crypto pki authenticate through cws whitelisting

crypto pki authenticate

To authenticate the certification authority (CA) (by getting the certificate of the CA), use the **crypto pki authenticate** command in global configuration mode.

crypto pki authenticate *name*

Syntax Description

name	The name of the CA. This is the same name used when the CA was declared with the **crypto ca identity** command .

Command Default

No default behavior or values.

Command Modes

Global configuration

Command History

Release	Modification
11.3T	The **crypto ca authenticate** command was introduced.
12.3(7)T	This command replaced the **crypto ca authenticate** command.
12.2(18)SXE	This command was integrated into Cisco IOS Release 12.2(18)SXE.
12.2(33)SRA	This command was integrated into Cisco IOS Release 12.2(33)SRA.
12.4(24)T	Support for IPv6 Secure Neighbor Discovery (SeND) was added.

Usage Guidelines

This command is required when you initially configure CA support at your router.

This command authenticates the CA to your router by obtaining the self-signed certificate of the CA that contains the public key of the CA. Because the CA signs its own certificate, you should manually authenticate the public key of the CA by contacting the CA administrator when you enter this command.

If you are using Router Advertisements (RA) mode (using the **enrollment**command) when you issue the **crypto pki authenticate** command, then registration authority signing and encryption certificates will be returned from the CA and the CA certificate.

This command is not saved to the router configuration. However. the public keys embedded in the received CA (and RA) certificates are saved to the configuration as part of the Rivest, Shamir, and Adelman (RSA) public key record (called the “RSA public key chain”).

Note If the CA does not respond by a timeout period after this command is issued, the terminal control will be returned so that it remains available. If this happens, you must reenter the command. Cisco IOS software will not recognize CA certificate expiration dates set for beyond the year 2049. If the validity period of the CA certificate is set to expire after the year 2049, the following error message will be displayed when authentication with the CA server is attempted: error retrieving certificate :incomplete chain If you receive an error message similar to this one, check the expiration date of your CA certificate. If the expiration date of your CA certificate is set after the year 2049, you must reduce the expiration date by a year or more.

Examples

In the following example, the router requests the certificate of the CA. The CA sends its certificate and the router prompts the administrator to verify the certificate of the CA by checking the CA certificate's fingerprint. The CA administrator can also view the CA certificate's fingerprint, so you should compare what the CA administrator sees to what the router displays on the screen. If the fingerprint on the router's screen matches the fingerprint viewed by the CA administrator, you should accept the certificate as valid.

```
Router(config)#
crypto pki authenticate myca
Certificate has the following attributes:
Fingerprint: 0123 4567 89AB CDEF 0123
Do you accept this certificate? [yes/no] y#
```

Related Commands

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

crypto pki benchmark

To start or stop benchmarking data for Public Key Infrastructure (PKI) performance monitoring and optimization, use the **crypto pki benchmark**command in privileged EXEC mode.

crypto pki benchmark {**start** *limit* [**wrap**] | **stop**}

Syntax Description

start *limit*	Enables PKI benchmarking. The *limit* argument states the number of records from 0 to 9990 that can be stored for the benchmarking session. A limit of 0 indicates an unlimited number of records can be stored.
wrap	(Optional) Specifies a continuous flow of records. Once the maximum number of records is gathered, they are released and a new set of records is generated. If the **wrap** keyword is not specified, then benchmarking stops once the limit for the maximum number of records has been reached.
stop	Terminates PKI benchmarking data collection.

Command Modes

Privileged EXEC (#)

Command History

Release	Modification
15.1(3)T	This command was introduced.

Usage Guidelines

Use the **crypto pki benchmark start**command to start the collection of PKI benchmarking performance monitoring and optimization data. Use the **crypto pki benchmark stop**command to stop the collection of the PKI benchmarking performance monitoring and optimization data.

Use the **show crypto pki benchmarks command to view the collection data.**

Use the **clear crypto pki benchmarks command to clear the**PKI benchmarking performance monitoring and optimization data and release all memory associated with this data.

The IOS PKI Performance Monitoring and Optimization feature enables you to collect the following types of PKI performance data:

- Time to validate entire certificate chain.
- Time to verify each certificate.
- Time to check revocation status for each certificate.
- Time to fetch certificate revocation list (CRL) database for each fetch location.
- Time to fetch Simple Certificate Enrollment Protocol (SCEP) method capabilities to retrieve the CRL.
- Time to process each CRL.
- Time to process the Online Certificate Status Protocol (OCSP) response. OCSP is a certificate revocation mechanism.
- Time to fetch Authentication, Authorization, and Accounting (AAA).

- CRL size.
- Validation result.
- Validation Bypass (pubkey cached).
- Method used to fetch a CRL.
- PKI session identifier.
- Crypto engine used (hardware, software, etoken).

Examples

The following example starts PKI benchmarking data and collects 20 records. Once 20 records are collected, they are released and a new set of 20 records is generated.

```
Router# crypto pki benchmark start 20 wrap
```

Related Commands

Command	Description
clear crypto pki benchmarks	Clears PKI benchmarking performance monitoring and optimization data and releases all memory associated with this data.
show crypto pki benchmarks	Displays benchmarking data for PKI performance monitoring and optimization that was collected.

crypto pki cert validate

To determine if a trustpoint has been successfully authenticated, a certificate has been requested and granted, and if the certificate is currently valid, use the **crypto pki cert validate** command in global configuration mode.

crypto pki cert validate *trustpoint*

Syntax Description

trustpoint	The trustpoint to be validated.

Command Default

No default behavior or values.

Command Modes

Global configuration

Command History

Release	Modification
12.3(8)T	This command was introduced. Also, effective with Cisco IOS Release 12.3(8)T, this command replaced the **crypto ca cert validate** command.
12.2(18)SXE	This command was integrated into Cisco IOS Release 12.2(18)SXE.
12.2(33)SRA	This command was integrated into Cisco IOS Release 12.2(33)SRA.

Usage Guidelines

The **crypto pki cert validate** command validates the router's own certificate for a given trustpoint. Use this command as a sanity check after enrollment to verify that the trustpoint is properly authenticated, a certificate has been requested and granted for the trustpoint, and that the certificate is currently valid. A certificate is valid if it is signed by the trustpoint certification authority (CA), not expired, and so on.

Examples

The following examples show the possible output from the **crypto pki cert validate** command:

```
Router(config)# crypto pki cert validate ka
Validation Failed: trustpoint not found for ka
Router(config)# crypto pki cert validate ka
Validation Failed: can't get local certificate chain
Router(config)# crypto pki cert validate ka
Certificate chain has 2 certificates.
Certificate chain for ka is valid
Router(config)# crypto pki cert validate ka
Certificate chain has 2 certificates.
Validation Error: no certs on chain
Router(config)# crypto pki cert validate ka
Certificate chain has 2 certificates.
Validation Error: unspecified error
```

Related Commands

Command	Description
crypto pki trustpoint	Declares the certification authority that the router should use.
show crypto pki trustpoints	Displays the trustpoints that are configured in the router.

crypto pki certificate chain

To enter the certificate chain configuration mode, use the **crypto pki certificate chain**command in global configuration mode.

crypto pki certificate chain *name*

Syntax Description

name	Specifies the name of the certificate authority (CA). The name must match that which was declared for the CA using the **crypto pki trustpoint**command.

Command Default

No default behavior or values.

Command Modes

Global configuration

Command History

Release	Modification
11.3 T	The **crypto ca certificate chain** command was introduced.
12.3(7)T	This command replaced the **crypto ca certificate chain** command.
12.2(18)SXE	This command was integrated into Cisco IOS Release 12.2(18)SXE.
12.4(2)T	The command output was modified to distinguish the current active certificate and the rollover certificate in the certificate chain.
12.2(33)SRA	This command was integrated into Cisco IOS Release 12.2(33)SRA.

Usage Guidelines

This command puts you into certificate chain configuration mode. When you are in certificate chain configuration mode, you can delete certificates using the **certificate** command.

You need to be in certificate chain configuration mode to delete certificates.

Examples

The following example deletes the router's certificate. In this example, the router had a general-purpose RSA key pair with one corresponding certificate. The show command is used to determine the serial number of the certificate to be deleted.

```
Router# show crypto pki certificates
Certificate
  Subject Name
    Name: myrouter.example.com
    IP Address: 10.0.0.1
  Status: Available
  Certificate Serial Number: 0123456789ABCDEF0123456789ABCDEF
  Key Usage: General Purpose

CA Certificate
  Status: Available
  Certificate Serial Number: 3051DF7123BEE31B8341DFE4B3A338E5F
  Key Usage: Not Set
Router# configure terminal
Router(config)# crypto pki certificate chain myca
```

```
Router(config-cert-chain)# no certificate 0123456789ABCDEF0123456789ABCDEF
% Are you sure you want to remove the certificate [yes/no]? yes
% Be sure to ask the CA administrator to revoke this certificate.
Router(config-cert-chain)# exit
```

The following example shows a certificate chain with an active CA certificate and a shadow, or rollover, certificate:

```
Router# configure terminal
Router(config)# crypto pki certificate chain myca
certificate 06
certificate ca 01
certificate rollover 0B
 ! This is the peer's shadow PKI certificate.
certificate rollover ca 0A
! This is the CA shadow PKI certificate
```

This example shows how the certificate chain is rewritten when rollover actually happens:

```
Router# configure terminal

Router(config)# crypto pki certificate chain myca
 certificate 0B
 certificate ca 0A
```

Related Commands

Command	Description
certificate	Adds certificates manually.

crypto pki certificate map

To define certificate-based access control lists (ACLs), use the **crypto pki certificate map** command in ca-certificate-map configuration mode. To remove the certificate-based ACLs, use the no form of this command.

crypto pki certificate map *label sequence-number*
no crypto pki certificate map *label sequence-number*

Syntax Description

label	A user-specified label that is referenced within the **crypto pki trustpoint** command.
sequence-number	A number that orders the ACLs with the same label. ACLs with the same label are processed from lowest to highest sequence number. When an ACL is matched, processing stops with a successful result.

Command Default

None

Command Modes

Ca-certificate-map configuration (ca-certificate-map)

Command History

Release	Modification
12.2(15)T	The **crypto ca certificate map** command was introduced.
12.3(7)T	This command replaced the **crypto ca certificate map** command.
12.2(18)SXD	This command was integrated into Cisco IOS Release 12.2(18)SXD.
12.4(9)T	The **serial-number** field name was introduced.
Cisco IOS XE Release 2.4	This command was implemented on the Cisco ASR 1000 series routers.

Usage Guidelines

Issuing this command places the router in ca-certificate-map configuration mode where you can specify several certificate fields together with their matching criteria. The general form of these fields is as follows:

```
field-name match-criteria match-value
```

The *field-name* field in the above example is one of the certificate fields. Field names are similar to the names used in the ITU-T X.509 standard. The *field-name* is a special field that matches any subject name or related name field in the certificate, such as the **alt-subject-name, subject-name**, and **unstructured-subject-name** fields.

- **alt-subject-name** -- Case-insensitive string.
- **expires-on** --Date field in the format dd mm yyyy hh:mm:ss or mmm dd yyyy hh:mm:ss.
- **issuer-name** -- Case-insensitive string.
- **name** -- Case-insensitive string.
- serial-number--Case-insensitive string.
- **subject-name** --Case-insensitive string.

- **unstructured-subject-name** -- Case-insensitive string.
- **valid-start** --Date field in the format dd MM. yyy hh:mm:ss or mmm dd yyyy hh:mm:ss.

Note For Yang environment, the date and time format for both the **expires-on** date and **valid-start-** field follow the same format. The string UTC should always be appended to the date and time as in yang environment the time is only accepted as Universal Time, Coordinated (UTC).

- **expires-on** -- Case sensitive string. Date field in the format mmm dd yyyy hh:mm:ss UTC.
- **valid-start** -- Case sensitive string. Date field in the format mmm dd yyyy hh:mm:ss UTC.

Note The time portion is optional in both the **expires-on** date and **valid-start** field and defaults to 00:00:00 if not specified. The time is interpreted according to the time zone offset configured for the router. The string **utc** can be appended to the date and time when they are configured as Universal Time, Coordinated (UTC) rather than local time.

The *match-criteria* field in the example is one of the following logical operators:

- **eq** --equal (valid for name and date fields)
- **ne** --not equal (valid for name and date fields)
- **co** --contains (valid only for name fields)
- **nc** --does not contain (valid only for name fields)
- **lt** --less than (valid only for date fields)
- **ge** --greater than or equal to (valid only for date fields)

The *match-value* field is a case-insensitive string or a date.

Examples

The following example shows how to configure a certificate-based ACL that will allow any certificate issued by Company to an entity within the company.com domain. The label is Company, and the sequence is 10.

```
crypto pki certificate map Company 10
 issuer-name co Company
 unstructured-subject-name co company.com
```

The following example accepts any certificate issued by Company for an entity with DIAL or organizationUnit component ou=WAN. This certificate-based ACL consists of two separate ACLs tied together with the common label Group. Because the check for DIAL has a lower sequence number, it is performed first. Note that the string “DIAL” can occur anywhere in the subjectName field of the certificate, but the string WAN must be in the organizationUnit component.

```
crypto pki certificate map Group 10
 issuer-name co Company
 subject-name co DIAL
```

```
crypto pki certificate map Group 20
 issuer-name co Company
 subject-name co ou=WAN
```

Case is ignored in string comparisons; therefore, DIAL in the previous example will match dial, DIAL, Dial, and so on. Also note that the component identifiers (o=, ou=, cn=, and so on) are not required unless it is desirable that the string to be matched occurs in a specific component of the name. (Refer to the ITU-T security standards for more information about certificate fields and components such as ou=.)

If a component identifier is specified in the match string, the exact string, including the component identifier, must appear in the certificate. This requirement can present a problem if more than one component identifier is included in the match string. For example, “ou=WAN,o=Company” will not match a certificate with the string “ou=WAN,ou=Engineering,o=Company” because the “ou=Engineering” string separates the two desired component identifiers.

To match both “ou=WAN” and “o=Company” in a certificate while ignoring other component identifiers, you could use this certificate map:

```
crypto pki certificate map Group 10
 subject-name co ou=WAN
 subject-name co o=Company
```

Any space character proceeding or following the equal sign (=) character in component identifiers is ignored. Therefore “o=Company” in the proceeding example will match “o = Company,” “o =Company,” and so on.

The following example shows a CA map file used to certificate serial number session control:

```
crypto pki trustpoint CA1
  enrollment url http://CA1
  ip-address FastEthernet0/0
  crl query ldap://CA1_ldap
  revocation-check crl
  match certificate crl-map1
  crypto pki certificate map crl-map1 1
  serial-number ne 489d
```

Related Commands

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

crypto pki certificate query (ca-trustpoint)

To specify that certificates should not be stored locally but retrieved from a certification authority (CA) trustpoint, use the **crypto pki certificate query** command in ca-trustpoint configuration mode. To cause certificates to be stored locally per trustpoint, use the no form of this command.

crypto pki certificate query
no crypto pki certificate query

Syntax Description

This command has no arguments or keywords.

