-map)#virtual-ip ipv4 192.0.2.1 virtual-host
webauth.mywlc.mydomain.com
```

To ensure that the certificate is trusted by the web browser make a note of the following:

- The Common Name (or a SAN field) must match the URL visited by the browser. Since SAN configuration is not supported in the certificate signing request (CSR), in 17.3. x versions either from the CLI or GUI, you can use OpenSSL to generate a certificate signing request (CSR) containing the SAN fields.
- The certificate should be within its validity period. Note that some browsers now do not trust certificates with a validity period of more than one year, particularly for certificates that pertain to client web browser, i.e. web admin and webauth.

- The certificate must be issued by a CA or chain of CA whose root is trusted by the browser. For this, the certificate provided by the web server must contain all the certificates of the chain until (not necessarily included) a certificate trusted by the client browser (typically the Root CA).



CHAPTER 4

Configure a Trustpoint on Catalyst 9800

Trustpoints, as explained earlier, are abstract containers that include the identity certificate of the CA that signed the device certificate, CA-specific trustpoint configuration parameters, and an association with an enrolled identity certificate. Depending on the configuration, these certificates can be the default (self-signed certificate) controller certificate or can also be a third-party certificate. The default workflow for configuring a trustpoint is outlined below. Based on the certificate type being used, you may or may not have to go through all the steps.

- Workflow to configure a Trustpoint for a IOS XE device self-signed/local certificate on the controller

Note A trustpoint for a self-signed certificate does not require any explicit configuration. When you enable the HTTPS server, it generates a self-signed certificate automatically using default values. This has been noted here to acquaint you with the types of trustpoints available on the Catalyst 9800 controller.

- Workflow to Configure a Trustpoint for a Third-party Certificate on Catalyst 9800, on page 13

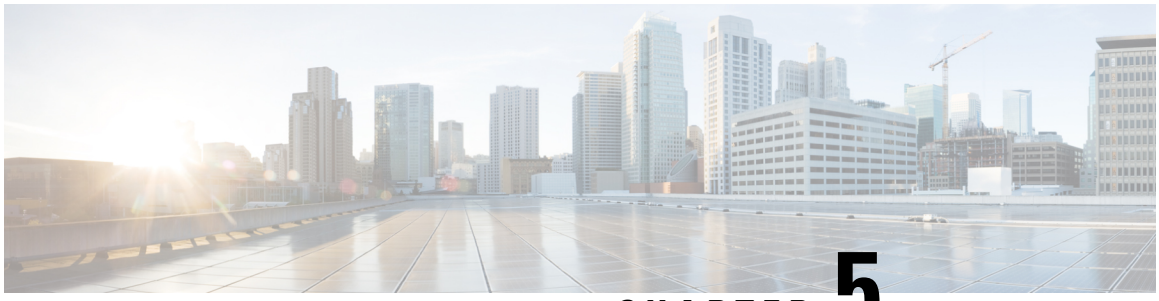
  Most of the steps outlined below are for configuring a third-party certificate that can be used for webadmin, web authentication, local eap authentication and AP join using locally significant certificates.

- Workflow to Configure a Trustpoint for a Self-signed Certificate on Catalyst 9800-CL, on page 49

  This configuration is for the virtual controller that needs a self-signed certificate for AP Join.

	Task	Purpose
Step1	Create an RSA key for the trustpoint.	An RSA key pair consists of a public key and a private key. The public key must be included in the certificate enrollment request. After the certificate is granted to the controller by your Certificate Authority (CA), the public key is included in the certificate so that peers can use it to encrypt data that is sent (back) to the controller. The private key is kept on the controller and used to decrypt the data sent by peers and to digitally sign transactions when negotiating with peers.

	Task	Purpose
Step 2	Create a trustpoint.	This is a container that corresponds to the CA from which the controller needs to receive a certificate. This container holds the identity and intermediate/CA certificate along with the keys. It is important to associate the key pair generated above with a trustpoint, to get the certificates for the device from the trustpoint. Declares the trustpoint and a given name and enters ca-trustpoint configuration mode. `Device(config-ca)#crypto pki trustpoint name`
Step 3	Authenticate the trustpoint.	The certificate of the CA must be authenticated before the device can be issued its own certificate and before certificate enrollment can occur. Authentication of the CA typically occurs only when you initially configure PKI support at your controller. To authenticate the CA, issue the following command in configuration mode, which authenticates the CA to your controller by obtaining the certificate of the issuer CA that contains the public key of the CA. `Device(config)#crypto pki authenticate trustpoint name`
Step 4	Enroll the trustpoint.	Specifies the method to obtain a certificate from a certification authority (CA); occurs between the end host (controller) that requests the certificate and the CA. The controller can request a certificate from the CA, either manually or automatically. Depending on your enrollment method you can either receive the certificate directly in this step or you will have to generate a CSR and send it to your CA for signing. After you have received the signed certificate, from the CA, install the certificate.
Step 5	Assign the trustpoint to a particular service.	Points the service to use the appropriate certificate.



CHAPTER 5

Workflow to Configure a Trustpoint for a Third-party Certificate on Catalyst 9800

Third-party certificates on the Catalyst 9800 controller can be used for any of the use cases discussed above. Creating a trustpoint and the RSA key pair starts the process of requesting a certificate from the CA server. The name of the trustpoint, the public RSA key pair of the host and additional details like the subject name, domain name are bundled in the certificate request, thereby binding them together. Catalyst 9800 supports EC key pair as well, but this document focuses on the RSA key pair only.

There are many ways to enroll your trustpoint and receive a certificate from the CA. Depending on the configuration, you can:

- Enroll the Trustpoint automatically.

  The Catalyst 9800 controller supports automatic certificate enrollment protocols like Simple Certificate Enrollment Protocol (SCEP) and Enrollment over Secure Tunnel (EST) to forward and receive certificate requests generated on the controller to the CA.

- Enroll the Trustpoint manually.

  The Catalyst 9800 controller supports manual enrollment that uses the PKCS#12 Certificate Signing Request (CSR) mechanism to issue certificates for the controller. Subsequent to the CSR request, the signed certificate for the controller, together with the CA root certificate, are uploaded to the controller. Note that it is also possible to use OpenSSL or any other utility to generate the keys and the CSR.

After the request is approved, the CA signs the request with its private key and returns the completed certificate to the controller. The controller writes the certificate to a storage area such as NVRAM and uses it to communicate with other devices.

Configuration is possible both from the controller's CLI or from the controller's WebUI. You can use the method that suits you better.

What to do next

Configuration Using the Catalyst 9800 CLI, on page 14	Step 1	Create RSA Key
	Step 2	Create Trustpoint
	Step 3	Authenticate and enroll trustpoint manually OR Authenticate and enroll trustpoint automatically
	Step 4	Assign Trustpoint
Configuration Using the Catalyst 9800 WebUI, on page 37	Step 1	Create RSA Key
	Step 2	Create, authenticate and enroll trustpoint manually OR Create, authenticate and enroll trustpoint automatically
	Step 3	Assign Trustpoint

- Configuration Using the Catalyst 9800 CLI, on page 14
- Configuration Using the Catalyst 9800 WebUI, on page 37

Configuration Using the Catalyst 9800 CLI

The following steps show how to generate an RSA key, configure a trustpoint, request a certificate from an external Certificate Authority using manual enrollment or automatic enrollment and finally use the trustpoint for a particular service.

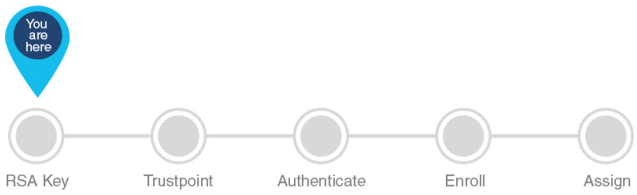
The progress bar is a visual indication of the steps that you are expected to complete in the module before you move on to the next configuration step.

What to do next

Create an RSA Key Pair using the CLI, on page 14

Create an RSA Key Pair using the CLI

Keys in a PKI system are used to encrypt and decrypt data. A key pair (a public and a private key) is required before you can obtain a certificate for your controller. The end host (here the controller) must generate a pair of keys and exchange the public key with the certification authority (CA) to obtain a certificate and enroll in a PKI. To generate key pairs, perform the following procedure on the controller's CLI:



Before you begin

Ensure that you have an understanding of the PKI framework.

Step 1 **enable**

Example:

```
Device>enable
```

Enables privileged EXEC mode. Enter your password, if prompted.

Step 2 **configure terminal**

Example:

```
Device#configure terminal
```

Enters global configuration mode.

Step 3 **crypto key generate rsa**

Example:

```
Device(config)#crypto key generate rsa
```

Generates one, general-purpose RSA key pair. The default key modulus is 1024. To specify other modulus sizes, use the **modulus** keyword.

Step 4 **crypto key generate rsa label** *key-pair-label*

Example:

```
Device(config)#crypto key generate rsa general-keys label ewlc-keys
```

(Optional) Assigns a label to each key pair. The label is referenced by the trustpoint that uses the key pair. Hence, we recommend that you use the same name for both key pair and trustpoint. If you do not assign a label, the key pair is automatically labeled, *Default-RSA-Key*.

Here we have named the key, *ewlc-keys*.

Step 5 **exit**

Example:

```
Device(config)#exit
```

Exits global configuration mode and returns to privileged EXEC mode.

Step 6 **show crypto key mypubkey rsa** *name of key*

Example:

```
Device#show crypto key mypubkey rsa ewlc-keys
```

(Optional) Displays the RSA public keys of your controller.

Verifies key pairs that you have generated.

Step 7 **write memory**

Example:

```
Device#write memory
```

Saves the keypair you have generated into secure storage.

This concludes the successful creation of an RSA keypair.

What to do next

Create a Trustpoint using the CLI, on page 16

Create a Trustpoint using the CLI

Trustpoints help to manage and track CAs and certificates that are used by the different services on the controller. Trustpoints work with RSA key pairs, hence we recommend that you use the same name for the key pair and trustpoint during configuration. To configure a trustpoint, perform the following steps:

Before you begin

Ensure that you have created a RSA keypair to be associated with the trustpoint.

Step 1 **enable**

Example:

```
Device>enable
```

Enables privileged EXEC mode. Enter your password, if prompted.

Step 2 **configure terminal**

Example:

```
Device#configure terminal
```

Enters global configuration mode.

Step 3 **crypto pki trustpoint** *trustpoint-name*

Example:

```
Device(config)#crypto pki trustpoint ewlc-cert
```

Creates a trustpoint that corresponds to the CA from which the controller needs to receive a certificate. Enters the crypto ca trustpoint configuration mode, which controls CA-specific trustpoint parameters that you will start configuring.

Step 4 Do one of the following:

- **enrollment url** *url*

