



Persistent Self-Signed Certificates

The Persistent Self-Signed Certificates feature saves a certificate generated by a secure HTTP (HTTPS) server for the Secure Sockets Layer (SSL) handshake in a router's startup configuration.

Feature History for Persistent Self-Signed Certificates

Release	Modification
12.3(14)T	This feature was introduced.

Finding Support Information for Platforms and Cisco IOS Software Images

Use Cisco Feature Navigator to find information about platform support and Cisco IOS software image support. Access Cisco Feature Navigator at <http://www.cisco.com/go/fn>. You must have an account on Cisco.com. If you do not have an account or have forgotten your username or password, click **Cancel** at the login dialog box and follow the instructions that appear.

Contents

- [Prerequisites for Persistent Self-Signed Certificates, page 1](#)
- [Restrictions for Persistent Self-Signed Certificates, page 2](#)
- [Information About Persistent Self-Signed Certificates, page 2](#)
- [How to Configure a Persistent Self-Signed Certificate, page 4](#)
- [Configuration Examples for Persistent Self-Signed Certificates, page 8](#)
- [Additional References, page 11](#)
- [Command Reference, page 13](#)
- [Glossary, page 28](#)

Prerequisites for Persistent Self-Signed Certificates

You must load an image that supports SSL.



Corporate Headquarters:
Cisco Systems, Inc., 170 West Tasman Drive, San Jose, CA 95134-1706 USA

Copyright © 2005 Cisco Systems, Inc. All rights reserved.

Restrictions for Persistent Self-Signed Certificates

You can configure only one trustpoint for a persistent self-signed certificate.

Information About Persistent Self-Signed Certificates

To use the Persistent Self-Signed Certificates feature, you need to understand the following concepts:

- [Feature Overview of Persistent Self-Signed Certificates, page 2](#)
- [Benefits of Persistent Self-Signed Certificates, page 3](#)

Feature Overview of Persistent Self-Signed Certificates

Cisco IOS software has an HTTPS server that allows access to web-based management pages using a secure SSL connection. SSL requires the server to have an X.509 certificate that is sent to the client (web browser) during the SSL handshake to establish a secure connection between the server and the client (Figure 1).

Figure 1 Sample Topology



The client expects the SSL server's certificate to be verifiable using a certificate the client already possesses.

If Cisco IOS software does not have a certificate that the HTTPS server can use, the server generates a self-signed certificate by calling a public key infrastructure (PKI) application programming interface (API). When the client receives this self-signed certificate and is unable to verify it, intervention is needed. The client asks you if the certificate should be accepted and saved for future use. If you accept the certificate, the SSL handshake continues.

Future SSL handshakes between the same client and the server use the same certificate. However, if the router is reloaded, the self-signed certificate is lost. The HTTPS server must then create a new self-signed certificate. This new self-signed certificate does not match the previous certificate so you are once again asked to accept it.

Requesting acceptance of the router's certificate each time that the router reloads can be annoying and may present an opportunity for an attacker to substitute an unauthorized certificate during the time that you are being asked to accept the certificate.

The Persistent Self-Signed Certificates feature overcomes all these limitations by saving a certificate in the router's startup configuration.

Benefits of Persistent Self-Signed Certificates

Enhanced Security

Having a persistent self-signed certificate stored in the router's startup configuration (NVRAM) lessens the opportunity for an attacker to substitute an unauthorized certificate because the browser is able to compare the certificate offered by the router with the previously saved certificate and warn you if the certificate has changed.

Ease of Use

Having a persistent self-signed certificate stored in the router's startup configuration eliminates the user intervention that was necessary to accept the certificate every time that the router reloads.

Improved Performance

Because user intervention is no longer necessary to accept the certificate, the secure connection process is faster.

How to Configure a Persistent Self-Signed Certificate



Note

This section is optional because if you enable the HTTPS server, it generates a self-signed certificate automatically using default values.

This section contains the following procedures:

- [Configuring a Trustpoint and Specifying Self-Signed Certificate Parameters, page 4](#) (optional)
- [Enabling the HTTPS Server: Example, page 8](#) (optional)
- [Verifying the Configuration, page 6](#) (optional)

Configuring a Trustpoint and Specifying Self-Signed Certificate Parameters

Perform the following task to configure a trustpoint and specify self-signed certificate parameters.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **crypto pki trustpoint *name***
4. **enroll selfsigned**
5. **subject-name [x.500-name]**
6. **rsa keypair *key-label* [*key-size* [*encryption-key-size*]]**
7. **crypto pki enroll *name***
8. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Router> enable	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	configure terminal Example: Router# configure terminal	Enters global configuration mode.
Step 3	crypto pki trustpoint <i>name</i> Example: Router(config)# crypto pki trustpoint local	Declares the certification authority (CA) that your router should use and enters ca-trustpoint configuration mode. Note Effective with Cisco IOS Release 12.3(8)T, the crypto pki trustpoint command replaced the crypto ca trustpoint command.

	Command or Action	Purpose
Step 4	enrollment selfsigned Example: Router(ca-trustpoint)# enrollment selfsigned	Specifies self-signed enrollment.
Step 5	subject-name [<i>x.500-name</i>] Example: Router(ca-trustpoint)# subject-name	(Optional) Specifies the requested subject name to be used in the certificate request. <ul style="list-style-type: none"> If the <i>x-500-name</i> argument is not specified, the fully qualified domain name (FQDN), which is the default subject name, is used.
Step 6	rsakeypair key-label [<i>key-size</i> [<i>encryption-key-size</i>]] Example: Router(ca-trustpoint)# rsakeypair examplekeys 1024 1024	(Optional) Specifies which key pair to associate with the certificate. <ul style="list-style-type: none"> <i>key-label</i> will be generated during enrollment if it does not already exist or if the auto-enroll regenerate command was issued. Specify the <i>key-size</i> for generating the key, and specify the <i>encryption-key-size</i> to request separate encryption, signature keys, and certificates. <p>Note If this command is not enabled, the FQDN key pair is used.</p>
Step 7	crypto pki enroll name Example: Router(ca-trustpoint)# crypto pki enroll local	Tells the router to generate the persistent self-signed certificate.
Step 8	end Example: Router(ca-trustpoint)# end	(Optional) Exits ca-trustpoint configuration mode.