Command Default

CA trustpoints are stored locally in the router's NVRAM.

Command Modes

Ca-trustpoint configuration

Command History

Release	Modification
12.2(8)T	The **crypto ca certificate query (ca-trustpoint)** command was introduced.
12.3(7)T	This command replaced the **crypto ca certificate query (ca-trustpoint)** command.
12.2(18)SXD	This command was integrated into Cisco IOS Release 12.2(18)SXD.
12.2(18)SXE	This command was integrated into Cisco IOS Release 12.2(18)SXE.
12.2(33)SRA	This command was integrated into Cisco IOS Release 12.2(33)SRA.

Usage Guidelines

Normally, certain certificates are stored locally in the router's NVRAM, and each certificate uses a moderate amount of memory. To save NVRAM space, you can use this command to put the router into query mode, preventing certificates from being stored locally; instead, they are retrieved from a specified CA trustpoint when needed. This will save NVRAM space but could result in a slight performance impact.

The **crypto pki certificate query** command is a subcommand for each trustpoint; thus, this command can be disabled on a per-trustpoint basis.

Before you can configure this command, you must enable the **crypto pki trustpoint**command , which puts you in ca-trustpoint configuration mode.

Note This command deprecates the **crypto ca certificate query**command in global configuration mode. Although you can still enter the global configuration command, the configuration mode and command will be written back as ca-trustpoint.

Examples

The following example shows how to prevent certificates and certificate revocation lists (CRLs) from being stored locally on the router; instead, they are retrieved from the "ka" trustpoint when needed.

```
crypto pki trustpoint ka
```

```
.
.
.
crypto pki certificate query
```

Related Commands

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

crypto pki certificate storage

To specify the local storage location for public key infrastructure (PKI) credentials, use the **crypto pki certificate storage**command in global configuration mode. To restore the default behavior, that is to store PKI credentials to NVRAM, use the no form of this command.

crypto pki certificate storage *location-name*
no crypto pki certificate storage

Syntax Description

location-name	Name of the local storage device. • Default is NVRAM.

Command Default

NVRAM is the default local storage location if this command is not issued.

Command Modes

Global configuration (config)

Command History

Release	Modification
12.4(2)T	This command was introduced.
12.2(33)SRA	This command was integrated into Cisco IOS Release 12.2(33)SRA.
12.2(33)SXH	This command was integrated into Cisco IOS Release 12.2(33)SXH.

Usage Guidelines

All Cisco platforms support NVRAM and flash local storage. Depending on your platform, you may have other supported local storage options including bootflash, slot, disk, USB flash, or USB token.

During run time, you can specify what active local storage device you would like to use to store PKI credentials. You must have the following system requirements before you can specify PKI credentials local storage location:

- A Cisco IOS Release 12.4(2)T PKI-enabled image or a later image
- A platform that supports storing PKI credentials as separate files
- A configuration that contains at least one certificate
- An accessible local file system

When using a local storage device to store PKI data, the following restrictions are applicable:

- Only local file systems may be used. An error message will be displayed if a remote file system is selected, and the command will not take effect.
- A subdirectory may be specified if supported by the local file system. NVRAM does not support subdirectories.
- Settings will take effect only when the running configuration is saved to the startup configuration.

If the keys are generated on the etoken, then the default storage loaction for the certificates is the etoken

for the device certificates. The CA certificates are stored in NVRAM. This allows for the credentials(keysa nd certificates) to be stored together on the removable media by default.

Examples

The following configuration example shows how to store certificates to the certs subdirectory. The certs subdirectory does not exist and is automatically created.

```
Router# dir nvram:
 114  -rw-        4687                    <no date>  startup-config
 115  ----        5545                    <no date>  private-config
 116  -rw-        4687                    <no date>  underlying-config
   1  ----          34                    <no date>  persistent-data
   3  -rw-         707                    <no date>  ioscaroot#7401CA.cer
   9  -rw-         863                    <no date>  msca-root#826E.cer
  10  -rw-         759                    <no date>  msca-root#1BA8CA.cer
  11  -rw-         863                    <no date>  msca-root#75B8.cer
  24  -rw-        1149                    <no date>  storagename#6500CA.cer
  26  -rw-         863                    <no date>  msca-root#83EE.cer
129016 bytes total (92108 bytes free)
Router# config terminal
Enter configuration commands, one per line.  End with CNTL/Z.
Router(config)# crypto pki certificate storage disk0:/certs
Requested directory does not exist -- created
Certificates will be stored in disk0:/certs/
Router(config)# end
Router# write
*May 27 02:09:00:%SYS-5-CONFIG_I:Configured from console by consolemem
Building configuration...
[OK]
Router# directory disk0:/certs
Directory of disk0:/certs/
  14  -rw-         707  May 27 2005 02:09:02 +00:00  ioscaroot#7401CA.cer
  15  -rw-         863  May 27 2005 02:09:02 +00:00  msca-root#826E.cer
  16  -rw-         759  May 27 2005 02:09:02 +00:00  msca-root#1BA8CA.cer
  17  -rw-         863  May 27 2005 02:09:02 +00:00  msca-root#75B8.cer
  18  -rw-        1149  May 27 2005 02:09:02 +00:00  storagename#6500CA.cer
  19  -rw-         863  May 27 2005 02:09:02 +00:00  msca-root#83EE.cer
47894528 bytes total (20934656 bytes free)
! The certificate files are now on disk0/certs:
```

Related Commands

Command	Description
show crypto pki certificates storage	Displays the current PKI certificate storage location.

crypto pki crl cache

To set the maximum amount of volatile memory used to cache certificate revocation lists (CRLs), use the **crypto pki crl cache**command in privileged EXEC mode. To restore the default value, use the **no** form of this command.

crypto pki crl cache *cache-size*
no crypto pki crl cache *cache-size*

Syntax Description

cache-size	The maximum CRL cache size in kilobytes. • The default value is 512 kilobytes. The value specified must be an integer. Specifying a cache size of zero disables CRL caching.

Command Default

The default CRL cache size is set to 512 kilobytes.

Command Modes

Privileged EXEC (#)

Command History

Release	Modification
12.4(20)T	This command was introduced.
Cisco IOS XE Release 2.4	This command was implemented on the Cisco ASR 1000 series routers.

Usage Guidelines

The CRL cache is a global cache that holds all CRLs downloaded by the router regardless of the trustpoint configuration. The impact on router memory depends upon the CRL cache size configured by the administrator. Configuring the CRL cache size allows the amount of memory used for the CRL cache to be reduced (for instance, if low memory conditions exist) or to be increased for better performance (for instance, when a large number of CRLs are being processed).

If the **crypto pki crl cache**command is issued, regardless of the CRL cache size value set, the CRL cache size will be included in the configuration. Issuing the **no crypto pki crl cache**commandwill remove the CRL cache size from the configuration.

When a CRL is stored in the CRL cache, it is condensed at least one-fifth of its original size. Therefore, more CRLs can be stored in the CRL cache than would be expected based on the CRL size before being cached.

Note To configure CRL caching for a given trustpoint, you may issue either the **crl-cache none** or **crl cache delete-after** command. To disable caching of CRLs for a given trustpoint, use the **crl-cache none** command. To set a maximum age for CRLs in the cache for a given trustpoint, use the **crl cache delete-after** command.

Examples

The following example sets the maximum CRL cache size to 2048 kilobytes and then shows sample output of the **show crypto pki crls**command:

```
Router# crypto pki crl cache 2048
```

```
Router# show crypto pki crls
     CRL Issuer Name:
           cn=ioscs,l=Anytown,c=US
           LastUpdate: 02:53:41 GMT Mar 6 2007
           NextUpdate: 02:53:41 GMT Mar 13 2007
           Retrieved from CRL Distribution Point:
                ** CDP Not Published - Retrieved via SCEP
     CRL DER is 475 bytes
     CRL is stored in parsed CRL cache
     Parsed CRL cache current size is 1705 bytes
     Parsed CRL cache maximum size is 2048 bytes
```

Related Commands

Command	Description
crl cache delete-after	Deletes a CRL from the cache after the specified number of minutes.
crl cache none	Disables caching of all CRLs.
crypto pki crl request	Requests that a new CRL be obtained immediately from the CA.
show crypto pki crls	Displays the current CRL on the router.

crypto pki crl request

To request that a new certificate revocation list (CRL) be obtained immediately from the certification authority, use the **crypto pki crl request**command in global configuration mode.

crypto pki crl request *name*

Syntax Description

name	Specifies the name of the CA. This is the same name used when the CA was declared with the **crypto pki trustpoint** command.

Command Default

Normally, the router requests a new CRL when it is verifying a certificate and there is no CRL cached.

Command Modes

Global configuration

Command History

Release	Modification
11.3 T	The **crypto ca crl request**command was introduced.
12.3(7)T	This command replaced the **crypto ca crl request** command.
12.2(18)SXE	This command was integrated into Cisco IOS Release 12.2(18)SXE.
12.2(33)SRA	This command was integrated into Cisco IOS Release 12.2(33)SRA.

Usage Guidelines

A CRL lists all the certificates of the network device that have been revoked. Revoked certificates will not be honored by your router; therefore, any IPSec device with a revoked certificate cannot exchange IP Security traffic with your router.

The first time your router receives a certificate from a peer, it will download a CRL from the CA. Your router then checks the CRL to make sure the certificate of the peer has not been revoked. (If the certificate appears on the CRL, it will not accept the certificate and will not authenticate the peer.)

A CRL can be reused with subsequent certificates until the CRL expires. If your router receives the certificate of a peer after the applicable CRL has expired, it will download the new CRL.

If your router has a CRL which has not yet expired, but you suspect that the contents of the CRL are out of date, use the **crypto pki crl request** command to request that the latest CRL be immediately downloaded to replace the old CRL.

This command is not saved to the configuration.

Note This command should be used only after the trustpoint is enrolled.

Examples

The following example immediately downloads the latest CRL to your router:

```
crypto pki crl request
```

crypto pki enroll

To obtain the certificates 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

name	The name of the CA. Use the same name as when you declared the CA using the **crypto pki trustpoint**command.

Command Default

No default behavior or values.

Command Modes

Global configuration (config)

Command History

Release	Modification
11.3T	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.
12.2(33)SRA	This command was integrated into Cisco IOS Release 12.2(33)SRA.
12.2SX	This command is supported in the Cisco IOS Release 12.2SX train. Support in a specific 12.2SX release of this train depends on your feature set, platform, and platform hardware.
12.4(24)T	Support for IPv6 Secure Neighbor Discovery (SeND) was added.

Usage Guidelines

This command requests certificates from the CA for all of your router's Rivest, Shamir, and Adelmen (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 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 certificates, you must reissue the command.

Note If you are using a Secure Shell (SSH) service, you should set up specific RSA key pairs (different private keys) for the trustpoint and the SSH service. (If the Public Key Infrastructure [PKI] and the SSH infrastructure share the same default RSA key pair, a temporary disruption of SSH service could occur. The RSA key pair could become invalid or change because of the CA system, in which case you would not be able to log in using SSH. You could receive the following error message: "key changed, possible security problem.")

Responding to Prompts

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

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 certificates. 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 your router's serial number should be included in the obtained certificate. The serial number is not used by IP Security (IPsec) 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. 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, which 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 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 in 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
crypto map local address	Specifies and names an identifying interface to be used by the crypto map for IPsec traffic.
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 export pem

To export a certificate and Rivest, Shamir, and Adleman (RSA) key pair that is associated with a trustpoint in a privacy-enhanced mail (PEM)-formatted file, use the **crypto pki export pem** command in global configuration mode. To remove any of the configured parameters, use the **no** form of this command.

crypto pki export *trustpoint* **pem** {**terminal** | **url** *destination-url*} {**3des** | **des**} **password** *password-phrase* [**rollover**]
no crypto pki export *trustpoint* **pem** {**terminal** | **url** *destination-url*} {**3des** | **des**} **password** *password-phrase* [**rollover**]

Syntax Description

trustpoint	Name of the trustpoint that the associated certificate and RSA key pair exports. The *trustpoint* argument must match the name that was specified through the **crypto pki trustpoint** command.
terminal	Specifies the certificate and RSA key pair that is displayed in PEM format on the console terminal.
url *destination-url*	Specifies the URL of the file system where your router should export the certificate and RSA key pairs.
3des	(Optional) Exports the trustpoint using the Triple Data Encryption Standard (3DES) encryption algorithm.
des	(Optional) Exports the trustpoint using the DES encryption algorithm.
password-phrase	Specifies the encrypted password phrase that is used to encrypt the PEM file for export. **Note** The password phrase can be any phrase that is at least eight characters in length; it can include spaces and punctuation, excluding the question mark (?), which has special meaning to the Cisco IOS parser.
rollover	(Optional) Exports certificate authority (CA) shadow, or rollover, certificate.

Command Default

Certificates and RSA keys are not exported.

Command Modes

Global configuration (config)

Command History

Release	Modification
12.3(4)T	The **crypto ca export pem** command was introduced.
12.3(7)T	This command was introduced. This command replaced the **crypto ca export pem** command.
12.2(18)SXE	This command was integrated into Cisco IOS Release 12.2(18)SXE.
12.4(2)T	This command was modified. The **rollover** keyword was added.

Release	Modification
12.2(33)SRA	This command was integrated into Cisco IOS Release 12.2(33)SRA.
15.2(1)T	This command was modified. Support was added in the CLI for hiding the password in an exported PEM-formatted file with the introduction of the **password** keyword followed by the *password-phrase* argument.

Usage Guidelines

Note Security threats, as well as the cryptographic technologies to help protect against them, are constantly changing. For more information about the latest Cisco cryptographic recommendations, see the Next Generation Encryption (NGE) white paper.

The **crypto pki export pem** command allows you to export certificate and RSA key pairs in PEM-formatted files. The PEM files can then be imported back into the Cisco IOS router (via the **crypto pki import pem** command) or other public key infrastructure (PKI) applications.

The RSA keys in PEM-formatted files can be exported from the following source URL file systems:

Table 1: Destination URL File Systems from Which RSA Keys in PEM-formatted Files Are Exported

File System	Description
archive:	Exports from the archive file system.
disk0:	Exports from the disc0 file system.
disk1:	Exports from the disc1 file system.
ftp:	Exports from the FTP file system.
http:	Exports from the HTTP file system. The URL must be in the following formats: • http://*pem_location*:80, where *pem_location* is the Domain Name System (DNS) • http://*ipv4-address*:80. For example: http://10.10.10.1:80 • *http://[ipv6-address]:80*. For example: http://[2001:DB8:1:1::1]:80. The IPv6 address is in hexadecimal notation and must be encased in brackets in the URL.
https:	Exports from the HTTPS file system. The URL must use the same formats as the HTTP: file system formats.
null:	Exports from the null file system.
nvram:	Exports from the non-volatile random-access memory (NVRAM) file system.
pram:	Exports from the parameter random-access memory (PRAM) file system.
rcp:	Exports from the remote copy protocol (rcp) file system
scp:	Exports from the secure copy protocol (scp) file system.