```
Device(config-ca-trustpoint)#enrollment url http://<CA server IP>/certsrv/mscep/mscep.dll
```

```
Device(ca-trustpoint)#enrollment url http://10.29.67.142:80/certsrv/mscep/mscep.dll
```

Requests automatic enrollment using SCEP with the specified trustpoint and configures the enrollment URL.

- **enrollment terminal**

```
Device(config-ca-trustpoint)#enrollment terminal
```

Requests manual enrollment with the specified trustpoint by pasting the certificate received from the CA into the terminal.

Step 5 **subject-name** *subject_name*

Example:

```
Device(config-ca-trustpoint)#subject-name C=MX, ST=Nuevo Leon, L=Guadalupe, O=lab-wireless,
OU=mex-wireless, CN=public-guest.lab-kcg.com
```

Creates subject name parameters for the trustpoint.

Table 2: Subject Name Parameters

Field	Description
Domain Name/Common Name	Refers to the subject to which the certificate will be issued to. The fully qualified domain name (FQDN) of the controller. This must match exactly what you type in your web browser to reach the controller, or you will receive a name mismatch error. Depending on what your certificate requirement is for (webauth, webadmin, AP join) You must specify either the virtual IP address of your 9800 controller, the hostname associated with the virtual IP address of your 9800 controller, the management IP address or the hostname associated with the management IP address.
Country Code	The two-letter ISO code for the country where your organization is located.
State	The state/region where your organization is located. This shouldn't be abbreviated.
Location	The place where your organization is located.
Organisation	The legal name of your organization. This should not be abbreviated and should include suffixes such as Inc, Corp, or LLC.
Email Address	An email address used to contact your organization.

Step 6 **rsakeypair** *RSA_key*

Example:

```
Device(ca-trustpoint)#rsakeypair ewlc-keys
```

Maps RSA key with that of the trustpoint.

Step 7 **revocation**{**crl** | **none** | **ocsp**}

Example:

```
Device(ca-trustpoint)#revocation none
```

Sets one or more methods for revocation checking: CRL, OCSP, and none.

Step 8 **exit**

Example:

```
Device(ca-trustpoint)#exit
```

Exits global configuration mode and returns to privileged EXEC mode.

This concludes the successful creation of an RSA keypair and a trustpoint.

What to do next

Authenticate and Enroll the Trustpoint using the CLI , on page 18

Authenticate and Enroll the Trustpoint using the CLI

Certificate enrollment, which is the process of obtaining a certificate from a certification authority (CA), occurs between the end host that requests the certificate and the CA. Each peer that participates in the public key infrastructure (PKI) must enroll with a CA. You can choose to enroll the trustpoint manually or automatically. Select from the options below.

Authenticate and Enroll a Trustpoint Manually using the CLI , on page 18

OR

Authenticate and Enroll a Trustpoint Automatically using the CLI, on page 22

Authenticate and Enroll a Trustpoint Manually using the CLI

Manual certificate enrollment can be set up via TFTP or the manual cut-and-paste method. Both options can be used if your CA does not support SCEP or if a network connection between the controller and CA is not possible.

This configuration shows how to manually enroll, obtain and install the CA server certificate and the controller's device certificate. It uses an existing enterprise CA (Windows Certificate Server 2012) and does not cover the steps to set up a Windows Certificate Server CA from scratch. This procedure involves the following.

- Authenticate the trustpoint - Obtain and accept issuer-certificate of CA-server used to sign the device certificate.
- Enroll the trustpoint - Obtain the signed device certificate from the Certificate Authority by creating a Certificate Signing Request (CSR) and submitting the CSR to the CA.

- Install the certificate - Load the cetificate into the Wireless LAN Controller.

To authenticate , enroll and install the trustpoint manually using the cut-and-paste method, perform the following procedure on the controller:

Before you begin

Before you authenticate and enroll a trustpoint you should:

- have created an RSA key pair and a trustpoint and specified the enrollment method as manual by issuing the command **Device(config-ca-trustpoint)# enrollment terminal pem** . See step 4 of Create a Trustpoint using the CLI, on page 16 to configure this.
- understand the certificate extensions and procedure to convert it to a format, acceptable to the controller.
- understand the transport type that you will use to import the certificate or certificate chain from your CA in case the keys and Certificate Signing Request were generated outside the controller.

Step 1 Go to your enterprise CA page in the browser usually (*https://<CA-ip>/certserv*). Authenticate as administrator and click **Download a CA certificate**, **Certificate Chain** or **CRL**.

Step 2 In the **Encoding Method**, click the **Base 64 encoded radio** button and click **Download CA Certificate**.

Step 3 Copy the Base 64 encoded CA certificate contents into a notepad.

Step 4 Log into the controller's CLI either by SSH or Telnet and enter the following commands to import the CA certificate to the controller.

a) enable

Example:

```
Device>enable
```

Enables privileged EXEC mode. Enter your password if prompted.

b) **configure terminal**

Example:

```
Device#configure terminal
```

Enters global configuration mode.

c) **crypto pki authenticate** *trustpoint*

Example:

```
Device(config)#crypto pki authenticate ewlc-cert
Enter the base 64 encoded CA certificate.
End with a blank line or the word "quit" on a line by itself
-----BEGIN CERTIFICATE-----
*Issuing CA certificate*
-----END CERTIFICATE-----
```

```
Certificate has the following attributes:
Fingerprint MD5: DD05391A 05B62573 A38C18DD CDA2337C
Fingerprint SHA1: 596DD2DC 4BF26768 CFB14546 BC992C3F F1408809
% Do you accept this certificate? [yes/no]: yes
Trustpoint CA certificate accepted.
% Certificate successfully imported
```

This step authenticates the trustpoint and imports the Issuer CA certificate for the configured trustpoint. In other words, having the issuer certificate ensures that the controller trusts the CA that issues the device certificate. Ensure that the issuer certificate that will sign the controller's CSR is in .pem format. The certificate in this example is named **ewlc-cert** but you can chose the name you prefer, to differentiate between your certificates.

Note In case you have several levels of CAs, you must paste the issuing CA certificate here, i.e. the CA that issued your device certificate and only that one, not the chain. You will then need to create a trustpoint for each extra level of CA and repeat this step only for each of those trustpoints (i.e. authenticate a CA for each level).

d) **cypto pki enroll** *trustpoint*

Example:

```
Device(config)#crypto pki enroll ewlc-cert
% Start certificate enrollment ..
% The subject name in the certificate will include: C=MX, ST=Nuevo Leon, L=Guadalupe,
O=lab-wireless, OU=mex-wireless, CN=public-guest.lab-kcg.com
% The subject name in the certificate will include: 9800 WLC-karlcisn-Public.lab-kcg.com
% Include the router serial number in the subject name? [yes/no]: no
% Include an IP address in the subject name? [no]: no
Display Certificate Request to terminal? [yes/no]: yes
Certificate Request follows:
-----BEGIN CERTIFICATE REQUEST-----
*9800 WLC CSR*
-----END CERTIFICATE REQUEST-----
---End - This line not part of the certificate request--- Redisplay enrollment request? [yes/no]:
no
```

Generates a certificate for signing data and depending on the type of keys that you have configured, for encrypting data. The certificate request will be displayed on the console terminal so that it can be manually copied (or cut) to be sent to the CA.

Step 5 Send the full string of the certificate to the CA to get it signed.

Example:

```
-----BEGIN CERTIFICATE REQUEST-----
*9800 WLC CSR*
-----END CERTIFICATE REQUEST-----
```

Step 6 Go to your enterprise CA page in the browser usually (*https://<CA-ip>/certsrv*). Authenticate as administrator and click **Request a certificate**.

Step 7 Click the **Advanced Certificate Request** and enter the CSR details in the Certificate Template drop-down list, by selecting, **Web Server** and **Submit**.

Step 8 Click the **Base 64 encoded** radio button and download the certificate.

Note Ensure that your **Base 64 encoded** certificate is in .pem format. If it is in a different format, you will need to convert it to a format acceptable by the controller. See the procedure on how to convert to a different format in the Troubleshoot Common Issues for Certificate Configuration section of this guide.

Step 9 Log into the controller CLI either by SSH or Telnet and enter the following commands to import the device certificate that you received from your CA to the controller.

• If the keys and CSR were generated on the controller.

a) **crypto pki import** *trustpoint* **certificate**

Example:

```
Device(config)#crypto pki import ewlc-cert certificate
Enter the base 64 encoded certificate.
End with a blank line or the word "quit" on a line by itself
-----BEGIN CERTIFICATE-----
 *9800 WLC Signed Certificate*
 -----END CERTIFICATE-----
 quit
 % Router Certificate successfully imported
```

Import and install the signed device certificate that you got from your CA into the controller.

b) **show crypto pki certificates** *trustpoint-name*

Example:

```
Device# do sh crypto pki certificates ewlc-cert
Certificate
Status: Available
Certificate Serial Number (hex): 00A2020356CF31C818 Certificate Usage: General Purpose
Issuer:
cn=CA-KCG-lab
ou=lab-mex-wireless
o=mex-wireless
l=Guadalupe
st=Nuevo Leon
c=MX
Subject:
Name: *.lab-kcg.com
cn=*.lab-kcg.com
ou=lab-mex-wireless
o=mex-wireless
l=Benito Juarez
st=CDMX
c=MX
Validity Date:
start date: 17:14:54 UTC Feb 15 2018
end date: 17:14:54 UTC Mar 11 2023
Associated Trustpoints: ewlc-cert
Storage: nvram:CA-KCG-lab#C818.cer
```

Verifies that the enrollment process was successful by displaying certificate details issued for the controller and the CA certificate for the trustpoint.

• If the keys and CSR were generated outside the WLC

a) **crypto pki import** *trustpoint certificate-filename* **pkcs12** *tftp://TFTP-IP trustpoint certificate-filename* **password** *trustpoint-cert-password*

Example:

```
Device(config)#crypto pki import manual-tp pkcs12 tftp://9.7.44.186/xxx/9800_vwlc_ssc.pfx password
 cisco123
% Importing pkcs12...Reading file from tftp://9.7.44.186/xxx/9800_vwlc_ssc.pfx
Loading xxx/9800_vwlc_ssc.pfx from 9.7.44.186 (via Vlan39): !
[OK - 3709 bytes]