Enabling the HTTPS Server

Perform the following task to enable the HTTPS server.

Prerequisites

To specify parameters, you must create a trustpoint and configure it. To use default values, delete any existing self-signed trustpoints. Deleting all self-signed trustpoints causes the HTTPS server to generate a persistent self-signed certificate using default values as soon as it is enabled.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **ip http secure-server**
4. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	<pre>enable</pre> <p>Example: Router> enable </p>	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	<pre>configure terminal</pre> <p>Example: Router# configure terminal </p>	Enters global configuration mode.
Step 3	<pre>ip http secure-server</pre> <p>Example: Router(config)# ip http secure-server </p>	Enables the secure HTTP web server. <p>Note A key pair (modulus 1024) and a certificate are generated.</p>
Step 4	<pre>end</pre> <p>Example: Router(config)# end </p>	Exits global configuration mode.

**Note**

You must issue a **write memory** command to save the configuration. This saves the self-signed certificate and the HTTPS server in enabled mode.

Verifying the Configuration

Perform the following task to verify that a self-signed certificate and a trustpoint have been created.

SUMMARY STEPS

1. enable
2. show crypto pki certificates [*trustpoint-name* [verbose]]
3. show crypto pki trustpoints [*status* | *label* [*status*]]
4. end

DETAILED STEPS

	Command or Action	Purpose
Step 1	<p>enable</p> <p>Example: Router> enable</p>	<p>Enables privileged EXEC mode.</p> <ul style="list-style-type: none"> Enter your password if prompted.
Step 2	<p>show crypto pki certificates [<i>trustpoint-name</i> [<i>verbose</i>]]</p> <p>Example: Router# show crypto pki certificates local verbose</p>	<p>Displays information about your certificate, the certification authority certificate, and any registration authority certificates.</p> <p>Note Effective with Cisco IOS Release 12.3(7)T, the show crypto pki certificates command replaced the show crypto ca certificates command.</p>
Step 3	<p>show crypto pki trustpoints [<i>status</i> <i>label</i> [<i>status</i>]]</p> <p>Example: Router# show crypto pki trustpoints status</p>	<p>Displays the trustpoints that are configured in the router.</p> <p>Note Effective with Cisco IOS Release 12.3(7)T, the show crypto pki trustpoints command replaced the show crypto ca trustpoints command.</p>
Step 4	<p>exit</p> <p>Example: Router# end</p>	<p>(Optional) Exits privileged EXEC mode.</p>

Configuration Examples for Persistent Self-Signed Certificates

This section contains the following configuration examples:

- [Creating a Persistent Self-Signed Certificate: Example, page 8](#)
- [Enabling the HTTPS Server: Example, page 8](#)
- [Verifying the Configuration: Example, page 9](#)

Creating a Persistent Self-Signed Certificate: Example

In the following example, a trustpoint named local is declared, its enrollment is requested, and a self-signed certificate with an IP address is generated:

```
Router# configure terminal

Enter configuration commands, one per line.  End with CNTL/Z.

Router(config)# crypto pki trustpoint local

Router(ca-trustpoint)# enrollment selfsigned

Router(ca-trustpoint)# end

Router# configure terminal

Enter configuration commands, one per line.  End with CNTL/Z.

Router(config)# crypto pki enroll local

Nov 29 20:51:13.067: %SSH-5-ENABLED: SSH 1.99 has been enabled
Nov 29 20:51:13.267: %CRYPTO-6-AUTOGEN: Generated new 512 bit key pair

% Include the router serial number in the subject name? [yes/no]: yes

% Include an IP address in the subject name? [no]: yes

Enter Interface name or IP Address[]: ethernet 0

Generate Self Signed Router Certificate? [yes/no]: yes

Router Self Signed Certificate successfully created
```



Note

A router can have only one self-signed certificate. If you attempt to enroll a trustpoint configured for a self-signed certificate and one already exists, you receive a notification and are asked if you want to replace it. If so, a new self-signed certificate is generated to replace the existing one.

Enabling the HTTPS Server: Example

In the following example, the HTTPS server is enabled and a default trustpoint is generated since one was not previously configured:

```
Router# configure terminal

Enter configuration commands, one per line.  End with CNTL/Z.
```

```
Router(config)# ip http secure-server

% Generating 1024 bit RSA keys ...[OK]

*Dec 21 19:14:15.421:%PKI-4-NOAUTOSAVE:Configuration was modified. Issue "write memory"
to save new certificate
Router(config)#
```

**Note**

You need to save the configuration to NVRAM if you want to keep the self-signed certificate and have the HTTPS server enabled following router reloads.

The following message also appears:

```
*Dec 21 19:14:10.441:%SSH-5-ENABLED:SSH 1.99 has been enabled
Router(config)#
```

**Note**

Creation of the key pair used with the self-signed certificate causes the Secure Shell (SSH) server to start. This behavior cannot be suppressed. You may want to modify your access control lists (ACLs) to permit or deny SSH access to the router.