File System	Description
snmp:	Exports from the Simple Network Management Protocol (SNMP).
system:	Exports from the system file system.
tftp:	Exports from the Trivial File Transfer Protocol (TFTP) file system. **Note** The URL must be in the form: tftp://*pem_location/file_specification*
tmpsys:	Exports from the Cisco IOS tmpsys file system.
unix:	Exports from the UNIX file system.

Examples

The following example shows how to generate and export the RSA key pair "aaa" and certificates of the router in PEM files that are associated with the trustpoint named "mycs":

```
Router(config)# crypto key generate rsa general-keys label aaa exportable

The name for the keys will be:aaa
Choose the size of the key modulus in the range of 360 to 2048 for your General Purpose
Keys. Choosing a key modulus greater than 512 may take a few minutes.
!
How many bits in the modulus [512]:
% Generating 512 bit RSA keys ...[OK]
!
Router(config)# crypto pki trustpoint mycs
Router(ca-trustpoint)# enrollment url http://mycs
Router(ca-trustpoint)# rsakeypair aaa
Router(ca-trustpoint)# exit
Router(config)# crypto pki authenticate mycs
Certificate has the following attributes:
Fingerprint:C21514AC 12815946 09F635ED FBB6CF31
% Do you accept this certificate? [yes/no]:y
Trustpoint CA certificate accepted.
!
Router(config)# crypto pki enroll mycs
%
% 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 fully-qualified domain name in the certificate will be:Router
% The subject name in the certificate will be:bizarro.cisco.com
% Include the router serial number in the subject name? [yes/no]:n
% Include an IP address in the subject name? [no]:n
Request certificate from CA? [yes/no]:y
% Certificate request sent to Certificate Authority
% The certificate request fingerprint will be displayed.
% The 'show crypto ca certificate' command will also show the fingerprint.

Router(config)# Fingerprint: 8DA777BC 08477073 A5BE2403 812DD157

00:29:11:%CRYPTO-6-CERTRET:Certificate received from Certificate Authority
```

```
Router(config)# crypto pki export aaa pem terminal 3des password cisco123
% CA certificate:
-----BEGIN CERTIFICATE-----
MIICAzCCAa2gAwIBAgIBATANBgkqhkiG9w0BAQUFADBOMQswCQYDVQQGEwJVUzES
<snip>
waDeNOSI3WlDa0AWq5DkVBkxwgn0TqIJXJOCttjHnWHK1LMcMVGn
-----END CERTIFICATE-----

% Key name:aaa
    Usage:General Purpose Key
-----BEGIN RSA PRIVATE KEY-----
Proc-Type:4,ENCRYPTED
DEK-Info:DES-EDE3-CBC,ED6B210B626BC81A

Urguv0jnjwOgowWVUQ2XR5nbzzYHI2vGLunpH/IxIsJuNjRVjbAAUpGk7VnPCT87
<snip>
kLCOtxzEv7JHc72gMku9uUlrLSnFH5slzAtoC0czfU4=
-----END RSA PRIVATE KEY-----

% Certificate:
-----BEGIN CERTIFICATE-----
MIICTjCCAfigAwIBAgICIQUwDQYJKoZIhvcNAQEFBQAwTjELMAkGA1UEBhMCVVMx
<snip>
6xlBaIsuMxnHmr89KkKkYlU6
-----END CERTIFICATE-----
```

Related Commands

Command	Description
crypto pki import pem	Imports certificates and RSA keys to a trustpoint from PEM-formatted files.
crypto pki trustpoint	Declares the CA that your router should use.
enrollment url (ca-trustpoint)	Specifies the enrollment parameters of a CA.

crypto pki export pkcs12 password

To export Rivest, Shamir, and Adleman (RSA) keys within a Public-key cryptography standards number 12 (PKCS12) file at a specified location, use the **crypto pki export pkcs12 password** command in global configuration mode. To remove any of the configured parameters, use the **no** form of this command.

crypto pki export *trustpointname* **pkcs12** *destination-url* **password** *password-phrase*
no crypto pki export *trustpointname* **pkcs12** *destination-url* **password** *password-phrase*

Syntax Description

trustpointname	Name of the trustpoint that issues the certificate that a user is going to export. When you export the PKCS12 file, the trustpoint name is the RSA key name.
destination-url	Location of the PKCS12 file to which a user wants to import the RSA key pair.
password-phrase	Password phrase that is used to encrypt the PKCS12 file for export.

Command Default

RSA keys within a PKCS12 file are not exported.

Command Modes

Global configuration

Command History

Release	Modification
12.2(15)T	The **crypto ca export pkcs12** command was introduced.
12.3(7)T	This command was introduced. This command replaced the **crypto ca export pkcs12** command.
12.2(18)SXE	This command was integrated into Cisco IOS Release 12.2(18)SXE.
12.2(33)SRA	This command was integrated into Cisco IOS Release 12.2(33)SRA.
15.2(1)T	This command was modified. Support was added in the CLI for hiding the password in an exported PKCS12-formatted file with the introduction of the **password** keyword followed by the *password-phrase* argument.

Usage Guidelines

Public-key cryptography standards were devised and published by RSA Security. A PKCS12 file has a format commonly used to store private keys with accompanying public key certificates that is protected with a password-based symmetric key. The **crypto pki export pkcs12 password** command creates a PKCS12 file that contains an RSA key pair. The PKCS12 file, along with a certificate authority (CA), is exported to the location that you specify with the destination URL. If you decide not to import the file to another router, you must delete the file.

Security Measures

Keep the PKCS12 file stored in a secure place with restricted access.

An RSA key pair is more secure than a password phrase because the private key in the key pair is not known by multiple parties. When you export an RSA key pair to a PKCS12 file, the RSA key pair now is only as secure as the password phrase.

To create a good password phrase, be sure to include numbers, as well as both lowercase and uppercase letters. Avoid publicizing the password phrase by mentioning it in e-mail or cell phone communications because the information could be accessed by an unauthorized user.

The RSA keys can be exported from the following destination URL file systems:

Table 2: Destination URL File Systems from Which RSA Keys Exported

File System	Description
archive:	Exports from the archive file system.
cns:	Exports from the cns file system. The Cisco CNS Configuration Engine is a web-based system for automatically distributing configuration files to Cisco IOS network devices.
disk0:	Exports from the disc0 file system.
disk1:	Exports from the disc1 file system.
ftp:	Exports from the FTP file system.
http:	Exports from the HTTP file system. The URL must be in the following formats: • http://*pkcs12_location*:80, where *pkcs12_location* is the Domain Name System (DNS). • http://*ipv4-address*:80. For example: http://10.10.10.1:80. • *http://[ipv6-address]:80*. For example: http://[2001:DB8:1:1::1]:80. The IPv6 address is in hexadecimal notation and must be enclosed in brackets in the URL.
https:	Exports from the HTTPS file system. The URL must use the same formats as the HTTP: file system formats.
null:	Exports from the null: file system.
nvram:	Exports from the non-volatile random-access Memory (NVRAM) file system.
pram:	Exports from the parameter random-access memory (PRAM) file system.
rcp:	Exports from the remote copy protocol (rcp) file system.
scp:	Exports from the secure copy protocol (scp) file system.
snmp:	Exports from the Simple Network Management Protocol (SNMP).
system:	Exports from the system file system.
tar:	Exports from the UNIX tar file system.
tftp:	Exports from the Trivial File Transfer Protocol (TFTP) file system. **Note** The URL must be in the form: tftp://*pkcs12_location/file_specification* .
tmpsys:	Exports from the Cisco IOS tmpsys file system.

File System	Description
unix:	Exports from the UNIX file system.
xmodem:	Exports from the Cisco xmodem file system.
ymodem:	Exports from the Cisco ymodem file system.

Examples

The following example exports an RSA key pair with a trustpoint named “mytp” to an HTTP file:

```
Router(config)# crypto pki export mytp pkcs12 http://[2001:DB8:1:1::1]:80 password myexport
 mycompany
```

Related Commands

Command	Description
crypto pki import pkcs12 password	Imports RSA keys.

crypto pki http max-buffer-size

To set the maximum http receive buffer for PKI, use the **crypto pki http <max-buffer-size>** command in the global configuration mode.

crypto pki http max-buffer-size <*max-buffer-size*>

Syntax Description

max-buffer-size	Specifies the maximum limit for the http buffer.

Command Default

No default behavior or values.

Command Modes

Global configuration (config)

Command History

Release	Modification
Cisco IOS XE Release 17.2.1	This command was introduced in the 17.2.1 release, and was later enabled in the Cisco IOS XE 16.6.8, 16.9.5, and 16.12.3 releases.

Usage Guidelines

The **crypto pki http max-buffer-size** command enables you to set the maximum http receive buffer for PKI. By default, the http max-buffer size is 10MB. You can increase this value till 100MB and reduce the value till 1MB by using this command.

It is recommended that you set the max-buffer-size only when you see the following error displayed during PKI transactions: *(debug crypto pki transaction) "CRYPTO_PKI: HTTP Payload is more than the allowed buffer size"*.

Example

```
Router(config)#crypto pki http max-buffer-size ?
  <1-100>  Specify the size in MB

Router(config)#crypto pki http max-buffer-size 9
```

crypto pki import

To import a certificate manually via TFTP or as a cut-and-paste at the terminal, use the **crypto pki import**command in global configuration mode.

crypto pki import *name* **certificate**

Syntax Description

name **certificate**	Name of the certification authority (CA). This name is the same name used when the CA was declared with the **crypto pki trustpoint** command.

Command Default

No default behavior or values

Command Modes

Global configuration

Command History

Release	Modification
12.2(13)T	The **crypto ca import** command was introduced.
12.3(7)T	This command replaced the **crypto ca import** command.
12.2(18)SXD	This command was integrated into Cisco IOS Release 12.2(18)SXD.
12.2(33)SRA	This command was integrated into Cisco IOS Release 12.2(33)SRA.
12.4(24)T	Support for IPv6 Secure Neighbor Discovery (SeND) was added.

Usage Guidelines

You must enter the **crypto pki import** command twice if usage keys (signature and encryption keys) are used. The first time the command is entered, one of the certificates is pasted into the router; the second time the command is entered, the other certificate is pasted into the router. (It does not matter which certificate is pasted first.)

Examples

The following example shows how to import a certificate via cut-and-paste. In this example, the CA trustpoint is “MS.”

```
crypto pki trustpoint MS
 enroll terminal
 crypto pki authenticate MS
!
crypto pki enroll MS
crypto pki import MS certificate
```

Related Commands

Command	Description
crypto pki trustpoint	Declares the CA that your router should use.
enrollment	Specifies the enrollment parameters of your CA.
enrollment terminal	Specifies manual cut-and-paste certificate enrollment.

crypto pki import pem

To import certificates and Rivest, Shamir, and Adleman (RSA) keys to a trustpoint from privacy-enhanced mail (PEM)-formatted files, use the **crypto pki import pem** command in global configuration mode. To remove any of the configured parameters, use the **no** form of this command.

crypto pki import *trustpoint* **pem** [{**check** | **exportable***usage-keys*}] {**terminal** | **url** *source-url*} **password** *password-phrase*
no crypto pki import *trustpoint* **pem** [{**check** | **exportable***usage-keys*}] {**terminal** | **url** *source-url*} **password** *password-phrase*

Syntax Description

trustpoint	Name of the trustpoint that is associated with the imported certificates and RSA key pairs. The *trustpoint* argument must match the name that was specified through the **crypto pki trustpoint** command.
check	(Optional) Specifies that an outdated certificate is not allowed.
exportable	(Optional) Specifies that the imported RSA key pair can be exported again to another Cisco device such as a router.
usage-keys	(Optional) Specifies that two RSA special usage key pairs are imported (that is, one encryption pair and one signature pair), instead of one general-purpose key pair.
terminal	Specifies that certificates and RSA key pairs are manually imported from the console terminal.
url *source-url*	Specifies the URL of the file system where your router should import the certificates and RSA key pairs.
password *password-phrase*	Specifies the encrypted password phrase that is used to encrypt the PEM file for import. **Note** The password phrase can be any phrase that is at least eight characters in length; it can include spaces and punctuation, excluding the question mark (?), which has special meaning to the Cisco IOS parser.

Command Default

Certificates and RSA keys are not imported.

Command Modes

Global configuration (config)

Command History

Release	Modification
12.3(4)T	The **crypto ca import pem** command was introduced.
12.3(7)T	This command was introduced. This command replaced the **crypto ca import pem** command.

Release	Modification
12.2(18)SXE	This command was integrated into Cisco IOS Release 12.2(18)SXE.
12.2(33)SRA	This command was integrated into Cisco IOS Release 12.2(33)SRA.
12.2 XN	This command was modified. The **check** keyword was added.
15.2(1)T	This command was modified. Support was added in the CLI for hiding the password in an imported PEM-formatted file with the introduction of the **password** keyword followed by the *password-phrase* argument.

Usage Guidelines

The **crypto pki import pem** command allows certificates and RSA key pairs in PEM-formatted files to be imported. The files can be previously exported from another router or generated from other public key infrastructure (PKI) applications.

The RSA keys in PEM-formatted files can be imported from the following source URL file systems:

Table 3: Source URL File Systems from Which RSA Keys in PEM-formatted Files are Imported

File System	Description
archive:	Imports from the archive file system
cns:	Imports from the CNS file system. The Cisco CNS Configuration Engine is a web-based system for automatically distributing configuration files to Cisco IOS network devices.
disk0:	Imports from the disc0 file system.
disk1:	Imports from the disc1 file system.
ftp:	Imports from the FTP file system.
http:	Imports from the HTTP file system. The URL must be in the following formats: • http://*pem_location:80*:80, where *pem_location:80* is the Domain Name System (DNS) • http://*ipv4-address*:80. For example: http://10.10.10.1:80 • *http://[ipv6-address]:80*. For example: http://[2001:DB8:1:1::1]:80. The IPv6 address is in hexadecimal notation and must be enclosed in brackets in the URL.
https:	Imports from the HTTPS file system. The URL must use the same formats as the HTTP: file system formats.
null:	Imports from the null: file system.
nvram:	Imports from the non-volatile random-access memory (NVRAM) file system.
pram:	Imports from the parameter random-access memory (PRAM) file system.
rcp:	Imports from the remote copy protocol (rcp) file system.
scp:	Imports from the secure copy protocol (scp) file system.

File System	Description
snmp:	Imports from the Simple Network Management Protocol (SNMP).
system:	Imports from the system file system.
tar:	Imports from the UNIX tar file system.
tftp:	Imports from the Trivial File Transfer Protocol (TFTP) file system. **Note** The URL must be in the form: tftp://*pem_location/file_specification*
tmpsys:	Imports from the IOS tmpsys file system.
unix:	Imports from the UNIX file system.
xmodem:	Imports from the Cisco xmodem file system.
ymodem:	Imports from the Cisco ymodem file system.

