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

  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

  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.