Verifying the Configuration: Example

The following example displays information about the self-signed certificate that you just created:

```
Router# show crypto pki certificates

Router Self-Signed Certificate
  Status: Available
  Certificate Serial Number: 01
  Certificate Usage: General Purpose
  Issuer:
    cn=IOS-Self-Signed-Certificate-3326000105
  Subject:
    Name: IOS-Self-Signed-Certificate-3326000105
    cn=IOS-Self-Signed-Certificate-3326000105
  Validity Date:
    start date: 19:14:14 GMT Dec 21 2004
    end date: 00:00:00 GMT Jan 1 2020
  Associated Trustpoints: TP-self-signed-3326000105
```

**Note**

The number 3326000105 above is the router's serial number and varies depending on the router's actual serial number.

The following example displays information about the key pair corresponding to the self-signed certificate:

```
Router# show crypto key mypubkey rsa

% Key pair was generated at: 19:14:10 GMT Dec 21 2004
Key name: TP-self-signed-3326000105
Usage: General Purpose Key
Key is not exportable.
Key Data:
  30819F30 0D06092A 864886F7 0D010101 05000381 8D003081 89028181 00B88F70
  6BC78B6D 67D6CFF3 135C1D91 8F360292 CA44A032 5AC1A8FD 095E4865 F8C95A2B
  BFD1C2B7 E64A3804 9BBD7326 207BD456 19BAB78B D075E78E 00D2560C B09289AE
```

```

6DECB8B0 6672FB3A 5CDAEE92 9D4C4F71 F3BCB269 214F6293 4BA8FABF 9486BCFC
2B941BCA 550999A7 2EFE12A5 6B7B669A 2D88AB77 39B38E0E AA23CB8C B7020301 0001
% Key pair was generated at: 19:14:13 GMT Dec 21 2004
Key name: TP-self-signed-3326000105.server
Usage: Encryption Key
Key is not exportable.
Key Data:
307C300D 06092A86 4886F70D 01010105 00036B00 30680261 00C5680E 89777B42
463E5783 FE96EA9E F446DC7B 70499AF3 EA266651 56EE29F4 5B003D93 2FC9F81D
8A46E12F 3FBAC2F3 046ED9DD C5F27C20 1BBA6B9B 08F16E45 C34D6337 F863D605
34E30F0E B4921BC5 DAC9EBBA 50C54AA0 BF551BDD 88453F50 61020301 0001
Router#

```



Note

The second key pair with the name TP-self-signed-3326000105.server is the SSH key pair and is generated once any key pair is created on the router and SSH starts up.

The following example displays information about the trustpoint named local:

```

Router# show crypto pki trustpoints

Trustpoint local:
  Subject Name:
    serialNumber=C63EBBE9+ipaddress=10.3.0.18+hostname=test.cisco.com
    Serial Number: 01
  Persistent self-signed certificate trust point

```

Additional References

The following sections provide references related to the Persistent Self-Signed Certificates feature.

Related Documents

Related Topic	Document Title
Security commands: complete command syntax, command mode, defaults, usage guidelines, and examples	<i>Cisco IOS Security Command Reference</i> , Release 12.3 T
Security features including trustpoints, certificate enrollment, and authentication	<i>Cisco IOS Security Configuration Guide</i> , Release 12.3
RSA key pairs	<i>Multiple RSA Key Pair Support</i> feature module, Release 12.2(8)T

Standards

Standards	Title
No new or modified standards are supported by this feature, and support for existing standards has not been modified by this feature.	—

MIBs

MIBs	MIBs Link
No new or modified MIBs are supported by this feature, and support for existing MIBs has not been modified by this feature.	To locate and download MIBs for selected platforms, Cisco IOS releases, and feature sets, use Cisco MIB Locator found at the following URL: http://www.cisco.com/go/mibs

RFCs

RFCs	Title
No new or modified RFCs are supported by this feature, and support for existing RFCs has not been modified by this feature.	—

Technical Assistance

Description	Link
The Cisco Technical Support website contains thousands of pages of searchable technical content, including links to products, technologies, solutions, technical tips, and tools. Registered Cisco.com users can log in from this page to access even more content.	http://www.cisco.com/techsupport

Command Reference

This section documents new and modified commands.

New Commands

- [enrollment selfsigned](#)

Modified Commands

- [crypto pki enroll](#)
- [crypto pki trustpoint](#)
- [show crypto pki certificates](#)
- [show crypto pki trustpoints](#)

crypto pki enroll

To obtain the certificate(s) for your router from the certificate authority (CA), use the **crypto pki enroll** command in global configuration mode. To delete a current enrollment request, use the **no** form of this command.

crypto pki enroll *name*

no crypto pki enroll *name*

Syntax Description

<i>name</i>	Specifies the name of the CA. Use the same name as when you declared the CA using the crypto pki trustpoint command.
-------------	---

Defaults

No default behavior or values.

Command Modes

Global configuration

Command History

Release	Modification
11.3 T	The crypto ca enroll command was introduced.
12.3(7)T	This command replaced the crypto ca enroll command.
12.3(14)T	The command was modified to include self-signed certificate information.

Usage Guidelines

This command requests certificates from the CA for all of your router's Rivest, Shamir, and Adelman (RSA) key pairs. This task is also known as enrolling with the CA. (Technically, enrolling and obtaining certificates are two separate events, but they both occur when this command is issued.)

Your router needs a signed certificate from the CA for each RSA key pairs of your router; if you previously generated general purpose keys, this command obtains the one certificate corresponding to the one general purpose RSA key pair. If you previously generated special usage keys, this command obtains two certificates corresponding to each of the special usage RSA key pairs.