Examples

The following example shows how to import PEM files to trustpoint "ggg" through TFTP:

```
Router(config)# crypto pki import ggg pem url tftp://10.1.1.2/johndoe/msca cisco1234
% Importing CA certificate...
Address or name of remote host [10.1.1.2]?
Destination filename [johndoe/msca.ca]?
Reading file from tftp://10.1.1.2/johndoe/msca.ca
Loading johndoe/msca.ca from 10.1.1.2 (via Ethernet0):!
[OK - 1082 bytes]
% Importing private key PEM file...
Address or name of remote host [10.1.1.2]?
Destination filename [johndoe/msca.prv]?
Reading file from tftp://10.1.1.2/johndoe/msca.prv
Loading johndoe/msca.prv from 10.1.1.2 (via Ethernet0):!
[OK - 573 bytes]
% Importing  certificate PEM file...
Address or name of remote host [10.1.1.2]?
Destination filename [johndoe/msca.crt]?
Reading file from tftp://10.1.1.2/johndoe/msca.crt
Loading johndoe/msca.crt from 10.1.1.2 (via Ethernet0):!
[OK - 1289 bytes]
% PEM files import succeeded.
Router(config)#
```

Related Commands

Command	Description
crypto pki export pem	Exports certificates and RSA keys that are associated with a trustpoint in a PEM-formatted file.
crypto pki trustpoint	Declares the CA that your router should use.
enrollment url (ca-trustpoint)	Specifies the enrollment parameters of a CA.

crypto pki import pkcs12 password

To import Rivest, Shamir, and Adleman (RSA) keys, use the **crypto pki import pkcs12 password** command in global configuration mode. To remove any of the configured parameters, use the **no** form of this command.

crypto pki import *trustpointname* **pkcs12** *source-url* **password** *password-phrase*
no crypto pki import *trustpointname* **pkcs12** *source-url* **password** *password-phrase*

Syntax Description

trustpointname	Name of the trustpoint who issues the certificate that a user is going to export or import. When importing, the trustpoint name will become the RSA key name.
source-url	The location of the PKCS12 file to which a user wants to export the RSA key pair.
password *password-phrase*	Enter the password phrase that must be entered to undo encryption when the RSA keys are imported.

Command Default

No default behavior or values

Command Modes

Global configuration (config)

Command History

Release	Modification
12.2(15)T	The **crypto ca import pkcs12** command was introduced.
12.3(7)T	This command was introduced. This command replaced the **crypto ca import pkcs12** command.
12.2(18)SXE	This command was integrated into Cisco IOS Release 12.2(18)SXE.
12.2(33)SRA	This command was integrated into Cisco IOS Release 12.2(33)SRA.
15.2(1)T	This command was modified. Support was added in the CLI for hiding the password in an imported PKCS12-formatted file with the introduction of the **password** keyword followed by the *password-phrase* argument.

Usage Guidelines

When you enter the **crypto pki import pkcs12 password** command, a key pair and a trustpoint are generated.

If the key pair and trustpoint that were generated need to be removed, then enter the **crypto key zeroize rsa** command to zeroize the key pair and enter the **no crypto pki trustpoint** command to remove the trustpoint.

Note After you import RSA keys to a target router, you cannot export those keys from the target router to another router.

The RSA keys can be imported from the following source URL file systems:

Table 4: Source URL File Systems from Which RSA Keys Imported

File System	Description
archive:	Imports from the archive file system.
check	The **check** keyword is used to validate a certificate on input from a file system. Any file system argument indicated in this table can be used following this keyword.
cns:	Imports from the CNS file system. The Cisco CNS Configuration Engine is a web-based system for automatically distributing configuration files to Cisco IOS network devices.
disk0:	Imports from the disc0 file system.
disk1:	Imports from the disc1 file system.
ftp:	Imports from the FTP file system.
http:	Imports from the HTTP file system. The URL must be in the following formats: • http://*pkcs12_location*:80, where *pkcs12_location* is the Domain Name System (DNS) • http://*ipv4-address*:80. For example: http://10.10.10.1:80 • *http://[ipv6-address]:80*. For example: http://[2001:DB8:1:1::1]:80. The IPv6 address is in hexadecimal notation and must be enclosed in brackets in the URL.
https:	Imports from the HTTPS file system. The URL must use the same formats as the HTTP file system formats.
null:	Imports from the null file system.
nvram:	Imports from the non-volatile random-access memory (NVRAM) file system.
pram:	Imports from the parameter random-access memory (PRAM) file system.
rcp:	Imports from the remote copy protocol (rcp) file system.
scp:	Imports from the secure copy protocol (scp) file system.
snmp:	Imports from the Simple Network Management Protocol (SNMP).
system:	Imports from the system file system.
tar:	Imports from the UNIX tar file system.
tftp:	Imports from the Trivial File Transfer Protocol (TFTP) file system. **Note** The URL must be in the form: tftp://pkcs12_location/file_specification.
tmpsys:	Imports from the IOS tmpsys file system.
unix:	Imports from the UNIX file system.
xmodem:	Imports from the Cisco xmodem file system.

File System	Description
ymodem:	Imports from the Cisco ymodem file system.

Examples

In the following example, an RSA key pair that has been associated with the trustpoint named "mytp" is to be imported:

```
Router(config)# crypto pki import mytp pkcs12 http://[2001:DB8:1:1::1]:80 password myimport
 mycompany
```

Related Commands

Command	Description
crypto pki export pkcs12 password	Exports RSA keys.
crypto key zeroize rsa	Deletes all RSA keys from your router.
crypto pki trustpoint	Declares the CA that your router should use.

crypto pki profile enrollment

To define an enrollment profile, use the **crypto pki profile enrollment**command in global configuration mode. To delete all information associated with this enrollment profile, use the **no** form of this command.

crypto pki profile enrollment *label*
no crypto pki profile enrollment *label*

Syntax Description

label	Name for the enrollment profile; the enrollment profile name must match the name specified in the **enrollment profile** command.

Command Default

An enrollment profile does not exist.

Command Modes

Global configuration

Command History

Release	Modification
12.2(13)ZH	This command was introduced.
12.3(4)T	This command was integrated into Cisco IOS Release 12.3(4)T.
12.3(7)T	This command replaced the **crypto ca profile enrollment** command.
12.2(18)SXE	This command was integrated into Cisco IOS Release 12.2(18)SXE.
12.2(33)SRA	This command was integrated into Cisco IOS Release 12.2(33)SRA.

Usage Guidelines

Before entering this command, you must specify a named enrollment profile using the **enrollment profile** in ca-trustpoint configuration mode.

After entering the **crypto pki profile enrollment** command, you can use any of the following commands to define the profile parameters:

- **authentication command** --Specifies the HTTP command that is sent to the certification authority (CA) for authentication.
- **authentication terminal** --Specifies manual cut-and-paste certificate authentication requests.
- **authentication url** --Specifies the URL of the CA server to which to send authentication requests.
- **enrollment command** --Specifies the HTTP command that is sent to the CA for enrollment.
- **enrollment terminal** --Specifies manual cut-and-paste certificate enrollment.
- **enrollment url** --Specifies the URL of the CA server to which to send enrollment requests.
- **parameter** --Specifies parameters for an enrollment profile. This command can be used only if the **authentication command** or the **enrollment command** is used.

Note The **authentication url**, **enrollment url**, **authentication terminal**, and **enrollment terminal** commands allow you to specify different methods for certificate authentication and enrollment, such as TFTP authentication and manual enrollment.

Examples

The following example shows how to define the enrollment profile named "E" and associated profile parameters:

```
crypto pki trustpoint Entrust
  enrollment profile E
  serial
crypto pki profile enrollment E
 authentication url  http://entrust:81
 authentication command  GET /certs/cacert.der
 enrollment url  http://entrust:81/cda-cgi/clientcgi.exe
 enrollment command  POST reference_number=$P2&authcode=$P1
&retrievedAs=rawDER&action=getServerCert&pkcs10Request=$REQ
 parameter 1 value aaaa-bbbb-cccc
 parameter 2 value 5001
```

Related Commands

Command	Description
crypto pki trustpoint	Declares the PKI trustpoint that your router should use.
enrollment profile	Specifies that an enrollment profile can be used for certificate authentication and enrollment.

crypto pki server

To enable a Cisco IOS certificate server (CS) and enter certificate server configuration mode, or to immediately generate shadow certification authority (CA) credentials, use the **crypto pki server** command in global configuration mode. To disable the certificate server (which is the default functionality), use the **no** form of this command.

crypto pki server *cs-label* [**rollover** [**cancel**]]
no crypto pki server *cs-label* [**rollover** [**cancel**]]

Syntax Description

cs-label	Name of the certificate server. **Note** The certificate server name should not exceed 13 characters.
rollover	(Optional) Immediately generates a shadow CA certificate. **Note** If the **auto-enroll** command has been issued with the **regenerate** keyword, shadow keys will also be generated. **Note** If the shadow certificate and keys are already present this command will fail.
cancel	(Optional) Deletes the exiting shadow CA certificate when used with the **rollover** keyword. Shadow keys will also be deleted if they exist.

Command Default

A certificate server is not enabled; the automatic CA certificate rollover process is not initiated.

Command Modes

Global configuration (config)

Command History

Release	Modification
12.3(4)T	This command was introduced.
12.4(2)T	The **rollover** and **cancel** keywords were introduced to support automated CA certificate rollover functionality.
12.2(33)SRA	This command was integrated into Cisco IOS Release 12.2(33)SRA.

Usage Guidelines

Once the **crypto pki server** command is entered, the certificate server configuration mode commands can be configured to deploy the public key infrastructure (PKI) by defining the default behavior of the CS, which limits user interface complexity. See the Related Commands section for more information on these commands.

Note All CS-related commands are optional; therefore any basic CS functionality that is not specified through the CLI for these commands uses their default value.

- **issuer-name** -- Specifies the distinguished name (DN) as the CA issuer name for the certificate server.

- **lifetime (certificate server)** --Specifies the lifetime of the CA or a certificate.
- **lifetime crl** --Defines the lifetime of the certificate revocation list (CRL) that is used by the certificate server.
- **shutdown** --Allows a certificate server to be disabled without removing the configuration.

Automated CA Certificate Rollover

CAs and their clients, have certificates with expiration dates that have to be reissued when the current certificate is about to expire. CAs also have key pairs used to sign client certificates. When the CA certificate is expiring it must generate a new certificate and possibly a new key pair. This process, called rollover, allows for continuous operation of the network while clients and the certificate server are switching from an expiring CA certificate to a new CA certificate.

Examples

The following example shows how to enable the certificate server “mycertserver”:

```
Router(config)# ip http server
Router(config)# crypto pki server mycertserver
Router(cs-server)# database url tftp://mytftp/johndoe/mycertserver
```

The following example shows how to disable the certificate server “mycertserver”:

```
Router(config)# no crypto pki server mycertserver
% This will stop the Certificate Server process and delete the server
  configuration
Are you sure you want to do this? [yes/no]: yes
% Do you also want to remove the associated trustpoint and
  signing certificate and key? [yes/no]: no
% Certificate Server Process stopped
```

The following example shows a shadow client certificate request from a terminal:

```
Router# crypto pki server mycs rollover request pkcs10 terminal

% Enter Base64 encoded or PEM formatted PKCS10 enrollment request.
% End with a blank line or "quit" on a line by itself.
-----BEGIN CERTIFICATE REQUEST-----
MIIBUTCBuwIBADASMRAwDgYDVQQDEwdOZXdSb290MIGfMA0GCSqGSIb3DQEBAQUA
A4GNADCBiQKBgQDMHeev1ERSs320zbLQQk+3lhV/R2HpYQ/iM6uT1jkJf5iy0UPR
wF/X16yUNmG+ObiGiW9fsASF0nxZw+fO7d2X2yh1PakfvF2wbP27C/sgJNOw9uPf
sBxEc40Xe0d5FMh0YKOSAShfZYKOflnyQR2Drmm2x/33QGol5QyRvjkeWQIDAQAB
oAAwDQYJKoZIhvcNAQEEBQADgYEALM90r4d79X6vxhD0qjuYJXfBCOvv4FNyFsjr
aBS/y6CnNVYySF8UBUohXYIGTWf4I4+sj6i8gYfoFUW1/L82djS18TLrUr6wpCOs
RqfAfps7HW1e4cizOfjAUU+C7lNcobCAhwF1o6q2nIEjpQ/2yfK9O7sb3SCJZBfe
eW3tyCo=
-----END CERTIFICATE REQUEST-----
```

The following example shows the **redundancy**, **show**, and **serial-number** keywords in the **crypto pki server**command.

```
Router(config)#crypto pki server MYCA
Router(cs-server)#grant auto
Router(cs-server)#redundancy
Router(cs-server)#serial-number 0x4c
Router(cs-server)#show
 redundancy
 serial-number 0x4C
```

```
 grant auto
end
```

Related Commands

Command	Description
auto-rollover	Enables the automated CA certificate rollover functionality.
cdp-url	Specifies a CDP to be used in certificates that are issued by the certificate server.
crl (cs-server)	Specifies the CRL PKI CS.
database archive	Specifies the CA certificate and CA key archive format--and the password--to encrypt this CA certificate and CA key archive file.
database level	Controls what type of data is stored in the certificate enrollment database.
database url	Specifies the location where database entries for the CS is stored or published.
database username	Specifies the requirement of a username or password to be issued when accessing the primary database location.
default (cs-server)	Resets the value of the CS configuration command to its default.
grant auto rollover	Enables automatic granting of certificate reenrollment requests for a Cisco IOS subordinate CA server or RA mode CA.
grant auto trustpoint	Specifies the CA trustpoint of another vendor from which the Cisco IOS certificate server automatically grants certificate enrollment requests.
grant none	Specifies all certificate requests to be rejected.
grant ra-auto	Specifies that all enrollment requests from an RA be granted automatically.

Command	Description
hash (cs-server)	Specifies the cryptographic hash function the Cisco IOS certificate server uses to sign certificates issued by the CA.
issuer-name	Specifies the DN as the CA issuer name for the CS.
lifetime (cs-server)	Specifies the lifetime of the CA or a certificate.
mode ra	Enters the PKI server into RA certificate server mode.
mode sub-cs	Enters the PKI server into sub-certificate server mode
redundancy (cs-server)	Specifies that the active CS is synchronized to the standby CS.
serial-number (cs-server)	Specifies whether the router serial number should be included in the certificate request.
show (cs-server)	Displays the PKI CS configuration.
shutdown (cs-server)	Allows a CS to be disabled without removing the configuration.

crypto pki server grant

To grant all or certain simple certificate enrollment protocol (SCEP) requests, use the **crypto pki server grant**command in privileged EXEC mode.

crypto pki server *cs-label* **grant** {**all***req-id*}

Syntax Description

cs-label	Name of the certificate server. The name must match the name specified via the **crypto pki server** command.
all	All certificate enrollment requests are granted.
req-id	ID associated with a specific enrollment request in the enrollment request database. Use the **crypto pki server info requests** command to display the ID.