CRYPTO_PKI: Imported PKCS12 file successfully.
sangudla-wlc(config)#
Mar 19 11:04:11.925: %PKI-6-TRUSTPOINT_CREATE: Trustpoint: manual-tp created succesfully
Mar 19 11:04:11.926: %CRYPTO_ENGINE-5-KEY_ADDITION: A key named manual-tp has been generated or
```

```
imported by pki-pkcs12
Mar 19 11:04:11.935: %PKI-6-PKCS12_IMPORT_SUCCESS: PKCS #12 import in to trustpoint manual-tp
successfully imported.
```

Import the certificate using the TFTP server and install the signed device certificate that you got from your CA into the controller.

Note Ensure that there are no spaces after the password.

b) **show crypto pki certificate**

Example:

```
Device#do sh crypto pki certificate ewlc-cert
Certificate
Status: Available
Certificate Serial Number (hex): 00A2020356CF31C818 Certificate Usage: General Purpose
Issuer:
cn=CA-KCG-lab
ou=lab-mex-wireless
o=mex-wireless
l=Guadalupe
st=Nuevo Leon
c=MX
Subject:
Name: *.lab-kcg.com
cn=*.lab-kcg.com
ou=lab-mex-wireless
o=mex-wireless
l=Benito Juarez
st=CDMX
c=MX
Validity Date:
start date: 17:14:54 UTC Feb 15 2018
end date: 17:14:54 UTC Mar 11 2023
Associated Trustpoints: ewlc-cert
Storage: nvram:CA-KCG-lab#C818.cer
```

Verifies that the enrollment process was successful by displaying certificate details issued for the controller and the CA certificate for the trustpoint.

This concludes the successful authentication and subsequent enrollment of the trustpoint. It means that the certificate requested by the controller from the CA server is available and ready to be assigned to a specific service.

What to do next

Assign a Trustpoint for a Specific Service using the CLI, on page 31

Authenticate and Enroll a Trustpoint Automatically using the CLI

The following configuration shows how to request a certificate from an external Certificate Authority using automatic enrollment . It does not include the steps for setting up a Windows Server 2012 Standard R2, neither does it cover the steps for setting up the Simple Certificate Enrollment Protocol (SCEP) server. Refer to the

SCEP document listed in Additional References for Trustpoint Configuration on Catalyst 9800, on page 57 for specific configuration details.

With SCEP, the CA and device certificates are received from the CA server, and later installed automatically on the controller.

This procedure involves the following.

- Authenticate the trustpoint - Obtain and accept issuer-certificate of CA-server used to sign the device certificate.
- Enroll the trustpoint - Obtain the signed device certificate from the Certificate Authority over SCEP.
- Install the certificate - Load the cetificate into the Wireless LAN Controller.

You can use automatic enrollment for any certificate. In this example, we will specifically talk about Locally Significant Certificates that are used for AP Join. Once you receive the certificates, you will need to provision the AP with the certificate.

Note Since LSCs can be used for both AP Join and 802.1x port authorization, the AP Authorization state is by default set to use for CAPWAAP-DTLS sessions.

Note that this document does not talk about the additional configurations required if you want to use the LSC for 802.1x port authorization.

To authenticate, enroll and install the certificate automatically using the SCEP server perform the following procedure on the controller:

Before you begin

Before you authenticate and enroll a trustpoint you should:

- have an set up an external Certificate Authority and SCEP server and have a good understanding of these.
- have created a RSA key pair and a trustpoint and specified the enrollment method as automatic by issuing the command **Device(config-ca-trustpoint)#enrollment url http://<CA serverIP>/certsrv/mscep/mscep.dll**>. This means that certificates will be obtained from the specified Certificate Authority sever over SCEP. See step 4 of Create a Trustpoint using the CLI, on page 16

Step 1 **enable**

Example:

```
Device>enable
```

Enables privileged EXEC mode. Enter your password, if prompted.

Step 2 **configure terminal**

Example:

```
Device#configure terminal
```

Enters global configuration mode.

Step 3 **crypto pki authenticate** *trustpoint*

Example:

```
Device(config)#crypto pki authenticate ewlc-cert
  Certificate has the following attributes: Certificate has the following attributes:
 Fingerprint MD5: DD05391A 05B62573 A38C18DD CDA2337C
 Fingerprint SHA1: 596DD2DC 4BF26768 CFB14546 BC992C3F F1408809
 % Do you accept this certificate? [yes/no]: yes
 Trustpoint CA certificate accepted.
 % Certificate successfully imported
```

Authenticate the trustpoint. This step imports the CA certificate for the configured trustpoint to ensure that the controller trusts your CA.

Note This step assumes that you have already obtained a Base 64 encoded CA certificate from the CA represented by the trustpoint.

Step 4 **crypto pki enroll** *trustpoint*

Example:

```
Device(config)#crypto pki enroll ewlc-cert
%
 % Start certificate enrollment ..
 % Create a challenge password. You will need to verbally provide this
 password to the CA Administrator in order to revoke your certificate.
 For security reasons your password will not be saved in the configuration.
 Please make a note of it.
 Password:
 Re-enter password:
% The subject name in the certificate will include: C=MX, ST= Nuevo Leon, L= Guadalupe, O=
lab-wireless, CN=public-guest.lab-kcg.com, OU=mex-wireless
% The subject name in the certificate will include: 9800-L.xyz.local
% Include the router serial number in the subject name? [yes/no]: no
% Include an IP address in the subject name? [no]: no
Request certificate from CA? [yes/no]: yes
% Certificate request sent to Certificate Authority
```

Enrolls the controller with the trustpoint. Generates a certificate for signing data and depending on the type of keys that you have configured, for encrypting data.

To complete enrollment, obtain a certificate for the certificate request generated by the **crypto pki enroll** command from the CA represented by the applicable trustpoint.

Step 5 **exit**

Example:

```
Device(config)#exit
```

Exits global configuration mode and enters privileged EXEC mode.

Step 6 **write memory**

Example:

```
Device#write memory
```

Saves your entries in the configuration file.

Step 7 **show crypto pki certificates**

Example:

```
Device#show crypto pki certificates ewlc-cert
```

Verifies that the enrollment process was successful by displaying certificate details issued for the controller and the CA certificate for the trustpoint.

```
Certificate
 Status: Available
 Certificate Serial Number (hex): 00A2020356CF31C818
 Certificate Usage: General Purpose
 Issuer:
 cn=CA-KCG-lab
 ou=lab-mex-wireless
 o=mex-wireless
 l=Guadalupe
 st=Nuevo Leon
 c=MX
 Subject:
 Name: *.lab-kcg.com
 cn=*.lab-kcg.com
 ou=lab-mex-wireless
 o=mex-wireless
 l=Benito Juarez
 st=CDMX
 c=MX
 Validity Date:
 start date: 17:14:54 UTC Feb 15 2018
 end date: 17:14:54 UTC Mar 11 2023
 Associated Trustpoints: cert-name
 Storage: nvram:CA-KCG-lab#C818.cer
```

This concludes the successful authentication and subsequent enrollment of the trustpoint. In other words, it means that the certificate requested by the controller from the CA server is available and ready to be used by a specific service.

What to do next

If you are using the LSC certificate for AP Join, first provision the AP with the LSC. Refer to Provision Access Points with Locally Significant Certificates using the CLI, on page 27. Next assign the trustpoint for AP Join using LSC, refer to Assign a Trustpoint for AP Join with LSC using the CLI, on page 31.

OR

If you want to use the automatically obtained certificate for any other service, refer to Assign a Trustpoint for a Specific Service using the CLI, on page 31

Configure AP with MIC/SUDI to join Controller with LSC using the CLI

Starting from release 17.5, you can onboard an AP with a MIC/SUDI certificate to join a LSC deployed controller. Earlier, an AP with the default MIC/SUDI certificate would fail to join a controller whose wireless management trustpoint had been set to use an LSC. You would need to separately provision the AP with the LSC on a staging server before it could join the controller using the LSC. With release 17.5, the new authorization policy on the AP allows APs with MIC to join an LSC deployed controller.

To enable authorization on the AP's certificate policy perform the following task on the controller:

Step 1 **enable**

Example:

```
Device>enable
```

Enables privileged EXEC mode. Enter your password if prompted.

Step 2 **configure terminal**

Example:

```
Device#configure terminal
```

Enters global configuration mode.

Step 3 **ap auth-list ap-cert-policy allow-mic-ap**

Example:

```
Device(config)#ap auth-list ap-cert-policy allow-mic-ap
```

Enables the AP certificate policy which allows APs with MIC to join during a CAPWAP-DTLS handshake.

Step 4 **ap auth-list ap-cert-policy allow-mic-ap trustpoint** *trustpoint-name*

Example:

```
Device(config)# ap auth-list ap-cert-policy allow-mic-ap trustpoint trustpoint-name
```

Note The trustpoint configuration is required only for the virtual controller since it uses a self-signed certificate.

Configures the trustpoint name for the controller certificate chain. When APs join the virtual controller, they need to Device(config)# ap auth-list ap-cert-policy allow-mic-ap trustpoint trustpoint-name be aware of the trustpoint being used by the wireless management interface. In all other appliance controller platforms, the default MIC certificate will be selected. This default certificate is manufacturer installed SUDI.

Step 5 **ap auth-list ap-cert-policy** {**mac-address** *AP-Ethenet MAC-address* | **serial number** *AP serial-number*} **policy-type mic**

Example:

```
(config)#ap auth-list ap-cert-policy mac-address
1111.2222.3333 policy-type mic

(config)#ap auth-list ap-cert-policy serial-number
FGL2125ANSD policy-type mic
```

Configures the list of APs based on Ethernet MAC address or based on the assembly serial number of the AP, that should join using MIC.

Step 6 **exit**

Example:

```
Device(config)#exit
```

Exits global configuration mode and enters privileged EXEC mode.

Step 7 **show ap auth-list ap-cert-policy**

Example:

```
Device#show ap auth-list ap-cert-policy
Authorize APs joining with MIC : ENABLED
MIC AP policy trustpoint
Name : CISCO_IDEVID_SUDI
Certificate status : Available
Certificate Type : MIC
Certificate Hash : xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx
```

Verifies if the AP has been authorized to join using MIC.

Step 8 **show ap auth-list ap-cert-policy mac-address**

Example:

```
Device#show ap auth-list ap-cert-policy mac-address
MAC address AP cert policy
---------------------------------
00a2.891e.0fe8 MIC
00a3.8e4a.7632 MIC
1111.2222.3333 MIC
2c5a.0f70.84dc MIC
5c71.0d2e.a2cc MIC
f4db.e643.a0d6 MIC

Device#show ap auth-list ap-cert-policy serial-number
Serial number AP cert policy
-------------------------------
FGL2125ANSD MIC
```

Verifies if the AP has been selected for the authorization policy, based on its MAC address or serial number.

What to do next

If you want the AP to use the LSC, provision it using the steps in Provision Access Points with Locally Significant Certificates using the CLI, on page 27.

Provision Access Points with Locally Significant Certificates using the CLI

Other than Manufacturer Installed Certificate (MIC) or Secure Unique Device Identifier (SUDI) certificates, Access Points (AP) can also be provisioned with Locally Significant Certificates (LSC). For APs to be provisioned with LSCs, the controller acts as a proxy for the AP and any request to issue and sign the CA certificate is initiated by the controller. Once the controller receives the third-party certificates, they are pushed from the controller to the AP and next the APs are provisioned with the LSC.

Other than Manufacturer Installed Certificate (MIC) or Secure Unique Device Identifier (SUDI) certificates, Access Points (AP) can also be provisioned with Locally Significant Certificates (LSC). For APs to be provisioned with LSCs, the controller acts as a proxy for the AP and any request to issue and sign the CA certificate is initiated by the controller. Once the controller receives the third-party certificates, they are pushed from the controller to the AP and next the APs are provisioned with the LSC.

For LSC certificates that have been issued by an intermediate certificate authority:

- Ensure that you select the associated trustpoint and RSA key pair , created earlier while provisioning the AP.
- Ensure that you import the complete chain of CA certificates into the Trustpool using the command.