If you already have a certificate for your keys you are unable to complete this command; instead, you are prompted to remove the existing certificate first. (You can remove existing certificates with the **no certificate** command.)

The **crypto pki enroll** command is not saved in the router configuration.



Note

If your router reboots after you issue the **crypto pki enroll** command but before you receive the certificate(s), you must reissue the command.

Responding to Prompts

When you issue the **crypto pki enroll** command, you are prompted a number of times.

First, you are prompted to create a challenge password. This password can be up to 80 characters in length. This password is necessary in the event that you ever need to revoke your router's certificate(s). When you ask the CA administrator to revoke your certificate, you must supply this challenge password as a protection against fraudulent or mistaken revocation requests.



Note

This password is not stored anywhere, so you need to remember this password.

If you lose the password, the CA administrator may still be able to revoke the router's certificate but will require further manual authentication of the router administrator identity.

You are also prompted to indicate whether or not your router's serial number should be included in the obtained certificate. The serial number is not used by IP Security or Internet Key Exchange, but may be used by the CA to either authenticate certificates or to later associate a certificate with a particular router. (Note that the serial number stored is the serial number of the internal board, not the one on the enclosure.) Ask your CA administrator if serial numbers should be included. If you are in doubt, include the serial number.

Normally, you would not include the IP address because the IP address binds the certificate more tightly to a specific entity. Also, if the router is moved, you would need to issue a new certificate. Finally, a router has multiple IP addresses, any of which might be used with IPSec.

If you indicate that the IP address should be included, you will then be prompted to specify the interface of the IP address. This interface should correspond to the interface that you apply your crypto map set to. If you apply crypto map sets to more than one interface, specify the interface that you name in the **crypto map local-address** command.

Examples

In the following example, a router with a general-purpose RSA key pair requests a certificate from the CA. When the router displays the certificate fingerprint, the administrator verifies this number by calling the CA administrator, who checks the number. The fingerprint is correct, so the router administrator accepts the certificate.

There can be a delay between when the router administrator sends the request and when the certificate is actually received by the router. The amount of delay depends on the CA method of operation.

```
Router(config)# crypto pki enroll myca
%
% 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: <mypassword>
Re-enter password: <mypassword>

% The subject name in the certificate will be: myrouter.example.com
% Include the router serial number in the subject name? [yes/no]: yes
% The serial number in the certificate will be: 03433678
% Include an IP address in the subject name [yes/no]? yes
Interface: ethernet0/0
Request certificate from CA [yes/no]? yes
% Certificate request sent to Certificate Authority
% The certificate request fingerprint will be displayed.
% The 'show crypto pki certificates' command will also show the fingerprint.
```

Some time later, the router receives the certificate from the CA and displays the following confirmation message:

```
Router(config)# Fingerprint: 01234567 89ABCDEF FEDCBA98 75543210

%CRYPTO-6-CERTRET: Certificate received from Certificate Authority

Router(config)#
```

If necessary, the router administrator can verify the displayed fingerprint with the CA administrator.

If there is a problem with the certificate request and the certificate is not granted, the following message is displayed on the console instead:

```
%CRYPTO-6-CERTREJ: Certificate enrollment request was rejected by Certificate Authority
```

The subject name in the certificate is automatically assigned to be the same as the RSA key pair's name. In the above example, the RSA key pair was named "myrouter.example.com." (The router assigned this name.)

Requesting certificates for a router with special usage keys would be the same as the previous example, except that two certificates would have been returned by the CA. When the router received the two certificates, the router would have displayed the same confirmation message:

```
%CRYPTO-6-CERTRET: Certificate received from Certificate Authority
```

Related Commands

Command	Description
debug crypto pki messages	Displays debug messages for the details of the interaction (message dump) between the CA and the router.
debug crypto pki transactions	Displays debug messages for the trace of interaction (message type) between the CA and the router.
show crypto pki certificates	Displays information about your certificate, the certificate of the CA, and any RA certificates.

crypto pki trustpoint

To declare the trustpoint that your router should use, use the **crypto pki trustpoint** command in global configuration mode. To delete all identity information and certificates associated with the trustpoint, use the **no** form of this command.

crypto pki trustpoint *name*

no crypto pki trustpoint *name*

Syntax Description

<i>name</i>	Creates a name for the trustpoint. (If you previously declared the trustpoint and just want to update its characteristics, specify the name you previously created.)
-------------	--

Defaults

Your router does not recognize any trustpoints until you declare a trustpoint using this command.

Command Modes

Global configuration

Command History

Release	Modification
12.2(8)T	The crypto ca trustpoint command was added.
12.2(15)T	The match certificate subcommand was introduced.
12.3(7)T	This command replaced the crypto ca trustpoint command. You can still enter the crypto ca trusted-root or crypto ca trustpoint command, but the command will be written in the configuration as “crypto pki trustpoint.”
12.3(14)T	The enrollment selfsigned subcommand was introduced.

Usage Guidelines

Use the **crypto pki trustpoint** command to declare a trustpoint, which can be a self-signed root certificate authority (CA) or a subordinate CA. Issuing the **crypto pki trustpoint** command puts you in ca-trustpoint configuration mode.

You can specify characteristics for the trustpoint using the following subcommands:

- **crl**—Queries the certificate revocation list (CRL) to ensure that the certificate of the peer has not been revoked.
- **default (ca-trustpoint)**—Resets the value of ca-trustpoint configuration mode subcommands to their defaults.
- **enrollment**—Specifies enrollment parameters (optional).
- **enrollment http-proxy**—Accesses the CA by HTTP through the proxy server.
- **enrollment selfsigned**—Specifies self-signed enrollment (optional).
- **match certificate**—Associates a certificate-based access control list (ACL) defined with the **crypto ca certificate map** command.
- **primary**—Assigns a specified trustpoint as the primary trustpoint of the router.