Command Default

If this command is not issued, the certificate server keeps the requests in a pending state.

Command Modes

Privileged EXEC

Command History

Release	Modification
12.3(4)T	This command was introduced.

Usage Guidelines

After you enable the **crypto pki server grant** command, your certificate server will immediately grant all specified certificate requests. Certificate requests that are not granted will expire after the time that was specified using the **lifetime enrollment-request** command.

Examples

The following example shows to grant all manual enrollment requests for the certificate server "mycs":

```
Router# crypto pki server mycs grant all
```

Related Commands

Command	Description
crypto pki server	Enables a Cisco IOS certificate server and enters certificate server configuration mode.
crypto pki server reject	Rejects all or certain SCEP requests.

crypto pki server info crl

Note Effective with Cisco IOS Release 12.4(20)T, the **crypto pki server info crl** command is replaced by the **show crypto pki server crl**command. See the **show crypto pki server crl** command for more information.

To display information regarding the status of the current certificate revocation list (CRL), use the **crypto pki server info crl** command in privileged EXEC mode.

crypto pki server *cs-label* **info crl**

Syntax Description

cs-label	Name of the certificate server. The name must match the name specified via the **crypto pki server** command.

Command Default

No default behavior or values

Command Modes

Privileged EXEC

Command History

Release	Modification
12.3(4)T	This command was introduced.
12.4(20)T	This command was replaced by the **show crypto pki server crl**command.

Usage Guidelines

CRLs are issued once every specified time period via the **lifetime crl** command. It is the responsibility of the network administrator to ensure that the CRL is available from the location that is specified via the **cdp-url** command. To access information, such as the lifetime and location of the CRL, use the **crypto pki server info crl**command.

Examples

The following example shows how to access CRL information for the certificate server “mycs”:

```
Router# crypto pki server mycs info crl
```

Related Commands

Command	Description
cdp-url	Specifies a CDP to be used in certificates that are issued by the certificate server.
crypto pki server	Enables a Cisco IOS certificate server and enter certificate server configuration mode.
lifetime crl	Defines the lifetime of the CRL that is used by the certificate server.

crypto pki server info requests

Note Effective with Cisco IOS Release 12.4(20)T, the **crypto pki server info requests** command is replaced by the **show crypto pki server requests**command. See the **show crypto pki server requests** command for more information.

To display all outstanding certificate enrollment requests, use the **crypto pki server info requests**command in privileged EXEC mode.

crypto pki server *cs-label* **info requests**

Syntax Description

cs-label	Name of the certificate server. The name must match the name specified via the **crypto pki server** command.

Command Default

No default behavior or values

Command Modes

Privileged EXEC

Command History

Release	Modification
12.3(4)T	This command was introduced.
12.4(2)T	The command output was modified to include shadow CA certificate information.
12.4(20)T	This command was replaced by the **show crypto pki server requests**command.

Usage Guidelines

A certificate enrollment request functions as follows:

- The certificate server receives the enrollment request from an end user, and the following actions occur:
  - A request entry is created in the enrollment request database with the initial state. (See the **show pki server** command for a complete list of certificate enrollment request states.)
  - The certificate server refers to the command-line interface (CLI) configuration (or the default behavior any time a parameter is not specified) to determine the authorization of the request. Thereafter, the state of the enrollment request is updated in the enrollment request database.
- At each Simple Certificate Enrollment Protocol (SCEP) query for a response, the certificate server examines the current request and performs one of the following actions:
  - Responds to the end user with a “pending” or “denied” state.
  - Forwards to the request to the certification authority (CA) core, where it will generate and sign the appropriate certificate, store the certificate in the enrollment request database, and return the request to the built-in certificate server SCEP server, who will reply to the end user with the certificate on the next SCEP request.

If the connection of the client has closed, the certificate server will wait for client user to request another certificate.

All enrollment requests transitions through the certificate enrollment states that are defined in the table below.

Table 5: Certificate Enrollment Request State Descriptions

Certificate Enrollment State	Description
initial	The request has been created by the SCEP server.
authorized	The certificate server has authorized the request.
malformed	The certificate server has determined that the request is invalid for cryptographic reasons.
denied	The certificate server has denied the request for policy reasons.
pending	The enrollment request must be manually accepted by the network administrator.
granted	The CA core has generated the appropriate certificate for the certificate request.

Examples

The following example shows output for the certificate server "certsrv1," which has a pending certificate enrollment request:

```
Router# crypto pki server certsrv1 info requests
   Enrollment Request Database:
   ReqID  State      Fingerprint                       SubjectName
   --------------------------------------------------------------------
   1      pending    0A71820219260E526D250ECC59857C2D serialNumber=2326115A+hostname=831.
```

The following example shows the output for shadow PKI certificate info requests:

```
Router# crypto pki server mycs info requests
Enrollment Request Database:
RA certificate requests:
  ReqID  State      Fingerprint                       SubjectName
--------------------------------------------------------------------
RA rollover certificate requests:
  ReqID  State      Fingerprint                       SubjectName
  --------------------------------------------------------------------
Router certificates requests:
  ReqID  State      Fingerprint                       SubjectName
--------------------------------------------------------------------
1      pending    A426AF07FE3A4BB69062E0E47198E5BF hostname=client
  Router rollover certificates requests:
  ReqID  State      Fingerprint                       SubjectName
  --------------------------------------------------------------------
  2      pending    B69062E0E47198E5BFA426AF07FE3A4B hostname=client
```

Related Commands

Command	Description
crypto pki server	Enables a Cisco IOS certificate server and enters PKI configuration mode.

crypto pki server password generate

To generate a password for simple certificate enrollment protocol (SCEP) requests that can be used only one time, use the **crypto pki server password generate**command in privileged EXEC mode.

crypto pki server *cs-label* **password generate** [*minutes*]

Syntax Description

cs-label	Name of the certificate server. The name must match the name specified via the **crypto pki server** command.
minutes	(Optional) Length of time, in minutes, that the password is valid. Valid times range from 1 to 1440 minutes. The default value is 60 minutes.

Command Default

If this command is not enabled, no password is created.

Command Modes

Privileged EXEC

Command History

Release	Modification
12.3(4)T	This command was introduced.

Usage Guidelines

SCEP, which is the only supported enrollment protocol, supports two client authentication mechanisms--manual and preshared key. Manual enrollment requires the administrator at the certification authority (CA) server to specifically authorize the enrollment requests; enrollment using preshared keys allows the administrator to preauthorize enrollment requests by generating a one-time password.

Note Only one password is valid at a time; if a second password is generated, the previous password is no longer valid.

Examples

The following example shows how to generate a one-time password that is valid for 75 minutes for the certificate server “mycs”:

```
Router# crypto pki server mycs password generate 75
```

Related Commands

Command	Description
crypto pki server	Enables a Cisco IOS certificate server and enters certificate server configuration mode.

crypto pki server reject

To reject all or certain Simple Certificate Enrollment Protocol (SCEP) requests, use the **crypto pki server reject** command in privileged EXEC mode.

crypto pki server *cs-label* **reject** {**all***req-id*}

Syntax Description

cs-label	Name of the certificate server. The name must match the name specified via the **crypto pki server** command.
all	All certificate enrollment requests are rejected.
req-id	ID associated with a specific enrollment request in enrollment request database. Use the **crypto pki server info requests** command to display the ID.

Command Default

If this command is not issued, the certificate server keeps the requests in a pending state.

Command Modes

Privileged EXEC

Command History

Release	Modification
12.3(4)T	This command was introduced.

Usage Guidelines

After you enable the **crypto pki server reject** command, your certificate server will immediately reject all certificate requests.

SCEP, which is the only supported enrollment protocol, supports two client authentication mechanisms--manual and preshared key. Manual enrollment requires the administrator at the certification authority (CA) server to specifically authorize the enrollment requests. The administrator can become overloaded if there are numerous enrollment requests. Thus, the **crypto pki server reject** command can be reduce user interaction by automatically rejecting all or specific enrollment requests.

Examples

The following example shows how reject all manual enrollment requests for the certificate server "mycs":

```
Router# crypto pki server mycs reject all
```

Related Commands

Command	Description
crypto pki server	Enables a Cisco IOS certificate server and enters certificate server configuration mode.
crypto pki server grant	Grants all or certain SCEP requests.
crypto pki server info requests	Displays all outstanding certificate enrollment requests.

crypto pki server remove

To remove enrollment requests that are in the certificate server Enrollment Request Database, use the **crypto pki server remove**command in privileged EXEC mode . This command does not have a **no** form.

crypto pki server *cs-label* **remove** {**all***req-id*}

Syntax Description

cs-label	Name of the certificate server.
all	Removes all enrollment requests.
req-id	Removes the specified enrollment request.

Command Default

Enrollment requests will remain in the certificate server database.

Command Modes

Privileged EXEC

Command History

Release	Modification
12.3(11)T	This command was introduced.

Usage Guidelines

After the certificate server receives an enrollment request, it can leave the request in pending, reject it, or grant it. Before this command was added, the request would be left in the Enrollment Request Database for 1 hour until the client polled the certficiate server for the result of the request. This command allows you to remove individual or all requests from the database, especially useful if the client leaves and never polls the certificate server.

In addition, the use of this command also allows the server to be returned to a clean slate with respect to the keys and transaction IDs. Thus, it is a useful command to use during troubleshooting with a Simple Certificate Enrollment Protocol (SCEP) client that may be behaving badly.

Examples

The following example shows that all enrollment requests are to be removed from the certificate server:

```
Router# enable
 Router# crypto pki server server1 remove all
```

Related Commands

Command	Description
crypto pki server info request	Displays all outstanding enrollment requests.

crypto pki server request pkcs10

To manually add a certificate request to the request database, use the **crypto pki server request pkcs10**command in privileged EXEC mode.**command** *argument* **keyword**

crypto pki server *cs-label* **request pkcs10** {*url* | **terminal**} [{**base64** | **pem** | **scep hex**}]

Syntax Description

cs-label	Name of the certificate server. The name must match the name specified via the **crypto pki server** command.
url	URL of the file systems from which the certificate server should retrieve the PKCS10 enrollment request and to which it should post the granted certificate. For a list of available options, see the table below. **Note** The request filename should have a ".req" extension and the granted certificate file name will have a ".crt" extension (see the URL example in the section "Examples" below).
terminal	Certificate requests will be manually pasted from the console terminal, and the granted certificate will be displayed on the console.
base64	(Optional) Specifies the certificate will be returned without privacy-enhanced mail (PEM) headers, regardless of whether PEM headers were used in the request.
pem	(Optional) Specifies the certificate will be returned with PEM headers automatically added to the certificate after the certificate is granted, regardless of whether PEM headers were used in the request.
scep hex	(Optional) Specifies the certificate will be returned in hexadecimal. Pending requests will also be synchronized with the standby certificate server in hexadecimal.

Command Default

None

Command Modes

Privileged EXEC (#)

Command History

Release	Modification
12.3(4)T	This command was introduced.
15.0(1)M	The command was modified to accept the PKCS10 certificate and the signing certificate in hexadecimal as well as in base64 encoding.

Usage Guidelines

Use the **crypto pki server request pkcs10** command to manually add a base64-encoded, PEM-formatted, or hexadecimal-encoded PKCS10 certificate enrollment request. This command is especially useful when the client does not have a network connection with the certificate server so that it can do Simple Certificate Enrollment Protocol (SCEP) enrollment. After the certificate is granted, the certificate will be displayed on the console terminal using base64 encoding if the **terminal** keyword is specified, or it will be sent to the file system that is specified using the *url* argument.

The *url* argument allows you to specify or change the location in which the certificate server retrieves the new certificate request and posts the granted certificate. The table below lists available file system options.

Table 6: crypto pki server request pkcs10 Options

Location	Description
cns:	Retrieves certificate from Cisco Networking Services (CNS): file system
flash:	Retrieves certificate from flash: file system
ftp:	Retrieves certificate from FTP: file system
http:	Retrieves certificate from HTTP: file system
https:	Retrieves certificate from Secure HTTP (HTTPS): file system
null:	Retrieves certificate from null: file system
nvram:	Retrieves certificate from NVRAM: file system
rcp:	Retrieves certificate from remote copy protocol (rcp): file system
scp:	Retrieves certificate from secure copy protocol (scp): file system
system:	Retrieves certificate from system: file system
tftp:	Retrieves certificate from TFTP: file system

Examples

The following example shows how to manually add a base64-encoded certificate request with PEM boundaries to the request database:

```
Router# crypto pki server mycs request pkcs10 terminal pem
% Enter Base64 encoded or PEM formatted PKCS10 enrollment request.
% End with a blank line or "quit" on a line by itself.
-----BEGIN CERTIFICATE REQUEST-----
MIIBdTCB3wIBADA2MQswCQYDVQQGEwJVUzEWMBQGA1UEChMNQ2lzY28gU3lzdGVt
czEPMA0GA1UEAxMGdGVzdCAxMIGfMA0GCSqGSIb3DQEBAQUAA4GNADCBiQKBgQDF
EFukc2lCFSHtDJn6HFR2n8rpdhlAYwcs0m68N3iRYHonv847h0/H6utTHVd2qEEo
rNw97jMRZk6BLhVDc05TKGHvUlBlHQWwc/BqpVI8WiHzZdskUH/DUM8kd67Vkjlb
e+FF7WrWT4FIO4vR4rF1V2p3FZ+A29UNc9Pi1s98nQIDAQABoAAwDQYJKoZIhvcN
AQEEBQADgYEAUQCGNzzNJwBOCwmEmG8XEGFSZWDmFlctm8VWvaZYMPOt+vl6iwFk
RmtD1Kg91Vw/qT5FJN8LmGUopOWIrwH4rUWON+TqtRmv2dgsdL5T4dx0sgG5E0s4
T302paxEHiHVRJpe8OD7FJgOvdsKRziCpyD4/Jfb1WnSVQZmvIYAxVQ=
-----END CERTIFICATE REQUEST-----

% Enrollment request pending, reqId=2

Router# crypto pki server mycs grant 2
% Granted certificate:
-----BEGIN CERTIFICATE-----
MIIB/TCCAWagAwIBAgIBAzANBgkqhkiG9w0BAQQFADAPMQ0wCwYDVQQDEwRteWNz
MB4XDTA0MDgyODAxMTcyOVoXDTA1MDgyODAxMTcyOVowNjELMAkGA1UEBhMCVVMx
FjAUBgNVBAoTDUNpc2NvIFN5c3RlbXMxDzANBgNVBAMTBnRlc3QgMTCBnzANBgkq
hkiG9w0BAQEFAAOBjQAwgYkCgYEAxRBbpHNpQhUh7QyZ+hxUdp/K6XYZQGMHLNJu
vDd4kWB6J7/OO4dPx+rrUx1XdqhBKKzcPe4zEWZOgS4VQ3NOUyhh71JQZR0FsHPw
aqVSPFoh82XbJFB/w1DPJHeu1ZI5W3vhRe1q1k+BSDuL0eKxdVdqdxWfgNvVDXPT
4tbPfJ0CAwEAAaNCMEAwHwYDVR0jBBgwFoAUggWpVwokbUtGIwGZGavh6C8Bq6Uw
```

```
HQYDVR0OBBYEFFD3jZ/d960qzCGKwKNtFvq85Xt6MA0GCSqGSIb3DQEBBAUAA4GB
AAE4MqerwbM/nO8BCyZAiDzTqwLGnNvzS4H+u3JCsm0LaxY+E3d8NbSY+HruXWaR
7QyjprDGFd9bftRoqGYuiQkupU13sIHEyf3C2KnXJB6imySvAiuaQrGdSuUSIhBO
Xfh/xdWo3XL1e3vtWiYUa4X6jPUMpn74HoNfB4/gHO7g
-----END CERTIFICATE-----
```