  ```
  Device(config)#crypto pki trustpool import
  ```

  The complete chain should be present on the controller, otherwise you will not be able to provision the AP. This step is not required, if the certificate has been issued by a root CA.

To provision the APs with the certificates perform the following task on the controller:

Before you begin

Before you start assigning the trustpoint for a specific service ensure that

- The trustpoint is valid.
- There is an RSA key pair.

Step 1 **enable**

Example:

```
Device>enable
```

Enables privileged EXEC mode. Enter your password, if prompted.

Step 2 **configure terminal**

Example:

```
Device#configure terminal
```

Enters global configuration mode.

Step 3 **ap lsc-provision subject-name-parameter**

Example:

```
Device(config)#ap lsc-provision subject-name-parameter country <country state <state> city <city>
domain <department> org <organization> email-address <mail address>
```

Configure subject name parameters for AP's device certificate.

Step 4 **ap lsc-provision join-attempt** *number of attempts*

Example:

```
Device(config)#ap lsc-provision join-attempt 10
```

Enter the number of unsuccesful join-attempts after which the AP uses the MIC to join the controller.

Step 5 **ap lsc-provision trustpoint** *trustpoint name*

Example:

```
Device(config)#ap lsc-provision trustpoint trustpoint-name
```

Selects the previously created trustpoint to be associated with this LSC.

Step 6 **ap lsc-provision key-size** *key size*

Example:

```
Device(config)#ap lsc-provision key-size 2048
```

Step 7 **ap lsc-provision mac-address**

Example:

```
Device(config)#ap lsc-provision mac-address 25-2e-65-43-eb-93
```

If LSC is required only on specific group of APs, configure an allowed list of AP mac-addresses.

Step 8 **ap lsc-provision**

Example:

```
Device(config)#ap lsc-provision
```

Enables LSC provisioning for all the APs joining the controller.

Step 9 **ap lsc-provision provision-list**

Example:

```
Device(config)#ap lsc-provision provision-list
```

Enables LSC provisioning for the allowed list of APs.

Step 10 **exit**

Example:

```
Device(config)#exit
```

Exits global configuration mode and enters privileged EXEC mode.

Step 11 **show ap lsc-provision summary**

Example:

```
Device#show ap lsc-provision summary
AP LSC-provisioning : Enabled for all APs
Trustpoint used for LSC-provisioning : AP-LSC
    Certificate chain status : Available
    Number of certs on chain : 2
    Certificate hash : 323b32f425e374f127d1e52541a5242b8f629e2a
LSC Revert Count in AP reboots : 4
AP LSC Parameters :
Country : MX
State : CDMX
City : Juarez
Orgn : Cisco TAC
Dept : Wireless TAC
Email : xyz@cisco.com
Key Size : 2048
EC Key Size : 384 bit

AP LSC-provision List :

Total number of APs in provision list: 2

Mac Addresses :
--------------
xxxx.xxxx.xxxx
xxxx.xxxx.xxxx
```

Verifies the details about the AP LSC provisioning configuration, along with the list of APs added to the provision list.

Step 12 **show crypto**

Example:

```
AP3802#show crypto

[...]

------------------------------------------------------------------------------
LSC: Enabled
----------------------------- Device Certificate -----------------------------
Certificate:
    Data:
        Version: 3 (0x2)
        Serial Number:
            73:00:00:00:0b:9e:c4:2e:6c:e1:54:84:96:00:00:00:00:00:0b
    Signature Algorithm: sha256WithRSAEncryption
        Issuer: DC=local, DC=chuu-domain, CN=CHUU-WIN12-CA
        Validity
            Not Before: May 13 01:22:13 2020 GMT
            Not After : May 13 01:22:13 2022 GMT
        Subject: C=MX, ST=CDMX, L=Juarez, O=Cisco TAC,
CN=ap3g3-286F7FCF53AC/emailAddress=josuvill@cisco.com
        Subject Public Key Info:
            Public Key Algorithm: rsaEncryption
                Public-Key: (2048 bit)

----------------------------- Root Certificate -------------------------------
Certificate:
    Data:
        Version: 3 (0x2)
        Serial Number:
            32:61:fb:93:a8:0a:4a:97:42:5b:5e:32:28:29:0d:32
    Signature Algorithm: sha256WithRSAEncryption
        Issuer: DC=local, DC=chuu-domain, CN=CHUU-WIN12-CA
        Validity
            Not Before: May 10 05:58:01 2019 GMT
            Not After : May 10 05:58:01 2024 GMT
        Subject: DC=local, DC=chuu-domain, CN=CHUU-WIN12-CA
        Subject Public Key Info:
            Public Key Algorithm: rsaEncryption
                Public-Key: (2048 bit)
```

Verifies the certificates installed in the AP from the AP CLI and ensures that both CA Root certificate and Device certificate are present

---

This concludes the successful authentication and subsequent enrollment of the trustpoint. It means that the certificate requested by the controller from the CA server is available and ready to be assigned to a specific service.

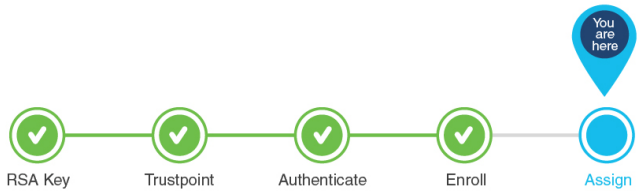
What to do next

Once the certificate is fully installed, the AP reboots, and starts the join process with the new certificate. Now that you have the LSC certificate installed on the AP, assign the certificate following the steps in Assign a Trustpoint for AP Join with LSC using the CLI, on page 31.

Assign a Trustpoint for a Specific Service using the CLI

Now that the trustpoint configuration is complete, how do you make use of the new certificates that have been created?

The following sections show how to assign the trustpoint to a specific service so that the right certificate is used for the right purpose. This step concludes your trustpoint configuration.



Assign a Trustpoint for AP Join with MIC or SSC using the CLI

The wireless management interface is used for AP Join. Note that both for physical controllers and for virtual controllers, no additional configuration is required to assign the trustpoint. The physical controller uses the default MIC or SUDI and the virtual controller uses the self-signed certificate.

However, if you have not generated the self-signed certificate for virtual controllers on Day 0, follow the procedure outlined in Workflow to Configure a Trustpoint for a Self-signed Certificate on Catalyst 9800-CL, on page 49.

This concludes the workflow of configuring a trustpoint.

What to do next

The above workflow should help you successfully configure a trustpoint. In case you have trustpoint configuration issues, refer to the resolutions to common problem scenarios listed in Troubleshoot Common Issues for Certificate Configuration , on page 51.

Assign a Trustpoint for AP Join with LSC using the CLI

When configured to use a Locally Significant Certificate (LSC), Access Points join the Controllers using an LSC. To set the wireless management trustpoint to use an LSC for AP Join, perform the following procedure:

Before you begin

- You should have configured a trustpoint for LSC and should have received a certificate from a third-party.
- The AP must have been provisioned with the LSC. For more information on how to do this refer to Provision Access Points with Locally Significant Certificates using the CLI, on page 27.

Procedure

	Command or Action	Purpose
Step 1	**enable** **Example:** `Device>enable`	Enables privileged EXEC mode. Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Device#configure terminal`	Enters global configuration mode.
Step 3	**wireless management trustpoint AP-LSC** **Example:** `Device(config)#wireless management trustpoint AP-LSC`	Assigns the LSC trustpoint for AP Join.
Step 4	**exit** **Example:** `Device(config)#exit`	Exits global configuration mode and returns to privileged EXEC mode.

Step 5	**show wireless management trustpoint**
	Example:
	`Device#show wireless management trustpoint`
	Example:
	`Device#show wireless management trustpoint`
		(Optional) Verifies that the wireless management is using the LSC trustpoint for AP Join.
		`show wireless management trustpoint`
		`Trustpoint Name : AP-LSC`
		`Certificate Info : Available`
		`Certificate Type : LSC`
		`Certificate Hash : feaa1751353f947f2311c0b7ab4c38206037bcd8`
		`Private key Info : Available`
		`FIPS suitability : Not Applicable`

This concludes the workflow of configuring a trustpoint.

Assign a Trustpoint for Web Authentication using the CLI

By default, web authentication uses the IOS XE device self-signed certificate to secure the connection between the user and the guest portal. If you want web authentication to use another certificate instead of the self-signed certificate, you must assign it through the web authentication parameter map.

Note Note that when you configure a trustpoint for web authentication purposes, the controller does not present the entire chain, but presents only the device and the CA certificate.

Before you begin

- Ensure that a certificate is installed on your controller.

Step 1 **enable**

Example:

```
Device>enable
```

Enables privileged EXEC mode. Enter your password if prompted.

Step 2 **configure terminal**

Example:

```
Device#configure terminal
```

Enters global configuration mode.

Step 3 **parameter-map type webauth global**

Example:

```
Device(config)#parameter-map type webauth global
```

Creates the parameter map.

Step 4 **trustpoint** *webauth cert*

Example:

```
Device(config-params-parameter-map)#trustpoint webauth-cert
```

Configures trustpoint for local web authentication.

Step 5 **virtual-ip ipv4** *ip-address* **virtual-host** *virtual hostname*

Example:

```
Device(config-params-parameter-map)#virtual-ip ipv4 192.0.2.1 virtual-host
test9800.eu-central-1.compute.internal
```

Ensures that the client trusts the web authentication certificate that has the matching hostname in the Common Name (CN) parameter of the certificate.

Step 6 **exit**

Example:

```
Device(config-params-parameter-map)#exit
```

Exits parameter configuration mode and returns to privileged EXEC mode.

Step 7 **show parameter-map type webauth global**

Example:

```
Device#sh parameter-map type webauth global
Parameter Map Name              : global
  Type                          : webauth
  Auth-proxy Init State time    : 120 sec
  Webauth max-http connection   : 100
  Webauth logout-window         : Enabled
  Webauth success-window        : Enabled
  Consent Email                 : Disabled
```

```
Sleeping-Client                 : Disabled
Webauth intercept https         : Disabled
Webauth Captive Bypass          : Disabled
Webauth bypass intercept ACL    :
Trustpoint name                 : webauth-cert
HTTP Port                       : 80
Watch-list:
  Enabled                       : no
Webauth login-auth-bypass:
```

Verifies that the WebAuth service is using the correct trustpoint.

This concludes the workflow of configuring a trustpoint.

Assign a Trustpoint for Webadmin using the CLI

By default, the HTTPS service uses the self-signed certificate generated by the controller's HTTPS server . If you want the HTTPS service to use a third-party certificate instead of the self-signed certificate, you must assign it using the CLI. Before assignning a new certificate, you must have completed the tasks mentioned below.

Note Note that when you configure a trustpoint for web admin purposes, the controller does not present the entire chain, but presents only the device and the CA certificate.

Before you begin

- Ensure that a certificate has been created for webadmin specifically and is saved.
- Ensure that the HTTP server has been restarted for this configuration to take effect.

Step 1 **enable**

Example:

```
Device>enable
```

Enables privileged EXEC mode. Enter your password if prompted.

Step 2 **configure terminal**

Example:

```
Device#configure terminal
```

Enters global configuration mode.

Step 3 **ip http secure-trustpoint** *trustpoint name*

Example:

```
Device(config)#ip http secure-server trustpoint ewlc-cert
```

Assigns the trustpoint to the HTTPS service.

Step 4 **show ip http server secure status**

Example:

```
Device#show ip http server secure status
HTTP secure server status: Enabled
HTTP secure server port: 443
HTTP secure server ciphersuite:
3des-ede-cbc-sha aes-128-cbc-sha
aes-256-cbc-sha dhe-aes-128-cbc-sha
ecdhe-rsa-3des-ede-cbc-sha
rsa-aes-cbc-sha2 rsa-aes-gcm-sha2
dhe-aes-cbc-sha2 dhe-aes-gcm-sha2
ecdhe-rsa-aes-cbc-sha2 ecdhe-rsa-aes-gcm-sha2
HTTP secure server TLS version: TLSv1.2
TLSv1.1 TLSv1.0
HTTP secure server client authentication:
Disabled
HTTP secure server trustpoint: ewlc-cert
HTTP
```

Verifies that the HTTPS service is using the correct trustpoint.

This concludes the workflow of configuring a trustpoint.

Assign a Trustpoint for Local EAP Authentication using the CLI

To assign a trustpoint for Local EAP authentication, perform the following procedure on the controller:

Before you begin

Ensure that the controller and the client each have their own device certifcate. They must also have a root certificate for the controller and a CA certificate for the client.

Procedure

	Command or Action	Purpose
Step 1	**enable** **Example:** `Device>enable`	Enables privileged EXEC mode. Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Device#configure terminal`	Enters global configuration mode.

	Command or Action	Purpose
Step 3	**eap profile** *profile-name* **Example:** `Device(config)#eap profile mylocapeap`	Configures an eap profile and specifies a profile name.
Step 4	**method peap** **Example:** `Device(config-eap-profile)#method peap`	Adds an allowed method for e.g. EAP-PEAP.
Step 5	**pki-trustpoint** *certificate name* **Example:** `Device(config-eap-profile)#pki-trustpoint admincert`	Sets the default pki trustpoint to be used for local eap authentication.
Step 6	**exit** **Example:** `Device(config)#exit`	Exits EAP configuration.

Step 7	**show run eap profiles**
	Example:
	`Device#show run eap profiles`
	`eap profile md5`
	`method md5`
	`end`
	
	`eap profile TLS`
	`method tls`
	`pki-trustpoint Self`
	`end`
	
	`eap profile MD5`
	`method md5`
	`end`
	
	`eap profile FAST`
	`method fast`
	`end`
		Shows the trustpoint configured for the EAP profile.

This completes the workflow for configuring a trustpoint.

What to do next

The above workflow should help you successfully configure a trustpoint. In case you have trustpoint configuration issues, refer to the resolutions to common problem scenarios listed in Troubleshoot Common Issues for Certificate Configuration , on page 51.

Configuration Using the Catalyst 9800 WebUI

The following steps show how to generate an RSA key, configure a trustpoint, request a certificate from an external Certificate Authority using manual enrollment or automatic enrollment and finally use the trustpoint for a particular service.

The progress bar is a visual indication of the steps that you are expected to complete in the module before you move on to the next configuration step.

What to do next

Create an RSA Key Pair using the GUI, on page 37

Create an RSA Key Pair using the GUI

Keys in a PKI system are used to encrypt and decrypt data. A key pair (a public and a private key) is required before you can obtain a certificate for your controller. The end host (here the controller) must generate a pair of keys and exchange the public key with the certification authority (CA) to obtain a certificate and enroll in a PKI. To generate a key pair, complete this task on the controller's GUI:

Before you begin

Ensure that you have an understanding of the PKI framework.

Step 1 Choose **Configuration** > **Security** > **PKI Management**.

Step 2 In the **Key Pair Generation** section, click **Add**.

a) Enter the **Key Name**. The label is referenced by the trustpoint that uses the key pair. If you do not assign a label, the key pair is automatically labeled, *Default-RSA-Key*.

b) In the **Key Type** options, select either **RSA Key** or **EC Key**. The default modulus size for the RSA key is 4096 and the default value for the EC key is 521.

c) In the **Modulus Size** field, enter the modulus value for the RSA key or the EC key.

d) Check the **Key Exportable** check box to export the key for backup or archive purposes. By default, this field is enabled.

e) Click **Generate**.

This successfully concludes the creation of a keypair.

What to do next

Create, Authenticate and Enroll a Trustpoint using the WebUI, on page 38

Create, Authenticate and Enroll a Trustpoint using the WebUI

Certificate enrollment, which is the process of obtaining a certificate from a certificate authority (CA), occurs between the end host that requests the certificate and the CA. Each peer that participates in the public key infrastructure (PKI) must enroll with a CA. You can choose to enroll the trustpoint manually or automatically after you have created a trustpoint. Select from the options below.

Create, Authenticate and Enroll a Trustpoint Manually using the WebUI, on page 38

OR

Create, Authenticate and Enroll a Trustpoint Automatically using the WebUI, on page 40

Create, Authenticate and Enroll a Trustpoint Manually using the WebUI

Trustpoints help to manage and track CAs and certificates that are used by the different services on the controller. Trustpoints work with RSA key pairs, hence we recommend that you use the same name for the key pair and trustpoint during configuration. A trustpoint needs to be declared to send certificate requests for the controller and also for obtaining the certificate authority's (CA) certificate. To create, authenticate and enroll a trustpoint manually perform this procedure on the controller's WebUI:

Step 1 Choose **Configuration** > **Security** > **PKI Management** > **Add Certificate**.

Step 2 Click **Generate Certificate Signing Request**.

a) In the **Certificate Name** field, enter the certificate name.

b) From the **Key Name** drop-down list, choose an RSA key pair. (Click the plus (+) icon under the **Key Pair Generation** tab to create new RSA key pairs.).

c) Enter values the **Country Code**, **Location**, **Organisation**, **State**, **Organizational Unit**, and the **Domain Name** fields.

Table 3: Subject Name Parameters

Field	Description
Domain Name/Common Name	The fully qualified domain name (FQDN) of the WLC server. This must match exactly what you type in your web browser to reach the controller, or you will receive a name mismatch error. Depending on what your certificate requirement is (for webauth,webadmin, AP join), you must specify either the virtual IP address of your 9800 controller, the hostname associated with the virtual IP address of your 9800 controller, the management IP address or the hostname associated with the management IP address.
Country Code	The two-letter ISO code for the country where your organization is located.
State	The state/region where your organization is located. This shouldn't be abbreviated.
Location	The place where your organization is located.
Organisation	The legal name of your organization. This should not be abbreviated and should include suffixes such as Inc, Corp, or LLC.
Email Address	An email address used to contact your organization.

d) Click **Generate**.

The generated Certificate Signing Request (CSR) is displayed on the right. Click **Copy** to copy and save a local copy. Click **Save to Device** to save the generated CSR to the /bootflash/csr directory.

Step 3 Click **Authenticate Issuer CA** .

a) From the **Trustpoint** drop-down list, choose the trustpoint label generated above, or any other trustpoint label that you want to authenticate.

b) In the **Issuer CA Certificate (.pem)** field, copy and paste the certificate of the issuer CA received in .pem format that signed the CSR.

Note This step assumes that you already have the PEM Base64 certificate of the issuing CA. Ensure that you copy and paste the PEM Base64 certificate of the issuing CA that signs the device certificate.

c) Click **Authenticate**.

Step 4 Depending on whether you received a PEM certificate or a PKCS12 certificate, select the import and install method.

- Import .PEM certificate for the device

a) From the **Trustpoint** drop-down list, choose the trustpoint label that was generated earlier for a particular service.

b) Copy the paste the certificate contents received from the CA and click **Import**.

- Import PKCS12 certificate for the device

a) From the **Transport Type** drop-down list, choose either **FTP**, **SFTP**, **TFTP**, **SCP**, or **Desktop (HTTPS)**.

For **FTP**, **SFTP**, and **SCP**, enter values in the **Server IP Address (IPv4/IPv6)**, **Username**, **Password**, **Certificate File Path**, **Certificate Destination File Name**, and **Certificate Password** fields. For **TFTP**, enter values in the **Server IP Address (IPv4/IPv6)**, **Certificate File Path**, **Certificate Destination File Name**, and **Certificate Password** fields.

For **Desktop (HTTPS)**, enter values in the **Source File Path** and **Certificate Password** fields.

b) Click **Import**.

This concludes the successful authentication and subsequent enrollment of the trustpoint. In other words, it means that the certificate requested by the controller from the CA server is available and ready to be used by a specific service.

If you have issues with certificates and formats, check Troubleshoot Common Issues for Certificate Configuration , on page 51 to find a solution to your problem.

What to do next

Assign a Trustpoint for a Specific Service using the WebUI, on page 44

Create, Authenticate and Enroll a Trustpoint Automatically using the WebUI

A trustpoint is an abstract container for an identity certifcate that can be used to secure communication between the client and the server. A trustpoint needs to be declared to send certificate requests for the controller and also for obtaining the certificate authority's (CA) certificate.

The following procedure shows how to request a certificate from an external Certificate Authority using automatic enrollment. It does not include the steps for setting up a Windows Server 2012 Standard R2, neither does it cover the steps for setting up the Simple Certificate Enrollment Protocol (SCEP) server. Refer to the SCEP document listed in Additional References for Trustpoint Configuration on Catalyst 9800, on page 57 for specific configuration details. With SCEP, the CA and device certificates are received from the CA server, and later installed automatically on the controller.

You can use automatic enrollment for any certificate. In this example, we will specifically talk about Locally Significant Certificates that are used for AP Join. Once you receive the certificates, you will need to provision the AP with the certificate.

Note Since LSCs can be used for both AP Join and 802.1x port authorization, the AP Authorization state is by default set to use for CAPWAAP-DTLS sessions.

Note that this document does not talk about the additional configurations required if you want to use the LSC for 802.1x port authorization.

To create, authenticate and enroll the trustpoint to request and receive a certificate from a third-party, complete this task on the controller's WebUI:s

Before you begin

You should have created a RSA key pair to be associated with the trustpoint.

Step 1 Choose **Configuration** > **Security** > **PKI Management**.

Step 2 In the **PKI Management** window, click the **Trustpoints** tab.

Step 3 In the **Add Trustpoint** dialog box, provide the following information:

a) In the **Label** field, enter a unique label for the trustpoint.

b) Select the **Enrollment Type** as **SCEP** and enter the enrollment URL in the **Enrollment URL** field, to automatically request and download a CA certificate from the CA server.

c) Check the **Authenticate** check box to authenticate the Trustpoint and get the CA server certificate.

d) In the **Subject Name** section, enter the **Country Code**, **State**, **Location**, **Organisation**, **Domain Name**, and **Email Address**.

Table 4: Subject Name Parameters

Field	Description
Domain Name/Common Name	The fully qualified domain name (FQDN) of the controller server. This must match exactly what you type in your web browser to reach the controller, or you will receive a name mismatch error. Depending on what your certificate requirement is (for webauth,webadmin, AP join), you must specify either the virtual IP address of your 9800 controller, the hostname associated with the virtual IP address of your 9800 controller, the management IP address or the hostname associated with the management IP address.
Country Code	The two-letter ISO code for the country where your organization is located.
State	The state/region where your organization is located. This shouldn't be abbreviated.
Location	The place where your organization is located.
Organisation	The legal name of your organization. This should not be abbreviated and should include suffixes such as Inc, Corp, or LLC.
Email Address	An email address used to contact your organization.

e) Check the **Key Generated** check box to view the available RSA keypairs. Choose an option from the **Available RSA Keypairs** drop-down list to associate the keypair with the trustpoint.

f) Check the **Enroll Trustpoint** check box to request the certificate for the controller from the Certificate Authority.

g) In the **Password** field, enter the password.

Also called the challenge password, this password must match the challenge password for your CA. In the Certificate Signing request template, you must enter the same challenge password that was configured for the SCEP server, otherwise the authentication between the controller and CA fails.

h) In the **Re-Enter Password** field, confirm the password.
i) Click **Apply to Device**.

The new trustpoint is added to the trustpoint name list.

This concludes the successful authentication and subsequent enrollment of the trustpoint. It means that the certificate requested by the controller from the CA server is available and ready to be assigned to a specific service.

What to do next

If you are using the LSC certificate for AP Join, first provision the AP with the LSC. Refer to Provision Access Points with Locally Significant Certificates, using the WebUI, on page 43. Next assign the trustpoint for AP Join using LSC, refer to Assign Trustpoint for AP Join with LSC using the WebUI, on page 45 .

OR

If you want to use the automatically obtained certificate for any other service, refer to the other services in Assign a Trustpoint for a Specific Service using the WebUI, on page 44 .

Configure AP with MIC/SUDI to join Controller with LSC on WebUI

Starting from release 17.5, you can onboard an AP with a MIC/SUDI certificate to join a LSC deployed controller. Earlier, an AP with the default MIC/SUDI certificate would fail to join a controller whose wireless management trustpoint had been set to use an LSC. You would need to separately provision the AP with the LSC on a staging server before it could join the controller using the LSC. With release 17.5, the new authorization policy on the AP allows MIC APs to join LSC deployed controller.

To enable sauthorization on the AP's certificate policy perform the following task on the controller:

Before you begin

Step 1 Configure the AP certificate policy by navigating to **Configuration** > **Wireless** > **Access Points** > **All Access Points** page and expand **AP Certificate Policy**.

a) Tap the **Authorize APs joining with MIC** toggle button to enable AP authorization.
b) Select the **Trustpoint** that should be used by the controller for AP Join. This configuration is required only for the virtual controllers that use a self-signed certificate for AP Join. In case of appliance controllers, the deafult is always the MIC/ SUDI certificate.

Step 2 Build a list of APs that should be allowed to join the controller with this configuration by adding to the **List of MAC Address and Serial Numbers**.

a) Configure the AP Authlist by selecting between **MAC Address** or **Serial Number** of the APs and enter the relevant details in the box below.
b) You can also upload a .csv file containing the above details. The AP Certificate Policy is added to the existing AP Inventory page.

Step 3 Click **Apply** to save the configuration.

What to do next

If you want the AP to use the LSC, provision it using the steps in Provision Access Points with Locally Significant Certificates, using the WebUI, on page 43.

Provision Access Points with Locally Significant Certificates, using the WebUI

Other than the Manufacturing Installed Certificate (MIC) or the Secure Unique Device Identifier (SUDI) certificates, Access Points can also be provisioned with Locally Significant Certificates (LSC). For APs to be provisioned with LSCs, the controller acts as a proxy for the AP and any request to issue and sign the CA certificate is initiated by the controller. Once the controller receives the third-party certificates, they are pushed from the controller to the AP and next the APs are provisioned with the LSC.

From release 17.5, for LSC certificates that have been issued by an intermediate certificate authority:

- ensure that you select the associated trustpoint and RSA key pair, created earlier while provisioning the AP.
- ensure that you import the complete chain of CA certificates into the Trustpool. Go to **Configuration** > **PKI Management** > **Trustpool** tab and use the **Import** button to import the CA certificate. The complete chain should be present on the controller, otherwise you will not be able to provision the AP. This step is not required, if the certificate has been issued by a Root CA.

To provision the APs with the certificates, perform the following task on the controller:

Before you begin

Step 1 Choose **Configuration** > **Wireless** > **Access Points** and expand the **LSC Provision** drop-down list.

Step 2 In the **Subject Name Parameters** section, enter the following details and click **Apply**.

Table 5: Subject Name Parameters

Field	Description
Domain Name/Common Name	The fully qualified domain name (FQDN) of the controller. This must match exactly what you type in your web browser to reach the WLC, or you will receive a name mismatch error.
Country Code	The two-letter ISO code for the country where your organization is location.
State	The state/region where your organization is located. This shouldn't be abbreviated.
Location	The place where your organization is located.
Organisation	The legal name of your organization. This should not be abbreviated and should include suffixes such as Inc, Corp, or LLC.
Email Address	An email address used to contact your organization.

Step 3 Enable LSC provisioning from the **Status** dropdown.

Step 4 Use the **Trustpoint Name** drop-down list to select the previously defined trustpoint associated with the LSC and the key associated with the trustpoint. Enter the retry attempts for the AP to join the controller. After the defined number of attempts, the AP will attempt to join using the MIC.

Step 5 If you want to trigger AP Provisioning using LSC, you can do so in the **Add APs to LSC Provision List** section, by selecting a CSV file containing the AP MAC address details and uploading it or by or addding specific APs defined in the MAC address list.

If you have selected the correct Trustpoint Name, then the **Certificate chain status** and **Number of certificates on chain** reflects the availability of the certificate along with the number of associated chain of certificates. If the status shows as **Not Available** then you must see if the entire chain has been imported or not. Depending on whether you enrolled the certificate automatically or manually, re-import the chain using the procedure in the respective sections.

Step 6 Click **Apply** to trigger AP LSC enrollment.

APs begin certificate request, download, and installation. Once the certificate is fully installed, the AP reboots, and starts the join process with the new certificate.

What to do next

Once the certificate is fully installed, the AP reboots, and starts the join process with the new certificate. Now that you have the LSC certificate installed on the AP, assign the certificate following the steps in Assign Trustpoint for AP Join with LSC using the WebUI, on page 45 .

Assign a Trustpoint for a Specific Service using the WebUI

Now that the trustpoint configuration is complete, how do you make use of the new certificates that have been created? The following sections show how to assign the trustpoint to a specific service so that the right certificate is used for the right purpose. This step concludes your trustpoint configuration.

Assign Trustpoint for AP Join with MIC or SSC using the WebUI

The wireless management interface is used for AP Join. In case the AP joins a physical controller, no additional configuration is required on the controller as it uses the default MIC/SUDI. The same is applicable for virtual controllers that have a self-signed certiificate.

However, for virtual controllers, if you have not generated the self-signed certificate on Day 0, the controller's management interface needs to be configured to use its self-signed certificate for AP Join. To do so, perform the following procedure on the controller's WebUI:

Before you begin

You should have a self-signed certificate for the virtual controller. This step is required only if you have not generated a certificate on Day 0. Follow the procedure to generate a self-signed certificate for the virtual

controller as outlined in Workflow to Configure a Trustpoint for a Self-signed Certificate on Catalyst 9800-CL, on page 49.

Step 1 On the **Configuration** > **Interface** > **Wireless** page, select the **VLAN Interface Name**.

Step 2 In the **Edit Management Interface** page, select the **Trustpoint** from the drop-down list. This is the self-signed certificate of the virtual controller.

Step 3 Click **Update & Apply to Device**.

This concludes the workflow of configuring a trustpoint.

What to do next

The above workflow should help you successfully configure a trustpoint. In case you have trustpoint configuration issues, refer to the resolutions to common problem scenarios listed in Troubleshoot Common Issues for Certificate Configuration , on page 51.

Assign Trustpoint for AP Join with LSC using the WebUI

Since the wireless management interface is used for AP Join, APs can join the controller using MIC as well as a third-party certificate also known as the Locally Significant Certificate (LSC). If the wireless management interface was previously configured to use the MIC and now you want the LSC to be used for AP Join, you will need to set the trustpoint on the wireless management interface to use the LSC. To do so, perform the following procedure on the controller's WebUI:

Before you begin

- You should have configured a trustpoint for LSC and should have received a certificate from a third-party.
- The AP must have been provisioned with the LSC. For more information on how to do this refer to Provision Access Points with Locally Significant Certificates, using the WebUI, on page 43

Step 1 On the **Configuration** > **Interface** > **Wireless** page, select the **VLAN Interface Name**.

Step 2 In the **Edit Management Interface** page, select the **Trustpoint** that represents the LSC, from the drop-down list.

Step 3 Click **Update & Apply to Device**.

This concludes the workflow of configuring a trustpoint.

What to do next

The above workflow should help you successfully configure a trustpoint. In case you have trustpoint configuration issues, refer to the resolutions to common problem scenarios listed in Troubleshoot Common Issues for Certificate Configuration , on page 51.

Assign Trustpoint for Web Admin using the WebUI

Point the HTTPS service to use the certificate for the web login portal. To do so, perform the following procedure on the controller's WebUI:

Step 1 On the **Administration** > **Management** > **HTTP/HTTPS/Netconf** page, tap to **Enable Trustpoint** under **HTTP Trust Point Configuration**.

Step 2 Select the trustpoint from the drop-down list that should be used for web admin authentication.

Step 3 Click **Apply**, for the configuration to take effect.

This completes the workflow of configuring a trustpoint.

What to do next

The above workflow should help you successfully configure a trustpoint. In case you have trustpoint configuration issues, refer to the resolutions to common problem scenarios listed in Troubleshoot Common Issues for Certificate Configuration , on page 51.

Assign Trustpoint for Web Authentication using the WebUI

By default, web authentication uses the IOS XE device self-signed certificate to secure the connection between the user and the guest portal. If you want web authentication to use another certificate instead of the self-signed certificate, you must assign it through the web authentication parameter map.

Note Note that when you configure a trustpoint for web authentication purposes, the controller does not present the entire chain, but presents only the device and the CA certificate.

Point the **Web Auth Parameter** to use the trustpoint for web authentication. To do so, perform the following procedure on the controller's WebUI:

Before you begin

Ensure that a certificate is installed on your controller.

Step 1 On the **Configuration** > **Security** > **Web Auth** page, select the **global** parameter.

Step 2 In the **Edit Web Auth Parameter** page, select the **Trustpoint** from the drop-down list that should be used for web authentication.

Step 3 Click **Update & Apply**.

This concludes the workflow of configuring a trustpoint.

What to do next

The above workflow should help you successfully configure a trustpoint. In case you have trustpoint configuration issues, refer to the resolutions to common problem scenarios listed in Troubleshoot Common Issues for Certificate Configuration , on page 51.

Assign Trustpoint for Local EAP Authentication

Point the EAP profile to use the trustpoint for local eap authentication. To do so, perform the following procedure on the controller's WebUI:

Before you begin

Ensure that the controller and the client each have their own device certifcate. They must also have a root certificate for the controller and a CA certificate for the client. Also,you must have a trustpoint configured for local EAP authentication.

Step 1 Go to the **Configuration** > **Security** > **Local EAP** > **Local EAP Profile** page, and select the profile.

Step 2 Select the trustpoint from the drop-down list that should be used for Local EAP Authentication.

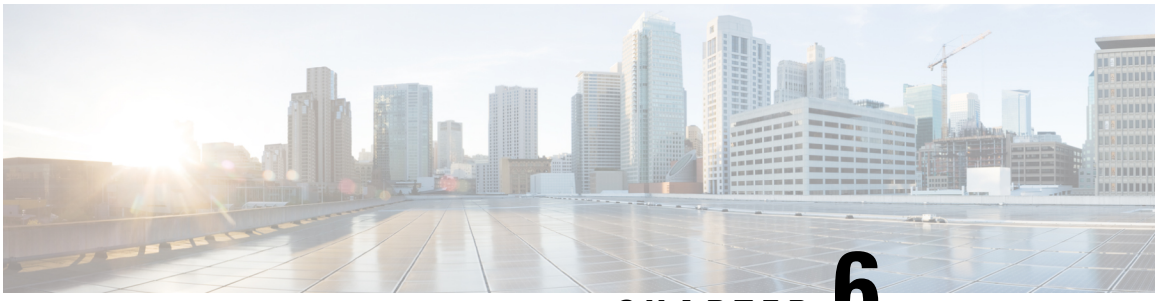
Step 3 Click **Apply**, for the configuration to take effect.

This concludes the workflow for configuring a trustpoint.

What to do next

The above workflow should help you successfully configure a trustpoint. To verify, you can go to **Configuration** > **Security** > **PKI Management** > **Trustpoints** tab to view the Trustpoint, its details and the related service using it.

In case you have trustpoint configuration issues, refer to the resolutions to common problem scenarios listed in Troubleshoot Common Issues for Certificate Configuration , on page 51.



CHAPTER 6

Workflow to Configure a Trustpoint for a Self-signed Certificate on Catalyst 9800-CL

The Catalyst 9800-CL or the virtual controller does not have a Manufacturing Installed Certificate (MIC). On Day 0, you have to explicitly generate a self-signed certificate, get it signed by your local IOS CA and import it using the Simple Certificate Enrollment Protocol (SCEP). Next, you need to map it to the wireless management interface, since the CAPWAP connection between the AP and controller uses the wireless management interface for authentication.

The configuration is possible using the CLI or Day 0 wizard on the WebUI of the virtual controller. However, we recommend that you use the script below to complete the configuration.

Before you begin

- Ensure that the VLAN interface is up and the IP is reachable.