- **root**—Defines the TFTP to get the CA certificate and specifies both a name for the server and a name for the file that will store the CA certificate.

The following example shows how to declare the CA named ka and specify enrollment and CRL parameters:

```
crypto pki trustpoint ka
  enrollment url http://kahului:80
```

The following example shows a certificate-based ACL with the label Group defined in a **crypto pki certificate map** command and included in the **match certificate** subcommand of the **crypto pki trustpoint** command:

```
crypto pki certificate map Group 10
  subject-name co ou=WAN
  subject-name co o=Cisco
!
crypto pki trustpoint pkil
  match certificate Group
```

The following example shows a self-signed certificate being designated for a trustpoint named local using the **enrollment selfsigned** subcommand of the **crypto pki trustpoint** command:

```
crypto pki trustpoint local
  enrollment selfsigned
```

Related Commands

Command	Description
crl	Queries the CRL to ensure that the certificate of the peer has not been revoked.
default (ca-trustpoint)	Resets the value of a ca-trustpoint configuration subcommand to its default.
enrollment	Specifies the enrollment parameters of your CA.
enrollment http-proxy	Accesses the CA by HTTP through the proxy server.
primary	Assigns a specified trustpoint as the primary trustpoint of the router.
root	Obtains the CA certificate via TFTP.

enrollment selfsigned

To specify self-signed enrollment for a trustpoint, use the **enrollment selfsigned** command in ca-trustpoint configuration mode. To delete self-signed enrollment from a trustpoint, use the **no** form of this command.

enrollment selfsigned

no enrollment selfsigned

Syntax Description This command has no arguments or keywords.

Defaults No default behavior or values

Command Modes ca-trustpoint configuration

Command History	Release	Modification
	12.3(14)T	This command was introduced.

Usage Guidelines Before you can use the **enrollment selfsigned** command, you must enable the **crypto pki trustpoint** command, which defines the trustpoint and enters ca-trustpoint configuration mode.

If you do not use this command, you should specify another enrollment method for the router by using an enrollment command such as **enrollment url** or **enrollment terminal**.

Examples The following example shows a self-signed certificate being designated for a trustpoint named local:

```
crypto pki trustpoint local
 enrollment selfsigned
```

Related Commands	Command	Description
	crypto pki trustpoint	Declares the CA that your router should use.

show crypto pki certificates

To display information about your certificate, the certification authority certificate, and any registration authority certificates, use the **show crypto pki certificates** command in privileged EXEC mode.

show crypto pki certificates [*trustpoint-name* [**verbose**]]

Syntax Description

<i>trustpoint-name</i>	(Optional) Name of the trustpoint. Using this argument indicates that only certificates that are related to the trustpoint are to be displayed.
verbose	(Optional) More detailed information is to be displayed.
Note	The verbose keyword can be used only if a trustpoint name is entered.

Command Modes

Privileged EXEC

Command History

Release	Modification
11.3 T	The show crypto ca certificates command was introduced.
12.2(13)T	The <i>trustpoint-name</i> argument was added.
12.3(7)T	This command replaced the show crypto ca certificates command.
12.3(8)T	The verbose keyword was added.
12.3(14)T	The command output was modified to include persistent self-signed certificate parameters.

Usage Guidelines

This command shows information about the following certificates:

- Your certificate, if you have requested one from the certificate authority (CA) (see the **crypto pki enroll** command)
- The certificate of the CA, if you have received the certificate of the CA (see the **crypto pki authenticate** command)
- RA certificates, if you have received registration authority (RA) certificates (see the **crypto pki authenticate** command)
- A self-signed certificate, if one has been requested

Examples

The following is sample output from the **show crypto pki certificates** command after you authenticated the CA by requesting the certificate of the CA and public key with the **crypto pki authenticate** command:

```
CA Certificate
  Status: Available
  Certificate Serial Number: 3051DF7123BEE31B8341DFE4B3A338E5F
  Key Usage: Not Set
```

The CA certificate might show Key Usage as “Not Set.”

The following is sample output from the **show crypto pki certificates** command, and it shows the certificate of the router and the certificate of the CA. In this example, a single, general-purpose Rivest, Shamir, and Adelman (RSA) key pair was previously generated, and a certificate was requested but not received for that key pair.

```
Certificate
  Subject Name
    Name: myrouter.example.com
    IP Address: 10.0.0.1
    Serial Number: 04806682
  Status: Pending
  Key Usage: General Purpose
  Fingerprint: 428125BD A3419600 3F6C7831 6CD8FA95 00000000
```

```
CA Certificate
  Status: Available
  Certificate Serial Number: 3051DF7123BEE31B8341DFE4B3A338E5F
  Key Usage: Not Set
```

Note that in the previous sample, the certificate status of the router shows “Pending.” After the router receives its certificate from the CA, the Status field changes to “Available” in the **show** output.