The following example shows how to retrieve a certificate request and add it to the request database (using the *url* argument):

Note The request file name should have a ".req" extension and the certificate file name a ".crt" extension.

```
Router# crypto pki server mycs request pkcs10 tftp://192.0.2.129/router5
% Retrieving Base64 encoded or PEM formatted PKCS10 enrollment request...
Reading file from tftp://192.0.2.129/router5.req
Loading router5.req from 192.0.2.129 (via Ethernet0): !
[OK - 582 bytes]
% Enrollment request pending, reqId=1
Router# crypto pki server mycs grant 1
% Writing out the granted certificate...
!Writing file to tftp://192.0.2.129/router5.crt!
```

The following example shows how to manually add a hexadecimal-encoded certificate request with PEM boundaries to the request database:

```
Router# crypto pki server mycs request pkcs10
 scep hex 0C4A3A2CA5C2E66DDCD740A4259759E2 5811E7CB133BAC936EF48C6187F4AD22 3
PKCS10 request in hex
Enter the PKCS10 in hexidecimal representation....
Router(config-pubkey)#3082010E 3081B902 0100301D 311B3019 06092A86 4886F70D 01090216 0C697073
Router(config-pubkey)#6563662D 33383435 61305C30 0D06092A 864886F7 0D010101 0500034B 00304802
Router(config-pubkey)#4100B660 EF764AD6 A896E03E 0D1A1A16 5450857C 9B2CC04E B61719E5 2216CBF2
Router(config-pubkey)#1973B464 17E78829 22CDBD87 FBD015F1 2A0A8DD7 5396EAA1 A2A65132 912466D2
Router(config-pubkey)#62C90203 010001A0 37301406 092A8648 86F70D01 09073107 13056369 73636F30
Router(config-pubkey)#1F060A60 86480186 F8450109 08311104 0F300D30 0B060355 1D0F0404 030205A0
Router(config-pubkey)#300D0609 2A864886 F70D0101 04050003 410062A5 81B4C7F2 BDCEE03D 998BAD2B
Router(config-pubkey)#1E763461 EBB812EB 4082E2BB 273AA5DD 74FF7E12 E16035E9 4525A041 AF65E48F
Router(config-pubkey)#F0E6E13C 2646F943 5C23A634 BC50BC1F 343A
Router(config-pubkey)#30820123 3081CE02 0101300D 06092A86 4886F70D 01010405 00301D31 1B301906
Router(config-pubkey)#092A8648 86F70D01 0902160C 69707365 63662D33 38343561 301E170D 30393031
Router(config-pubkey)#31323032 33323039 5A170D31 39303131 30303233 3230395A 301D311B 30190609
Router(config-pubkey)#2A864886 F70D0109 02160C69 70736563 662D3338 34356130 5C300D06 092A8648
Router(config-pubkey)#6F70D01 01010500 034B0030 48024100 B660EF76 4AD6A896 E03E0D1A 1A165450
Router(config-pubkey)#857C9B2C C04EB617 19E52216 CBF21973 B46417E7 882922CD BD87FBD0 15F12A0A
Router(config-pubkey)#8DD75396 EAA1A2A6 51329124 66D262C9 02030100 01300D06 092A8648 86F70D01
Router(config-pubkey)#01040500 03410041 B2EBC44A 7F5FD26A DBAAB574 655D0C5D 84CCC7B5 48643525
Router(config-pubkey)#E85E4E06 5465A27F 6066BC8C 52AF9FF4 CE6A9C66 44441BF0 053325DC 736FD696
Router(config-pubkey)#97F8335 DDA951
Router(config-pubkey)#quit
Enter the certificate in hexidecimal representation....
Router(config-pubkey)#quit
```

Related Commands

Command	Description
crypto pki server	Enables a Cisco IOS certificate server and enters certificate server configuration mode.
crypto pki server grant	Grants all or certain SCEP requests.

Command	Description
show crypto pki server	Displays the current state and configuration of a certificate server.

crypto pki server revoke

To revoke a certificate on the basis of its serial number, use the **crypto pki server revoke**command in privileged EXEC mode.

crypto pki server *cs-label* **revoke** *certificate-serial-number*

Syntax Description

cs-label	Name of the certificate server. The name must match the name specified via the **crypto pki server** command.
certificate-serial-number	Serial number of the certificate that is to be revoked. The serial number can be a hexadecimal number with the prefix "0x" (for example, 0x4c) or a decimal number (for example, 76).

Command Default

Certificates are revoked on the basis of their name.

Command Modes

Privileged EXEC

Command History

Release	Modification
12.3(4)T	This command was introduced.
15.0(1)M	The command was modified to remove the serial-number check against the last-issued serial number.

Usage Guidelines

When a new certificate revocation list (CRL) is issued, the certificate server obtains the previous CRL, makes the appropriate changes, and resigns the new CRL. A new CRL is issued after a certificate is revoked from the CLI. If this process negatively affects router performance, the crypto pki server revoke command can be used to revoke a list or range of certificates.

Note In Cisco IOS Release 15.0(1)M, the serial number to be revoked is not compared with the last-issued serial number.

Note A new CRL cannot be issued unless the current CRL is revoked or changed.

Examples

The following examples show how to revoke a certificate with the serial number 76 (for example, 0x4c in hexidecimal) from the certificate server "mycs":

```
Router# crypto pki server mycs revoke 76
Router# crypto pki server mycs revoke 0x4c
```

Related Commands

Command	Description
cdp-url	Specifies that CDP should be used in the certificates that are issued by the certificate server.
crypto pki server	Enables a Cisco IOS certificate server and enters certificate server configuration mode.

crypto pki server start

To enable a Cisco IOS certificate server, use the **crypto pki server start** command in privileged EXEC mode. To disable a certificate server, use the **crypto pki server stop** command.

crypto pki server *servername* **start**

Syntax Description

servername	Name of the certificate server. **Note** The certificate server name must not exceed 13 characters.

Command Default

The certificate server is disabled.

Command Modes

Privileged EXEC (#)

Command History

Release	Modification
15.0(1)M	This command was introduced.

Usage Guidelines

Using the **crypto pki server start** command is the same as using the **no shut** command in DSP configuration mode.

Examples

The following example shows how to enable a certificate server on a router:

```
Router# crypto pki server MYCA start
%Some server settings cannot be changed after CA certificate generation.
% Please enter a passphrase to protect the private key
% or type Return to exit
Password:
Re-enter password:
% Certificate Server enabled.
```

Related Commands

Command	Description
crypto pki server stop	Disables a Cisco IOS certificate server.
show crypto pki server	Displays the current state and configuration of a certificate server.

crypto pki server stop

To disable a Cisco IOS certificate server, use the **crypto pki server stop** command in privileged EXEC mode.

crypto pki server *servername* **stop**

Syntax Description

servername	Name of the certificate server.

Command Modes

Privileged EXEC (#)

Command History

Release	Modification
15.0(1)M	This command was introduced.

Usage Guidelines

Using the **crypto pki server stop** command is the same as using the **shutdown** command in DSP configuration mode.

Examples

The following example shows how to disable a certificate server:

```
Router# crypto pki server MYCA stop
Certificate server 'shut' event has been queued for processing.
```

Related Commands

Command	Description
crypto pki server start	Enables a Cisco IOS certificate server.
show crypto pki server	Displays the current state and configuration of a certificate server.

crypto pki server trim

To trim certificates from the certificate revocation list (CRL), use the **crypto pki server trim**command in privileged EXEC mode.

crypto pki server [*cs-label*] **trim** {**expired** [*start-number* [*end-number*] [**verbose**]] | **generate expired-list** [*start-number end-number*] [**url** *url*] | **url** *url* [**verbose**]}

Syntax Description

cs-label	Name of the certificate server. The name must match the name specified using the **crypto pki server** command.
expired	Specifies that the expired certificates are to be trimmed from the CRL.
start-number	The beginning of the certificate serial number range to check and trim from the CRL if the certificate has expired.
end-number	(Optional) The ending number of the certificate serial number range to check and trim from the CRL if the certificate has expired.
verbose	Displays information about the action taken on the certificates checked in the CRL.
generate	Generates information about CRL trimming.
expired-list	Generates information about trimmed expired certificates.
url *url*	Specifies the location of the expired certificate list, which contains a list of certificate serial numbers to be trimmed from the CRL.

Command Default

All certificates in the specified certificate server database will be searched to locate and to trim expired certificates.

Command Modes

Privileged EXEC (#)

Command History

Release	Modification
12.4(20)T	This command was introduced.
15.0(1)M	This command was modified in a release earlier than Cisco IOS Release 15.0(1)M. The **generate** keyword was added.

Usage Guidelines

This command trims expired certificates from the CRL. Only certificates that are expired and have accurate and complete information in the certificate database can be trimmed from the database.

Depending on the size and location of the certificate database, searching the database for expired certificates may be a time-consuming process. Depending on your environment, you may choose one of three methods to search and to trim your CRL:

- Search the entire certificate database.

This is usually the most time-consuming and resource-consuming method.

- Specify a range of certificate serial numbers to search.

If a large number of certificates are in your certificate database or if your certificate database is stored at a remote location (for example, TFTP or Secure Copy [SCP]) you may limit the range of certificates to search by specifying both the starting and ending certificate serial numbers. If no starting and ending certificate serial numbers are specified, the entire certificate database will be searched and all expired certificates will be trimmed.

- Use an input list to specify the expired certificates to be trimmed from the CRL.

This is the most scalable method because it divides the process into two steps: searching the certificate database for expired certificates and trimming the CRL. An input file listing expired certificate serial numbers may be generated using a Perl script or similar program, manually, or by issuing the **crypto pki server trim generate expired-list** command. The input list must follow the format as shown:

```
# CRL Trimming file generated on 01/31/2008
version=1
35
37
```

Lines that begin with a pound sign (#) are inserted comments. The second line contains a version string indicating the file type. Each remaining line (in this example lines 35 and 37) contains a certificate serial number indicating one certificate to be removed from the CRL.

Examples

The following example shows how to check and trim the CRL of all expired certificates in the certificate database for the certificate server “mycs”:

```
Router#
crypto pki server mycs trim expired
```

The following example shows how to check and trim the CRL of expired certificates within the certificate serial number range 0x1-0x3 in the certificate database for the certificate server “mycs”. The result is the same as generating and using an input file of expired certificate serial numbers, as shown in the next example.

```
Router# crypto pki server mycs trim expired 0x1 end 0x3
```

The following example shows how to generate a list of expired certificate serial numbers, store the list on an HTTP server, then use the resulting list to trim the CRL of all expired certificates for the certificate server “mycs”:

```
Router# crypto pki server mycs trim generate expired-list 0x1 0x3 url
http://databaselocation/expired-certs.1st
Router# crypto pki server mycs trim url http://databaselocation/expired-certs.1st
The following example shows how to check and trim the CRL for only one certificate serial
number in the certificate database for the certificate server "mycs." If the certificate
with the serial number 45 has expired, it will be trimmed from the CRL.
Router# crypto pki server mycs trim expired 0x2
The following example shows how to trim the CRL of all expired certificates for the
certificate server "mycs" and display the resulting action taken for each certificate serial
 number:
Router#
crypto pki server mycs trim expired verbose
Certificate 2: Expired. Removed from CRL.
Certificate F4240: Expired. Removed from CRL.
```

```
Certificate 4593: Not Removed.
Certificate 1234: Not Removed.
```

Related Commands

Command	Description
crypto pki server	Enables a Cisco IOS certificate server and enters certificate server configuration mode.
crypto pki server trim generate expired-list	Generates a list of expired certificates in the CRL.

crypto pki server trim generate expired-list

To generate a list of expired certificates in the current certificate revocation list (CRL), use the **crypto pki server trim generate expired-list**command in privileged EXEC mode.

crypto pki server *cs-label* **trim generate expired-list** [**start** *number* **end** *number*] [**url** *url*]

Syntax Description

cs-label	Name of the certificate server. The name must match the name specified via the **crypto pki server** command.
start *number*	(Optional) The first certificate serial number from which to begin searching the CRL for expired certificates. To locate expired certificates within a range both the starting certificate serial number and the ending certificate serial number must be specified.
end *number*	(Optional) The last certificate serial number that will be checked when searching the CRL for a range of expired certificates.
url *url*	(Optional) Specifies the location where the resulting list of expired certificates will be stored.

Command Default

All certificates in the specified certificate server database will be searched to locate expired certificates.

Command Modes

Privileged EXEC (#)

Command History

Release	Modification
12.4(20)T	This command was introduced.

Usage Guidelines

This command generates a list of expired certificates that are in the CRL for the specified certificate server. The resulting list of expired certificates may be used as input to the **crypto pki server trim** command to remove the listed certificates from the CRL resulting in trimming, or revoking, the expired certificates.

Only certificates that have accurate and complete information in the certificate database can be automatically added to the list of expired certificates and later trimmed from the database. Only CRL entries for expired certificates can be trimmed.

If there are a large number of certificates in your certificate database or if your certificate database is stored at a remote location, for example TFTP or SCP, you may limit the range of certificates to search by specifying both the starting and ending certificate serial numbers. If no starting and ending certificate serial numbers are specified, the entire certificate database will be searched and all expired certificates will be added to the expired certificates list.

A URL may be specified to save the list of expired certificates to a specified location. If no URL is specified, the list of expired certificates will be printed on your terminal. The list may then be cut and pasted to a file.

Examples

```
The following example shows both how to generate a list of expired certificates within the
 certificate serial number range 34-38 in the certificate database for the certificate
server "mycs" and how to save the resulting list to an HTTP location:
```

```
Router#
crypto pki server mycs trim generate expired-list start 34 end 38 url
http://databaselocation/expired-certs.1st
```

The following example shows the resulting list of expired certificates in the file expired-certs.1st:

```
# CRL Trimming file generated on 01/31/2008
version=1
35
37
```

Lines that begin with a pound sign (#) are inserted comments. The second line contains a version string indicating the file type. Each remaining line, in this example lines 35 and 37, contains a certificate serial number indicating one certificate to be removed from the CRL.