```
Device#show ip interface brief
Interface              IP-Address      OK?    Method Status      Protocol
GigabitEthernet0/1     unassigned      YES    unset  up          up
GigabitEthernet0/2     unassigned      YES    NVRAM  administratively down up
VLAN1                  unassigned      YES    NVRAM  administratively down  up
VLAN56                 9.9.56.40       YES    NVRAM  up          up
```

```
Device#ping 9.9.56.40
 Type escape sequence to abort.
     Sending 5, 100-byte ICMP Echos to 9.9.56.40, timeout is 2 seconds:
     !!!!!
     Success rate is 100 percent (5/5), round-trip min/avg/max = 1/1/1 ms
```

- Enable the HTTP Server

  The certificate server supports Simple Certificate Enrollment Protocol (SCEP) over HTTP. The HTTP server must be enabled on the virtual controller for the certificate server to use SCEP. If the HTTP server is not enabled, only manual PKCS12 enrollment is supported.

  To enable the HTTP server, use the following command:

  Device(config)#**ip http server**

- Synchronize the clock

  Mark the hardware clock as authoritative using the following command:

  Device(config)#**clock calendar-valid**

Create a certificate for the AP to join the virtual controller. It can either be created automatically when you select the option on the Day 0 flow or by using a command.

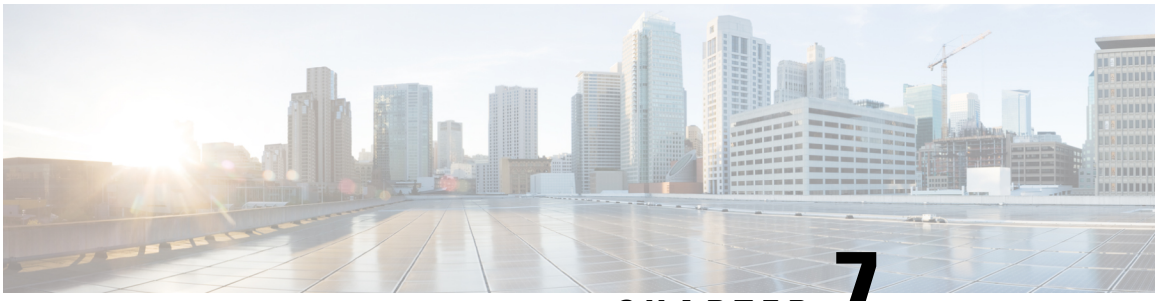
To configure the self-signed certificate, complete this command on the controller:

	Command	Purpose
Step 1	**enable** `Device>enable`	Enables privileged EXEC mode. Enter your password if prompted.
Step 2	**configure terminal** `Device#config t`	Enters global configuration mode.
Step 3	**wireless management interface vlan** *Vlan interface number* `Device(config)#wireless management interface vlan 122`	Specify the interface for the wireless management interface.
Step 4	**end** `Device#end`	Returns to privileged EXEC mode.
Step 5	**wireless config vwlc-ssc key-size** [**1024** | **2048** | **3072**] **signature-algo** [**sha1** | **sha256** | **sha384**]**password** [**0** | **7**] *password* `Device#wireless config vwlc-ssc key-size 2048 signature-algo sha256 password 0 cisco123`	Automates the creation of a self-signed certificate to be used for AP Join and assigns it to the Wireless Management Interface (WMI) automatically. **Note** This exec cli is not supported on native IPv6 deployments. We recommend that you use the Day 0 to generate a self-signed certificate for the Catalyst 9800-CL or manually configure the trustpoint.

Step 6	**show wireless management trustpoint**
	`Device#show wireless management trustpoint`
	`Trustpoint Name : ewlc-default-tp`
	`Certificate Info : Available`
	`Certificate Type : SSC`
	`Certificate Hash :`
	`e55e61b683181ff0999ef317bb5ec7950ab86c9e`
	`Private key Info : Available`
		Verifies the certificate installation.

This completes the trustpoint configuration for AP Join from the virtual controller.

In case you had skipped the Day 0 flow on the GUI for certificate/trustpoint configuration APs will not able to join. To configure this, on the virtual controller WebUI, go to **Configuration** > **Security** > **PKI Management**. In the **AP SSC Trustpoint** section and click **Generate** and enter the relevant details. For APs to join, map this trustpoint to the controller's wireless management interface. Refer to Assign Trustpoint for AP Join with MIC or SSC using the WebUI, on page 44 for further details.



CHAPTER 7

Troubleshoot Common Issues for Certificate Configuration

The following lists the common problems and resolution related to certificates. If your solution is not listed here, use the commands listed below to debug further.

Configuring the CA server "Failure Reason: Time has not been set. Cannot start the Certificate Server "

Possible Cause

Clock is not set on the controller.

Recommended Solution

Set the clock on 9800 using the command.