The following is sample output from the **show crypto pki certificates** command, and it shows the certificates of two routers and the certificate of the CA. In this example, special-usage RSA key pairs were previously generated, and a certificate was requested and received for each key pair.

```
Certificate
  Subject Name
    Name: myrouter.example.com
    IP Address: 10.0.0.1
  Status: Available
  Certificate Serial Number: 428125BDA34196003F6C78316CD8FA95
  Key Usage: Signature
```

```
Certificate
  Subject Name
    Name: myrouter.example.com
    IP Address: 10.0.0.1
  Status: Available
  Certificate Serial Number: AB352356AFCD0395E333CCFD7CD33897
  Key Usage: Encryption
```

```
CA Certificate
  Status: Available
  Certificate Serial Number: 3051DF7123BEE31B8341DFE4B3A338E5F
  Key Usage: Not Set
```

The following is sample output from the **show crypto pki certificates** command when the CA supports an RA. In this example, the CA and RA certificates were previously requested with the **crypto pki authenticate** command.

```
CA Certificate
  Status: Available
  Certificate Serial Number: 3051DF7123BEE31B8341DFE4B3A338E5F
  Key Usage: Not Set
```

```
RA Signature Certificate
  Status: Available
  Certificate Serial Number: 34BCF8A0
  Key Usage: Signature
```

```
RA KeyEncipher Certificate
  Status: Available
```

```
Certificate Serial Number: 34BCF89F
Key Usage: Encryption
```

The following is sample output from the **show crypto pki certificates** command using the optional *trustpoint-name* argument and **verbose** keyword. The output shows the certificate of a router and the certificate of the CA. In this example, general-purpose RSA key pairs were previously generated, and a certificate was requested and received for the key pair.

```
Certificate
  Status: Available
  Version: 3
  Certificate Serial Number: 18C1EE03000000004CBD
  Certificate Usage: General Purpose
  Issuer:
    cn=msca-root
    ou=pki msca-root
    o=cisco
    l=santa cruz2
    st=CA
    c=US
    ea=user@example.com
  Subject:
    Name: myrouter.example.com
    hostname=myrouter.example.com
  CRL Distribution Points:
    http://msca-root/CertEnroll/msca-root.crl
  Validity Date:
    start date: 19:50:40 GMT Oct 5 2004
    end   date: 20:00:40 GMT Oct 12 2004
  Subject Key Info:
    Public Key Algorithm: rsaEncryption
    RSA Public Key: (360 bit)
  Signature Algorithm: SHA1 with RSA Encryption
  Fingerprint MD5: 2B5F53E6 E3E892E6 3A9D3706 01261F10
  Fingerprint SHA1: 315D127C 3AD34010 40CE7F3A 988BBD5A CD528824
  X509v3 extensions:
    X509v3 Key Usage: A0000000
    Digital Signature
    Key Encipherment
    X509v3 Subject Key ID: D156E92F 46739CBA DFE66D2D 3559483E B41ECCF4
    X509v3 Authority Key ID: 37F3CC61 AF5E7C0B 434AB364 CF9FA0C1 B17C50D9
    Authority Info Access:
  Associated Trustpoints: msca-root
  Key Label: myrouter.example.com

CA Certificate
  Status: Available
  Version: 3
  Certificate Serial Number: 1244325DE0369880465F977A18F61CA8
  Certificate Usage: Signature
  Issuer:
    cn=msca-root
    ou=pki msca-root
    o=cisco
    l=santa cruz2
    st=CA
    c=US
    ea=user@example.com
  Subject:
    cn=msca-root
    ou=pki msca-root
    o=cisco
```

```

l=santa cruz2
st=CA
c=US
ea=user@example.com
CRL Distribution Points:
  http://msca-root.example.com/CertEnroll/msca-root.crl
Validity Date:
  start date: 22:19:29 GMT Oct 31 2002
  end   date: 22:27:27 GMT Oct 31 2017
Subject Key Info:
  Public Key Algorithm: rsaEncryption
  RSA Public Key: (512 bit)
Signature Algorithm: SHA1 with RSA Encryption
Fingerprint MD5: 84E470A2 38176CB1 AA0476B9 C0B4F478
Fingerprint SHA1: 0F57170C 654A5D7D 10973553 EFB0F94F 2FAF9837
X509v3 extensions:
  X509v3 Key Usage: C6000000
    Digital Signature
    Non Repudiation
    Key Cert Sign
    CRL Signature
  X509v3 Subject Key ID: 37F3CC61 AF5E7C0B 434AB364 CF9FA0C1 B17C50D9
  X509v3 Basic Constraints:
    CA: TRUE
  Authority Info Access:
  Associated Trustpoints: msca-root

```

The following example shows that a self-signed certificate has been created using a user-defined trustpoint:

```

Router Self-Signed Certificate
Status: Available
Certificate Serial Number: 01
Certificate Usage: General Purpose
Issuer:
  serialNumber=C63EBBE9+ipaddress=10.3.0.18+hostname=test.cisco.com
Subject:
  Name: router.cisco.com
  IP Address: 10.3.0.18
  Serial Number: C63EBBE9
  serialNumber=C63EBBE9+ipaddress=10.3.0.18+hostname=test.cisco.com
Validity Date:
  start date: 20:51:40 GMT Nov 29 2004
  end   date: 00:00:00 GMT Jan 1 2020
Associated Trustpoints: local

```

Related Commands

Command	Description
crypto pki authenticate	Authenticates the CA (by obtaining the certificate of the CA).
crypto pki enroll	Obtains the certificates of your router from the CA.
debug crypto pki messages	Displays debug messages for the details of the interaction (message dump) between the CA and the route.
debug crypto pki transactions	Displays debug messages for the trace of interaction (message type) between the CA and the router.

show crypto pki trustpoints

To display the trustpoints that are configured in the router, use the **show crypto pki trustpoints** command in privileged or user EXEC mode.

```
show crypto pki trustpoints [status | label [status]]
```

Syntax Description	status	(Optional) Trustpoint status.
	label	(Optional) Trustpoint name.
Defaults	If the <i>label</i> argument (trustpoint name) is not specified, command output is displayed for all trustpoints.	
Command Modes	Privileged EXEC User EXEC	
Command History	Release	Modification
	12.2(8)T	The show crypto ca trustpoints command was introduced.
	12.3(7)T	This command replaced the show crypto ca trustpoints command.
	12.3(11)T	The status keyword and <i>label</i> argument were added.
	12.3(14)T	The command output was modified to include persistent self-signed certificate parameters.
Usage Guidelines	If you enter the show crypto ca roots command, it will have the same effect as entering the show crypto pki trustpoints command.	