Related Commands

Command	Description
crypto pki server	Enables a Cisco IOS certificate server and enters certificate server configuration mode.
crypto pki server trim	Trims certificates from the certificate revocation list.

crypto pki server unrevoke

To recover a revoked certificate, that is to remove a certificate from the certificate revocation list (CRL), use the **crypto pki server unrevoke**command in privileged EXEC mode.

crypto pki server *cs-label* **unrevoke** *certificate-serial-number*

Syntax Description

cs-label	Name of the certificate server. The name must match the name specified via the **crypto pki server** command.
certificate-serial-number	Serial number of the certificate that is to be recovered. The serial number can be a hexadecimal number with the prefix "0x" (for example, 0x4c) or a decimal number (for example, 76).

Command Default

None.

Command Modes

Privileged EXEC (#)

Command History

Release	Modification
12.4(20)T	This command was introduced.

Usage Guidelines

If a certificate is erroneously revoked, either the client has to reenroll in the PKI or the administrator may recover the revoked certificate by issuing the **crypto pki server unrevoke** command. This command removes a certificate, specified by its serial number, from the CRL. The CRL is then resigned and can be republished.

Examples

The following examples show how to unrevoke a certificate with the serial number 76, or 0x4c in hexidecimal, from the certificate server "mycs":

```
Router# crypto pki server mycs unrevoke 76
Router# crypto pki server mycs unrevoke 0x4c
```

Related Commands

Command	Description
crypto pki server	Enables a Cisco IOS certificate server and enters certificate server configuration mode.
cyrpto pki server revoke	Revokes a certificate based on its serial number.

crypto pki token change-pin

To change the user PIN on the USB eToken, use the **crypto pki token change-pin** command in privileged EXEC mode.

crypto pki token *token-name* [**admin**] **change-pin** [*pin*]

Syntax Description

token-name	Name of USB token specified via the **crypto pki token login**command.
admin	(Optional) The router will change the administrative PIN on the USB token. If this keyword is not issued, the router will change the user pin.
pin	(Optional) User PIN required to access the etoken.

Command Default

None

Command Modes

Privileged EXEC

Command History

Release	Modification
12.3(14)T	This command was introduced.
12.4(11)T	This command was integrated into the Cisco 7200VXR NPE-G2 platform.
Cisco IOS XE Release 3.6	This command was integrated into Cisco IOS XE Release 3.6.

Usage Guidelines

If you want to change the administrative PIN on the token, you must be logged into the eToken as an admin via the **crypto pki token admin login**command.

After the user PIN has been changed, you must reset the login failure count to zero (via the **crypto pki token max-retries** command). The maximum number of allowable login failures is set (by default) to 15.

Examples

The following example shows that the user PIN was changed to 1234:

```
crypto pki token usbtoken0 admin login 5678
crypto pki token usbtoken0 change-pin 1234
```

Related Commands

Command	Description
crypto pki token login	Logs into the USB eToken.
crypto pki token max-retries	Sets the maximum number of allowed failed login attempts.

crypto pki token encrypted-user-pin

To encrypt a USB token PIN that is stored in private NVRAM, use the **crypto pki token encrypted-user-pin** command in global configuration mode. To decrypt the token's PIN, use the **no** form of this command.

crypto pki token {*token-name* | **default**} **encrypted-user-pin** [**write**] [**passphrase** *passphrase*]
no crypto pki token {*token-name* | **default**} **encrypted-user-pin** [**write**] [**passphrase** *passphrase*]

Syntax Description

token-name	Name of the token that will have its PIN encrypted.
default	Configures default values for tokens.
write	**(Optional) Writes to memory immediately after the passphrase is entered. This keyword saves the running configuration to NVRAM.**
passphrase *passphrase*	(Optional) Enables noninteractive command-line interface (CLI). If you do not issue this keyword, you will automatically be prompted for the passphrase. **Tip** Noninteractive CLI is provided for instances where users will not be responding to prompts, for example in scripts, configuration tools, or other automated processes. If you are issuing this command from the console, it is recommended that you use the interactive CLI to help protect against observation from unauthorized persons.

Command Default

The PIN stored in private NVRAM is not encrypted.

Command Modes

Global configuration (config)

Command History

Release	Modification
12.4(4)T	This command was introduced.
12.4(11)T	This command was integrated into the Cisco IOS Release 12.4(11)T and implemented on 7200VXR NPE-G2 platform.
15.0(1)M	This command was modified earlier than Cisco IOS Release 15.0(1)M. The **default** keyword was added.
Cisco IOS XE Release 3.6	This command was integrated into Cisco IOS XE Release 3.6.

Usage Guidelines

After the token's PIN is encrypted with the **crypto pki token encrypted-user-pin**command, no action is taken when you insert the token into the router. The user must log in to the router and enter the passphrase to decrypt the PIN before the router can use the PIN to log in to the token.

After the PIN has been successfully decrypted, the router will execute the configuration commands from the token at privilege level 15.

Tip It is recommended that you create a passphrase different from the token's PIN. Also, the user should log in to the token as a "normal user" (a privilege level 1 user), so the user cannot access commands that can alter the configuration of the router.

Examples

The following example shows the configuration of a user PIN and the encryption of that user PIN:

```
! Configure the user PIN.
Router(config)#
crypto pki token usbtoken0: user-pin
Enter password:
!
! Now, the user PIN can be encrypted.
!
Router(config)#
crypto pki token usbtoken0: encrypted-user-pin
 Enter passphrase:
Router(config)#
exit
Router#
Router#
show running config
.
.
.
     crypto pki token usbtoken0 user-pin *encrypted*
.
.
.
```

Related Commands

Command	Description
crypto pki token user-pin	Creates a PIN that automatically allows the router to log in to the USB token at router startup.
privilege	Configures a new privilege level for users and associates commands with that privilege level.

crypto pki token label

To set or change the name of a USB token label, use the **crypto pki token label** command in global configuration mode.

crypto pki token *device* **: label** *token-label*

Syntax Description

device:	Location or name of the USB device.
token-label	Specifies the label, or name, of the USB token. • *token-label* may be up to 31 alphanumeric characters in length, including dashes and underscores.

Command Default

No label is set. The USB token is known by its factory name.

Command Modes

Global configuration

Command History

Release	Modification
12.4(4)T	This command was introduced.
12.4(11)T	This command was integrated into the Cisco 7200VXR NPE-G2 platform.
Cisco IOS XE Release 3.6	This command was integrated into Cisco IOS XE Release 3.6.

Usage Guidelines

After you have logged in your USB token to the router, you may want to change the factory default label. Changing the default factory name to a unique name is useful when configuring multiple USB tokens for automatic login, secondary configuration files, or other token specific settings.

Note Either the device name or label may be used to specify the USB token. If using the device name, it is followed by a colon, ":".

Examples

The following example shows how to change the USB token label from the "oldlabel" to "newlabel" after the token has been logged in. The router will not use the "newlabel" until the next time the token is inserted or the router is reloaded:

```
Router#
Router# configure terminal
Router(config)# crypto pki token oldlabel label newlabel
Token label changed.
```

Related Commands

Command	Description
crypto pki token user-pin	Creates a PIN that automatically allows the router to log into the USB token at router startup.

crypto pki token lock

To lock the token, use the **crypto pki token lock**command in privileged EXEC mode.

crypto pki token *token-name* **lock** [**user-pin**] [**passphrase** *passphrase*]

Syntax Description

token-name	Name of the token that is to be locked.
user-pin	(Optional) Specifies the USB token PIN if set.
passphrase *passphrase*	(Optional) Enables the noninteractive command-line interface (CLI). If you do not issue this keyword, you will automatically be prompted for the passphrase. **Tip** The noninteractive CLI is provided for instances where users will not be responding to prompts, for example in scripts, configuration tools, or other automated processes. If you are issuing this command from the console, it is recommended that you use the interactive CLI to help protect against observation from unauthorized persons.

Command Default

The token is not locked.

Command Modes

Privileged EXEC

Command History

Release	Modification
12.4(4)T	This command was introduced.
12.4(11)T	This command was integrated into the Cisco 7200VXR NPE-G2 platform.
Cisco IOS XE Release 3.6	This command was integrated into Cisco IOS XE Release 3.6.

Usage Guidelines

After you have locked a token with the **crypto pki token lock** command, all Rivest, Shamir, and Adelman (RSA) keys that have been loaded from the token will be deleted and, if configured, the secondary “unconfig” file will run with full privileges.

Examples

The following example shows how to reload a router, unlock the PIN, and then lock the PIN again:

Router> **enable**

Password:

Router# **crypto pki token usbtoken0: unlock**

Token eToken is usbtoken0

Enter passphrase:

Token login to usbtoken0(eToken) successful

Router#

Sep 20 22:31:13.128: %CRYPTO-6-TOKENLOGIN: Cryptographic Token eToken

```
Login Successful
```

Router# **crypto pki token usbtoken0: lock**

Related Commands

Command	Description
crypto pki token name secondary unconfig file	Specifies a secondary “unconfig” file.
crypto pki token unlock	Unlocks the token and decrypts the PIN that is stored in private NVRAM.

crypto pki token login

To log into the USB eToken, use the **crypto pki token login** command in privileged EXEC mode.

crypto pki token *token-name* [**admin**] **login** [*pin*]

Syntax Description

token-name	Name of USB eToken.
admin	(Optional) The router will attempt to log into the token as an administrator. If this keyword is not issued, the router will attempt to log into the token as a user. **Note** If you want to change the PIN via the **crypto pki token change-pin** command, you must issue this keyword.
pin	(Optional) User PIN required to access the token. If a user PIN is not specified, the default PIN, 1234567890, is used.

Command Default

None

Command Modes

Privileged EXEC

Command History

Release	Modification
12.3(14)T	This command was introduced.
12.4(11)T	This command was integrated into the Cisco 7200VXR NPE-G2 platform.
Cisco IOS XE Release 3.6	This command was integrated into Cisco IOS XE Release 3.6.

Usage Guidelines

This command allows you to manually log into a USB eToken. To automatically log into an eToken, issue the **crypto pki token user-pin** command, which allows you to create a PIN for automatic login.

Examples

The following example shows how to log into the USB eToken manually:

```
crypto pki token usbtoken0:login 1234567890
```

Related Commands

Command	Description
crypto pki token logout	Logs the router out of the USB eToken.

crypto pki token logout

To log the router out of the USB eToken, use the **crypto pki token logout** command in privileged EXEC mode.

crypto pki token *token-name* **logout**

Syntax Description

token-name	Name of USB eToken specified via the **crypto pki token login**command.

Command Default

None

Command Modes

Privileged EXEC

Command History

Release	Modification
12.3(14)T	This command was introduced.
12.4(11)T	This command was integrated into the Cisco 7200VXR NPE-G2 platform.
Cisco IOS XE Release 3.6	This command was integrated into Cisco IOS XE Release 3.6.

Usage Guidelines

If you want to save any data to the USB eToken, you must log back into the eToken.

Examples

The following example shows how to successfully log out of a USB eToken:

```
crypto pki token usbtoken0:logout
Token eToken is usbtoken0
Token logout from usbtoken0(eToken) successful
*Jan 28 05:46:59.544:%CRYPTO-6-TOKENLOGOUT:Cryptographic Token eToken Logout Successful
```

Related Commands

Command	Description
crypto pki token login	Logs into the USB eToken.

crypto pki token max-retries

To set the maximum number of allowed failed login attempts, use the **crypto pki token max-retries**command in global configuration mode. To return to the default functionality (which is 15 failed login attempts), use the **no** form of this command.

crypto pki token {*token-name* | **default**} **max-retries** [*number*]
no crypto pki token {*token-name* | **default**} **max-retries** [*number*]

Syntax Description

token-name	Name of USB token that the router will log into.
default	Default value is to be used.
number	(Optional) Number of consecutive failed login attempts the router will allow before locking out the user. Available range: 0 to 15. Default value is 15.

Command Default

15 failed login attempts are allowed

Command Modes

Global configuration

Command History

Release	Modification
12.3(14)T	This command was introduced.
12.4(11)T	This command was integrated into the Cisco 7200VXR NPE-G2 platform.
Cisco IOS XE Release 3.6	This command was integrated into Cisco IOS XE Release 3.6.

Usage Guidelines

After the user PIN is changed via the **crypto pki token c hange-pin command**, the login failure count is automatically reset to 15; however, it is recommended that the login failure count be set to zero.

Examples

The following example shows how to change the allowed maximum number of failed login attempts to 20:

```
crypto pki token usbtoken0 max-retries 20
```

Related Commands

Command	Description
crypto pki token c hange-pin	Changes the user PIN number on the USB eToken.
crypto pki token login	Logs into the USB eToken.

crypto pki token removal timeout

To set the time interval that the router waits before removing the Rivest, Shamir, and Adelman (RSA) keys that are stored in the eToken, use the **crypto pki token removal timeout**command in global configuration mode. To return to the default functionality (which is no timeout), use the **no** form of this command.

crypto pki token {*token-name* | **default**} **removal timeout** [*seconds*]
no crypto pki token {*token-name* | **default**} **removal timeout** [*seconds*]

Syntax Description

token-name	Name of USB eToken that is being removed from the router.
default	Default value, which is automatic RSA key removal, is to be used.
seconds	(Optional) Time interval, in seconds, that the router waits before removing the RSA keys and tearing down IP Security (IPSec) tunnels associated with the specified eToken. Available range: 0 to 480. **Note** If a time interval is not specified, all RSA keys and associated tunnels are immediately torn down after the eToken is removed from the router.

Command Default

The default timeout is zero, which causes the RSA keys to be removed automatically after the eToken is removed from the router . The default appears in the running configuration as:

```
crypto pki token default removal timeout 0
```

Command Modes

Global configuration

Command History

Release	Modification
12.3(14)T	This command was introduced.
12.4(11)T	This command was integrated into the Cisco 7200VXR NPE-G2 platform.
Cisco IOS XE Release 3.6	This command was integrated into Cisco IOS XE Release 3.6.

Usage Guidelines

After the eToken is removed from the router, you can clear from your router any RSA keys that were obtained from the eToken; all IPSec tunnels that used those RSA keys for authentication are also torn down. Both the keys and tunnels are immediately cleared unless otherwise specified via the **crypto pki token removal timeout** command.

Although the RSA keys remain on the eToken, they can only be accessed with the correct PIN. Too many unsuccessful attempts to log into the eToken will disable the PIN and any further login attempts will be refused.

Note The **no** version of this command does not remove RSA keys from the router. To immediately remove RSA keys from the router, set the timeout value to zero.