```
Device(config)#clock calendar-valid
```

Configuring the CA server ""Error in receiving Certificate Authority certificate"

Possible Cause

Lost connectivity to the managemet interface.

Recommended Solution:

Check if the management interface IP of the virtual controller is reachable.

Error message “This certificate is not trusted”

Possible Cause

Missing entries in certificate chain.

Recommended Solution:

Build a chain of certificates beginning with the certificate of the CA that issued the controller certificate on the controller. Include all the CA certificates and place the trusted CA certificate on top. You must place the entire chain in the same file. This means your file contains content such as this example:

```
--- BEGIN CERTIFICATE ---
*device certificate*
--- END CERTIFICATE ---
```

```
--- BEGIN CERTIFICATE ---
*intermediate CA certificate*
--- END CERTIFICATE ---
--- BEGIN CERTIFICATE ---
*Root CA certificate*
--- END CERTIFICATE ---
```

Trying to import a PKCS12 certificate which is missing the CA

Possible Cause

The pfx file might not contain the entire chain.

Recommended Solution:

Use the command to troubleshoot certificate issues.

```
Device#debug crypto pki transactions
```

1. Export the private key out.

   ```
   openssl pkcs12 -in <pkcs12 file> -out cert.key -nocerts -nodes
   ```

2. Combine the certificate as PEM.

   ```
   openssl pkcs12 -in <pkcs12 file> -out certificate.pem -nokeys -clcerts
   ```

3. Download the intermediate CA certificate as PEM. In case of a public CA, you can download the PEM file from the internet. In case of several levels of the intermediate CA, you can combine all of them into a single PEM file and name it CA.pem.

4. Rebuild the PKCS 12 file from the key, device certificates and CA certificate.

   ```
   openssl pkcs12 -export -out fixedcertchain.pfx -inkey cert.key -in certificate.pem -
   certfile CA.pem
   ```

5. Import the “fixedcertchain” to the Catalyst 9800. You must place the entire chain in the same file. This means your file will contain content as below.

   ```
   --- BEGIN CERTIFICATE ---
    *device certificate*
   --- END CERTIFICATE ---
   --- BEGIN CERTIFICATE---
    *intermediate CA certificate*
   --- END CERTIFICATE ---
   --- BEGIN CERTIFICATE ---
    *Root CA certificate*
   --- END CERTIFICATE ---
   ```

Certificate cannot be uploaded to the controller.

Possible Cause

Not connected to the TFTP server .

Certificate validity date or isssuer details is incorrect.

Recommended Solution:

Check details using

```
Device#show crypto pki certificates <cert-name>
Certificate
Status: Available
Certificate Serial Number (hex): 00A2020356CF31C818 Certificate Usage: General Purpose
Issuer:
cn=CA-KCG-lab
ou=lab-mex-wireless
o=mex-wireless
l=Guadalupe
st=Nuevo Leon
c=MX
Subject:
Name: *.lab-kcg.com
cn=*.lab-kcg.com
ou=lab-mex-wireless
o=mex-wireless
l=Benito Juarez
st=CDMX
c=MX
Validity Date:
start date: 17:14:54 UTC Feb 15 2018
end date: 17:14:54 UTC Mar 11 2023
Associated Trustpoints: cert-name
Storage: nvram:CA-KCG-lab#C818.cer
```

AP with MIC/SUDI fails to join the controller

Possible cause

Issue with management interface trustpoint settings

Recommended Solution

In case the AP cannot join the controller, you can verify the if the trustpoint is corrrect

Physical controllers

```
show wireless management trustpoint
Trustpoint Name : CISCO_IDEVID_SUDI
Certificate Info : Available
Certificate Type : MIC
Private key Info : Available
FIPS suitability : Not Applicable
```

Virtual controllers

```
show wireless management trustpoint
Trustpoint Name : ewlc-tp1
Certificate Info : Available
Certificate Type : SSC
Certificate Hash :
4a5d777c5b2071c17faef376febc08398702184e
Private key Info : Available
FIPS suitability : Not Applicable
```

If required, remove the wireless management trustpoint for the controller to fall back on the SUDI trustpoint using the following command in the global configuration mode.

```
Device(config)#no wireless management trustpoint
```

Syslog Error message

```
Dec 31 18:32:04.072: %PKI-3-CERTIFICATE_INVALID_EXPIRED: Certificate chain
validation has failed. The certificate (SN: 0159BC17) has expired. Validity
period ended on 2027-02-23T22:32:59ZDec 31 18:32:04.073:
%CERT_MGR_ERRMSG-3-CERT_VALIDATION_ERR: Chassis 1 R0/0: wncd: Certificate
Validation Error, Cert validation
status:pki_ssl_status@pki_ssl_status:PKI_SSL_ERRORDec 31 18:32:04.073:
%DTLS_TRACE_MSG-3-WLC_DTLS_ERR: Chassis 1 R0/0: wncd: DTLS Error,
session:9.10.30.117[5256] MAC: 70db.9888.cc20, Certificate validation
failed
```

Possible Cause

AP is trying to join the controller using an expired certificate.

Recommended Solution:

Allow APs to join with expired certificates by configuring policy maps

1. Create a certificate map and add the rules.

   ```
   Device#configure terminal
   Device(config)#crypto pki certificate map map1 1
   Device(config)issuer-name co (issuer-name of AP/mobility peer's cert)
   ```

   Example:

   ```
   Device#configure terminal
   Device(config)#crypto pki certificate map map1 1
   Device(config)#issuer-name co Cisco Manufacturing CA
   ```

2. Allow this policy-map to validate expired certs, under the trustpool policy.

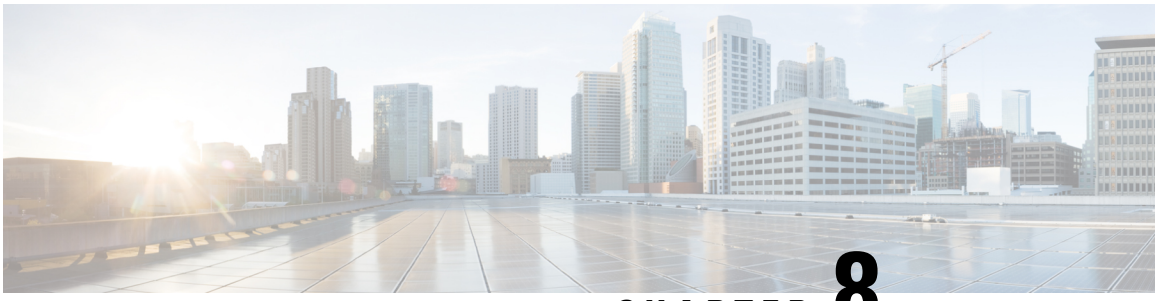
   ```
   Device#configure terminal
   Device(config)#crypto pki certificate map map1 2
   Device(config)issuer-name co act2 sudi ca
   ```

   ```
   Device#configure terminal
   Device(config)#crypto pki trustpool policy
   Device(config)#match certificate map1 allow expired-certificate
   ```

Table 6: Additional Debug Commands

Command	Description
debug crypto pki validation	Displays debugging messages related to public key infrastructure (PKI) path validation.
debug crypto pki transactions	Displays debugging messages related to public key infrastructure (PKI) certificates.
debug crypto ssl dtls events	Displays debugging messages related to encrypted ssl packets for DTLS events.
debug crypto ssl dtls errors	Displays debugging messages related to encrypted ssl packets for DTLS errors.
debug crypto ssl dtls packets	Displays debugging messages related to encrypted ssl packets for DTLS packet dump.

Command	Description
debug crypto ikev2	Displays debugging messages related to encrypted ikev2 traffic.
debug crypto est-client	Displays debugging messages related to clients who have been Enrolled Over Secure Transport(EST).
debug crypto pki scep	Displays debugging messages related to clients who have been enrolled over Simple Certificate Enrollment Protocol (SCEP).
debug crypto tls-tunnel error	Displays debugging messages related to errors in the tls-tunnel channel.
debug crypto tls-tunnel event	Displays debugging messages related to events in the tls-tunnel channel.



CHAPTER 8

Additional References for Trustpoint Configuration on Catalyst 9800

To get a detailed understanding of a particular area of trustpoint configuration, refer to the following documents:

Related Topic	Document Title
To understand the PKI Implentation	Public Key Infrastructure Configuration Guide, Cisco IOS XE Release 3S
To configure SCEP for LSC and use that for AP Join on Catalyst 9800 controllers	Configure SCEP for Locally Significant Certificate Provisioning on 9800 WLC
To understand of CSR generation and Third-Party Certiifcates for Catalyst 9800 controllers	Generate CSR for Third-Party Certificates and Download Chained Certificates to Catalyst 9800 Wireless Controllers
To understand how to generate a self-signed certificate using Day 0 wizard on the Catalyst 9800-CL virtual controller	Cisco Catalyst C9800-CL Wireless Controller Virtual Deployment Guide
To understand trustpoint configuration for local eap authentication on Catalyst 9800 controllers	Local EAP authentication on Catalyst 9800 WLC configuration example
To understand trustpoint related best practices for Catalyst 9800 controllers	Cisco Catalyst 9800 Series Configuration Best Practices

Examples of certificates available on Catalyst 9800

Here are a few examples of certificates available on the controller.

Manufacturing Installed Certificate

The following certificates are hardcoded in the physical appliance and used for AP Join by default.

```
Trustpoint CISCO_IDEVID_SUDI_LEGACY:
Subject Name:
cn=Cisco Manufacturing CA
o=Cisco Systems
Serial Number (hex): 6A6967B3000000000003
```

```
Certificate configured.


Trustpoint CISCO_IDEVID_SUDI_LEGACY0:
Subject Name:
cn=Cisco Root CA 2048
o=Cisco Systems
Serial Number (hex): 5FF87B282B54DC8D42A315B568C9ADFF
Certificate configured.

Trustpoint CISCO_IDEVID_SUDI:
Subject Name:
cn=Cisco Manufacturing CA SHA2
o=Cisco
Serial Number (hex): 02
Certificate configured.
Trustpoint CISCO_IDEVID_SUDI0:
Subject Name:
cn=Cisco Root CA M2
o=Cisco
Serial Number (hex): 01
Certificate configured.
```

Self-signed certificate available at startup or generated after factory reset

The following certificates can be used for webadmin, webauth or any other service by default, in the absence of a third-party certificate.

```
Trustpoint TP-self-signed-908292385:
Subject Name:
cn=IOS-Self-Signed-Certificate-908292385
Serial Number (hex): 01
Persistent self-signed certificate trust point
Using key label TP-self-signed-908292385

Trustpoint SLA-TrustPoint:
Subject Name:
cn=Cisco Licensing Root CA
o=Cisco
Serial Number (hex): 01
Certificate configured.
```