Examples

The following is sample output from the **show crypto pki trustpoints** command:

```
Router# show crypto pki trustpoints

Trustpoint bo:
  Subject Name:
    CN = bomborra Certificate Manager
    O = cisco.com
    C = US
    Serial Number:01
  Certificate configured.
  CEP URL:http://bomborra
  CRL query url:ldap://bomborra
```

The following is sample output from the **show crypto pki trustpoints** command when a persistent self-signed certificate has been configured:

```
Router# show crypto pki trustpoints

Trustpoint local:
  Subject Name:
```

```

serialNumber=C63EBBE9+ipaddress=10.3.0.18+hostname=test.cisco.com
  Serial Number: 01
Persistent self-signed certificate trust point

```

The following output using the **status** keyword shows that the trustpoint is configured in query mode and is currently trying to query the certificates (the certificate authority (CA) certificate and the router certificate are both pending):

```

Router# show crypto pki trustpoints status

Trustpoint yni:
  Issuing CA certificate pending:
    Subject Name:
      cn=nsca-r1 Cert Manager,ou=pki,o=cisco.com,c=US
    Fingerprint: C21514AC 12815946 09F635ED FBB6CF31
  Router certificate pending:
    Subject Name:
      hostname=trance.cisco.com,o=cisco.com
  Next query attempt:
    52 seconds

```

The following output using the **status** keyword shows that the trustpoint has been authenticated:

```

Router# show crypto pki trustpoints status

Trustpoint yni:
  Issuing CA certificate configured:
    Subject Name:
      cn=nsca-r1 Cert Manager,ou=pki,o=cisco.com,c=US
    Fingerprint: C21514AC 12815946 09F635ED FBB6CF31
  State:
    Keys generated ..... No
    Issuing CA authenticated ..... Yes
    Certificate request(s) ..... None

```

The following output using the **status** keyword shows that the trustpoint is enrolling and that two of the certificate requests are pending (Signature and Encryption):

```

Router# show crypto pki trustpoints status

Trustpoint yni:
  Issuing CA certificate configured:
    Subject Name:
      cn=nsca-r1 Cert Manager,ou=pki,o=cisco.com,c=US
    Fingerprint: C21514AC 12815946 09F635ED FBB6CF31
  Router Signature certificate pending:
    Requested Subject Name:
      hostname=trance.cisco.com
    Request Fingerprint: FAE0D74E BB844EA1 54B26698 56AB42EC
    Enrollment polling: 1 times (9 left)
    Next poll: 32 seconds
  Router Encryption certificate pending:
    Requested Subject Name:
      hostname=trance.cisco.com
    Request Fingerprint: F4E815DB D9D9B60F 9B5B1724 3E155DBF
    Enrollment polling: 1 times (9 left)
    Next poll: 44 seconds
  Last enrollment status: Pending
  State:
    Keys generated ..... Yes (Signature, Encryption)
    Issuing CA authenticated ..... Yes
    Certificate request(s) ..... Pending

```

The following output using the **status** keyword shows that enrollment has succeeded and that two router certificates have been granted (Signature and Encryption):

```
Router# show crypto pki trustpoints status

Trustpoint yni:
  Issuing CA certificate configured:
    Subject Name:
      cn=nsca-r1 Cert Manager,ou=pki,o=cisco.com,c=US
    Fingerprint: C21514AC 12815946 09F635ED FBB6CF31
  Router Signature certificate configured:
    Subject Name:
      hostname=trance.cisco.com,o=cisco.com
    Fingerprint: 8A370B8B 3B6A2464 F962178E 8385E9D6
  Router Encryption certificate configured:
    Subject Name:
      hostname=trance.cisco.com,o=cisco.com
    Fingerprint: 43A03218 C0AFF844 AE0C162A 690B414A
  Last enrollment status: Granted
  State:
    Keys generated ..... Yes (Signature, Encryption)
    Issuing CA authenticated ..... Yes
    Certificate request(s) ..... Yes
```

The following output using the **status** keyword shows that trustpoint enrollment has been rejected:

```
Router# show crypto pki trustpoints status

Trustpoint yni:
  Issuing CA certificate configured:
    Subject Name:
      cn=nsca-r1 Cert Manager,ou=pki,o=cisco.com,c=US
    Fingerprint: C21514AC 12815946 09F635ED FBB6CF31
  Last enrollment status: Rejected
  State:
    Keys generated ..... Yes (General Purpose)
    Issuing CA authenticated ..... Yes
    Certificate request(s) ..... None
```

The following output using the **status** keyword shows that enrollment has succeeded and that the router is configured for autoenrollment using a regenerated key. In addition, the running configuration has been modified so that it will not be saved automatically after autoenrollment.

```
Router# show crypto pki trustpoints status

Trustpoint yni:
  Issuing CA certificate configured:
    Subject Name:
      cn=nsca-r1 Cert Manager,ou=pki,o=cisco.com,c=US
    Fingerprint: C21514AC 12815946 09F635ED FBB6CF31
  Router General Purpose certificate configured:
    Subject Name:
      hostname=trance.cisco.com,o=cisco.com
    Fingerprint: FC365F95 E24D4B55 81347510 10FFE331
  Last enrollment status: Granted
  Next enrollment attempt:
    01:58:25 PST Feb 14 2004
    * A new key will be generated *
    * Configuration will not be saved after enrollment *
  State:
    Keys generated ..... Yes (General Purpose)
    Issuing CA authenticated ..... Yes
    Certificate request(s) ..... Yes
```

Table 1 describes the significant fields shown in the display.

Table 1 show crypto pki trustpoints Field Descriptions

Field	Description
Trustpoint	Name of the trustpoint.
Issuing CA certificate pending	The certificate authority (CA) certificate is being retrieved (query mode).
Issuing CA certificate [not] configured	A CA certificate is [not] configured.
Subject Name	Subject name of the indicated certificate.
Next query attempt	Time until the next query attempt (query mode).
Router certificate pending/Router [key usage] certificate pending	The trustpoint is attempting to obtain the certificate from the CA server (through query mode or enrollment).
Router [key usage] certificate configured	Certificate of the specified key usage is configured.
Requested Subject Name	Subject name used in the enrollment request (Public Key Cryptography Standards 10 [PKCS10]).
Fingerprint MD5/SHA1	Fingerprint of the indicated certificate (Message Digest 5 [MD5] or Secure Hash Algorithm 1 [SHA]1).
Request Fingerprint MD5/SHA1	Fingerprint of the PKCS10 enrollment request (MD5/SHA1).
Enrollment polling: [polled] times ([remaining] left)/Next poll: in seconds	Number of Simple Certificate Enrollment Protocol (SCEP) polling attempts that have been made and that remain before the router gives up/Time until the next polling attempt.
Last enrollment status: Pending/Granted/Rejected/Failed	Last enrollment attempt status (pending, granted, rejected, or failed).
Next enrollment attempt: <i>time</i> (Optional) A new key will be generated. (Optional) Configuration will not be saved after enrollment.	The trustpoint is configured to do auto-enrollment and the auto-enrollment will happen at <i>time</i> . (Optional) The trustpoint is configured to generate a new key when auto-enrollment occurs. (Optional) The running configuration is “dirty,” so the configuration will not be saved automatically after autoenrollment.
State	Current state of the trustpoint.
Keys generated	“Yes or No” and the key usage (General Purpose or Signature, Encryption).
Issuing CA authenticated	“Yes or No” if crypto CA authentication has been done successfully.
Certificate request(s)	Progress of current enrollment: “Pending,” “Yes,” (complete), or “None” (not in progress).

Related Commands

Command	Description
crypto pki trustpoint	Declares the CA that your router should use.

Glossary

certificate—Digital representation of user or device attributes, including a public key, that is signed with an authoritative private key.

certificate authority (CA)—A service responsible for managing certificate requests and issuing certificates to participating IPsec network devices. This service provides centralized key management for the participating devices and is explicitly entrusted by the receiver to validate identities and to create digital certificates.

enrollment—The process of obtaining a new certificate from a CA.

identity CA—A CA that issues a certificate verifying the identity of a router. An identity CA can be a root CA (and have a self-signed certificate), or it can have a chain of certificates that validate each CA between itself and the root CA. An identity CA uses its own certificate to sign the certificate of a router, thereby validating the identity of the router.

IP Security (IPsec)—A framework of open standards developed by the Internet Engineering Task Force (IETF). IPsec provides security for transmission of sensitive information over unprotected networks such as the Internet. IPsec acts at the network layer, protecting and authenticating IP packets between participating IPsec devices (“peers”), such as Cisco routers.

peer certificate—The certificate presented by a peer, which contains the peer’s public key and is signed by the peer’s identity CA.

public key infrastructure (PKI)—Provides trusted and efficient key and certificate management to support security protocols such as IPsec.

registration authority (RA)—A server that acts as a proxy for the CA so that CA functions can continue when the CA is offline.

Secure Sockets Layer (SSL)—An application-level protocol that enables secure transactions of data through privacy, authentication, and data integrity. It relies upon certificates, public keys, and private keys.

trustpoint—One or more identities that are considered trustworthy and can be used to validate other identities.

X.509—A widely-used standard for defining digital certificates for security purposes.



Note

Refer to [Internetworking Terms and Acronyms](#) for terms not included in this glossary.

CCVP, the Cisco logo, and Welcome to the Human Network are trademarks of Cisco Systems, Inc.; Changing the Way We Work, Live, Play, and Learn is a service mark of Cisco Systems, Inc.; and Access Registrar, Aironet, Catalyst, CCDA, CCDP, CCIE, CCIP, CCNA, CCNP, CCSP, Cisco, the Cisco Certified Internetwork Expert logo, Cisco IOS, Cisco Press, Cisco Systems, Cisco Systems Capital, the Cisco Systems logo, Cisco Unity, Enterprise/Solver, EtherChannel, EtherFast, EtherSwitch, Fast Step, Follow Me Browsing, FormShare, GigaDrive, HomeLink, Internet Quotient, IOS, iPhone, IP/TV, iQ Expertise, the iQ logo, iQ Net Readiness Scorecard, iQuick Study, LightStream, Linksys, MeetingPlace, MGX, Networkers, Networking Academy, Network Registrar, PIX, ProConnect, ScriptShare, SMARTnet, StackWise, The Fastest Way to Increase Your Internet Quotient, and TransPath are registered trademarks of Cisco Systems, Inc. and/or its affiliates in the United States and certain other countries.

All other trademarks mentioned in this document or Website are the property of their respective owners. The use of the word partner does not imply a partnership relationship between Cisco and any other company. (0711R)

Copyright © 2005 Cisco Systems, Inc. All rights reserved.