Examples

The following example shows how to set the time that the router will wait before removing the RSA keys that are stored in the eToken after the eToken has been removed from the router:

```
crypto pki token usbtoken0 removal timeout 60
```

Related Commands

Command	Description
crypto pki token logout	Logs the router out of the USB token.
crypto pki token max-retries	Sets the maximum number of allowed failed login attempts.

crypto pki token secondary config

To merge a specified file with the running configuration after the eToken is logged in to the router, use the **crypto pki token secondary config** command in global configuration mode. To remove the specified file, use **no** form of the command.

crypto pki token {*token-name* | **default**} **secondary config** [*file*]
no crypto pki token {*token-name* | **default**} **secondary config** [*file*]

Syntax Description

token-name	Name of USB eToken that will have its running configuration merged with the secondary configuration file.
default	Sets the default values for tokens.
file	(Optional) Name of the file that will be merged with the running configuration. **Note** The filename is relative to the eToken, so the name should not include a device name such as "usbtoken0:."

Command Default

A secondary configuration file does not exist.

Command Modes

Global configuration (config)

Command History

Release	Modification
12.3(14)T	This command was introduced.
12.4(11)T	This command was integrated into the Cisco 7200VXR NPE-G2 platform.
15.0(1)M	This command was modified earlier than Cisco IOS Release 15.0(1)M. The **default** keyword was added.
Cisco IOS XE Release 3.6	This command was integrated into Cisco IOS XE Release 3.6.

Usage Guidelines

Use the **crypto pki token secondary config** command if you want to merge, not overwrite, a file with the running configuration on the router. The secondary configuration is processed after the eToken is logged in to the router.

Examples

The following example shows how to merge the secondary configuration file "CONFIG1.CFG" with the current running configuration:

```
Router# configure terminal
Router(config)# crypto pki token default secondary config CONFIG1.CFG
```

Related Commands

Command	Description
crypto pki token login	Logs in to the USB eToken.

Command	Description
crypto pki token user-pin	Creates a PIN that automatically allows the router to log into the USB eToken at router startup.

crypto pki token secondary unconfig

To specify a secondary "unconfig" file and its location for a USB token, use the **crypto pki token secondary unconfig** command in global configuration mode. To remove secondary configuration elements from the running configuration, use the **no** form of this command.

crypto pki token {*token-name* | **default**} **secondary unconfig** [*file*]
no crypto pki token {*token-name* | **default**} **secondary unconfig** [*file*]

Syntax Description

token-name	Name of the token that is to be unlocked.
default	Configures default values for tokens.
file	(Optional) Name and location of the secondary configuration file.

Command Default

Secondary "unconfig" file will not be processed.

Command Modes

Global configuration (config)

Command History

Release	Modification
12.4(4)T	This command was introduced.
12.4(11)T	This command was integrated into the Cisco 7200VXR NPE-G2 platform.
15.0(1)M	This command was modified earlier than Cisco IOS Release 15.0(1)M. The **default** keyword was added.
Cisco IOS XE Release 3.6	This command was integrated into Cisco IOS XE Release 3.6.

Usage Guidelines

Configuration files that exist on a USB token are called secondary configuration files. If you create and configure a secondary configuration file, it is executed after the token is logged in. The existence of a secondary configuration file is determined by the presence of a secondary configuration file option in the Cisco IOS configuration stored in NVRAM.

When the token is removed, logged out, or the removal timer (if set) expires, a separate "unconfig" file is processed to remove all secondary configuration elements from the running configuration. Secondary configuration and secondary "unconfig" files are executed at privilege level 15 and are not dependent on the level of the user logged in.

Examples

The following example shows a how a secondary "unconfig" file might be used to remove secondary configuration elements from the running config. For example, a secondary configuration file might be used to set up a public key infrastructure (PKI) trustpoint. A corresponding "unconfig" file, named mysecondaryunconfigfile.cfg, might contain the following command:

no crypto pki trustpoint token-tp

If the token were removed and the following commands executed, the trustpoint and associated certificates would be removed from the router's running configuration:

```
Router#
configure terminal
Router(config)# no crypto pki token mytoken secondary unconfig mysecondaryunconfigfile.cfg
```

Related Commands

Command	Description
crypto pki token secondary config	Merges a specified secondary configuration file with the running configuration after the USB token is logged in to the router.
crypto pki token user-pin	Creates a PIN that automatically allows the router to log in to the USB token at router startup.

crypto pki token unlock

To unlock the token and decrypt the PIN that is stored in private NVRAM, use the **crypto pki token unlock** command in privileged EXEC mode.

crypto pki token *token-name* **unlock** [**user-pin**] [**passphrase** *passphrase*]

Syntax Description

token-name	Name of the token that is to be unlocked.
user-pin	(Optional) Specifies the USB token PIN if set.
passphrase *passphrase*	(Optional) Enables the noninteractive command-line interface (CLI). If you do not issue this keyword, you will automatically be prompted for the passphrase. **Tip** The noninteractive CLI is provided for instances where users will not be responding to prompts, for example in scripts, configuration tools, or other automated processes. **Note** If you are issuing this command from the console, it is recommended that you use the interactive CLI to help protect against observation from unauthorized persons.

Command Default

USB token is not unlocked, or decrypted.

Command Modes

Privileged EXEC

Command History

Release	Modification
12.4(4)T	This command was introduced.
12.4(11)T	This command was integrated into the Cisco 7200VXR NPE-G2 platform.
Cisco IOS XE Release 3.6	This command was integrated into Cisco IOS XE Release 3.6.

Usage Guidelines

After you unlock a token via the **crypto pki token unlock** command, the Cisco IOS software will treat the token as if it is automatically logged into the router. Any Rivest, Shamir, and Adelman (RSA) keys on the token are loaded onto the router and the secondary configuration file on the token is executed (if a secondary configuration file has been configured by the user). Secondary configuration files are executed with full user privileges.

Examples

The following example shows the configuration and encryption of a user PIN and then that the router is reloading and the user PIN is being unlocked.

! Configuring the user PIN

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

Router(config)# **crypto pki token usbtoken0: user-pin**

```
Enter password:
! Encrypt the user PIN
Router (config)# crypto pki token usbtoken0: encrypted-user-pin
Enter passphrase:
Router(config)# exit
Router#
Sep 20 21:51:38.076: %SYS-5-CONFIG_I: Configured from console by console
Router# show running-config
crypto pki token usbtoken0 user-pin *encrypted*
! Reloading the router.
Router> enable
Password:
! Decrypting the user pin.
Router# crypto pki token usbtoken0: unlock
Token eToken is usbtoken0
Enter passphrase:
Token login to usbtoken0(eToken) successful
Router#
Sep 20 22:31:13.128: %CRYPTO-6-TOKENLOGIN: Cryptographic Token eToken
Login Successful
```

Related Commands

Command	Description
crypto pki token user-pin	Creates a PIN that automatically allows the router to log into the USB token at router startup.

crypto pki token user-pin

To create a PIN that automatically allows the router to log in to the USB eToken at router startup, use the **crypto pki token user-pin** command in global configuration mode. To remove the stored PIN from the configuration, use t he **no** form of this command.

crypto pki token {*token-name* | **default**} **user-pin** [*pin*] [*token-pin*]
no crypto pki token {*token-name* | **default**} **user-pin** [*pin*] [*token-pin*]

Syntax Description

token-name	Name of USB eToken that the router will log in to.
default	Sets the default values for tokens.
user-pin	Specifies the PIN to access token.
pin	(Optional) User PIN required to log in to the eToken. The PINs are stored in private NVRAM. If a user PIN is not specified, the default PIN, 1234567890, will be used.
token-pin	(Optional) Token PIN name.

Command Default

If this command is not issued, the router cannot access the eToken.

Command Modes

Global configuration (config)

Command History

Release	Modification
12.3(14)T	This command was introduced.
12.4(11)T	This command was integrated into the Cisco 7200VXR NPE-G2 platform.
15.0(1)M	This command was modified earlier than Cisco IOS Release 15.0(1)M. The **default** keyword was added.
Cisco IOS XE Release 3.6	This command was integrated into Cisco IOS XE Release 3.6.

Usage Guidelines

After the eToken is plugged into the router, the router will use the specified PIN (or the default PIN if no PIN is specified) to automatically log in as the user.

Examples

The following example shows how to access the eToken via the user PIN “12345”:

```
crypto pki token usbtoken0 user-pin 12345
```

Related Commands

Command	Description
crypto pki login	Logs in to the USB eToken.
crypto pki token logout	Logs the router out of the USB eToken.

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* **redundancy**
no crypto pki trustpoint *name*

Syntax Description

name	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.)
redundancy	(Optional) Specifies that the key, and any certificates associated with it, should be synchronized to the standby certificate authority (CA).

Command Default

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

Your router uses unique identifiers during communication with Online Certificate Status Protocol (OCSP) servers, as configured in your network.

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.”
12.2(18)SXD	This command was integrated into Cisco IOS Release 12.2(18)SXD.
12.3(14)T	The **enrollment selfsigned**subcommand was introduced.
12.4(4)T	The **ocsp disable-nonce** subcommand was introduced.
12.2(33)SRA	This command was integrated into Cisco IOS Release 12.2(33)SRA.
15.0(1)M	This command was modified. The **redundancy**keyword was introduced.

Usage Guidelines

Declaring Truspoints

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.
- **ocsp disable-nonce** --Specifies that your router will not send unique identifiers, or nonces, during OCSP communications
- **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.

Specifying Use of Unique Identifiers

When using OCSP as your revocation method, unique identifiers, or nonces, are sent by default during peer communications with the OCSP server. The use of unique identifiers during OCSP server communications enables more secure and reliable communications. However, not all OCSP servers support the use of unique dentures, see your OCSP manual for more information. To disable the use of unique identifiers during OCSP communications, use the **ocsp disable-nonce** subcommand.

Examples

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 pki1
 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
```

The following example shows the unique identifier being disabled for OCSP communications for a previously created trustpoint named ts:

```
crypto pki trustpoint ts
 ocsp disable-nonce
```

The following example shows the **redundancy** keyword specified in the **crypto pki trustpoint** command:

```
Router(config)#crypto pki trustpoint mytp
Router(ca-trustpoint)#redundancy
Router(ca-trustpoint)#show
 redundancy
 revocation-check crl
end
```

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.

crypto pki trustpool import

To manually import (download) the certification authority (CA) certificate bundle into the public key infrastructure (PKI) trustpool to update or replace the existing CA bundle, use the **crypto pki trustpool import** command in global configuration mode. To remove any of the configured parameters, use the **no** form of this command.

crypto pki trustpool import {**clean** [{**terminal** | **url** *url*}] | **terminal** | **url** *url*}
no crypto pki trustpool import {**clean** [{**terminal** | **url** *url*}] | **terminal** | **url** *url*}

Syntax Description

clean	Specifies the removal of the downloaded PKI trustpool certificates before the new certificates are downloaded. Use the optional **terminal** keyword to remove the existing CA certificate bundle terminal setting or the **url** keyword and *url* argument to remove the URL file system setting.
terminal	Specifies the importation of a CA certificate bundle through the terminal (cut-and-paste) in Privacy Enhanced Mail (PEM) format.
url *url*	Specifies the importation of a CA certificate bundle through the URL.

Command Default

The PKI trustpool feature is enabled. The router uses the built-in CA certificate bundle in the PKI trustpool, which is updated automatically from Cisco.

Command Modes

Global configuration (config)

Command History

Release	Modification
15.2(2)T	This command was introduced.
15.1(1)SY	This command was integrated into Cisco IOS Release 15.1(1)SY.

Usage Guidelines

Note Security threats, as well as the cryptographic technologies to help protect against them, are constantly changing. For more information about the latest Cisco cryptographic recommendations, see the Next Generation Encryption (NGE) white paper.

PKI trustpool certificates are automatically updated from Cisco. When the PKI trustpool certificates are not current, use the **crypto pki trustpool import** command to update them from another location.

The *url* argument specifies or changes the URL file system of the CA. The table below lists the available URL file systems.

Table 7: URL File Systems

File System	Description
archive:	Imports from the archive file system.

File System	Description
cns:	Imports from the Cluster Namespace (CNS) file system.
disk0:	Imports from the disc0 file system.
disk1:	Imports from the disc1 file system.
ftp:	Imports from the FTP file system.
http:	Imports from the HTTP file system. The URL must be in the following formats: • http://*CAname:80*, where *CAname* is the Domain Name System (DNS) • http://*ipv4-address*:80. For example: http://10.10.10.1:80. • http://*[ipv6-address]:80*. For example: http://[2001:DB8:1:1::1]:80. The IPv6 address is in hexadecimal notation and must be enclosed in brackets in the URL.
https:	Imports from the HTTPS file system. The URL must use the same formats as the HTTP: file system formats.
null:	Imports from the null file system.
nvram:	Imports from NVRAM file system.
pram:	Imports from Parameter Random-access Memory (PRAM) file system.
rcp:	Imports from the remote copy protocol (rcp) file system.
scp:	Imports from the secure copy protocol (scp) file system.
snmp:	Imports from the Simple Network Management Protocol (SNMP).
system:	Imports from the system file system.
tar:	Imports from the UNIX tar file system.
tftp:	Imports from the TFTP file system. **Note** The URL must be in the form: tftp://*CAname/filespecification*.
tmpsys:	Imports from the Cisco IOS tmpsys file system.
unix:	Imports from the UNIX file system.
xmodem:	Imports from the xmodem simple file transfer protocol system.
ymodem:	Imports from the ymodem simple file transfer protocol system.

Examples

The following example shows how to remove all downloaded PKI trustpool CA certificates and subsequently update the CA certificates in the PKI trustpool by downloading a new CA certification bundle:

```
Router(config)# crypto pki trustpool import clean
Router(config)# crypto pki trustpool import url http://www.cisco.com/security/pki/trs/ios.p7b
```

The following example shows how to update the CA certificates in the PKI trustpool by downloading a new CA certification bundle without removing all downloaded PKI trustpool CA certificates:

```
Router(config)# crypto pki trustpool import url http://www.cisco.com/security/pki/trs/ios.p7b
```

Related Commands

Command	Description
cabundle url	Configures the URL from which the PKI trustpool CA bundle is downloaded.
chain-validation	Enables chain validation from the peer's certificate to the root CA certificate in the PKI trustpool.
crl	Specifes the certificate revocation list (CRL) query and cache options for the PKI trustpool.
crypto pki trustpool policy	Configures PKI trustpool policy parameters.
default	Resets the value of a ca-trustpool configuration command to its default.
match	Enables the use of certificate maps for the PKI trustpool.
ocsp	Specifies OCSP settings for the PKI trustpool.
revocation-check	Disables revocation checking when the PKI trustpool policy is being used.
show	Displays the PKI trustpool policy of the router in ca-trustpool configuration mode.
show crypto pki trustpool	Displays the PKI trustpool certificates of the router and optionally shows the PKI trustpool policy.
source interface	Specifies the source interface to be used for CRL retrieval, OCSP status, or the downloading of a CA certificate bundle for the PKI trustpool.
storage	Specifies a file system location where PKI trustpool certificates are stored on the router.

Command	Description
vrf	Specifies the VRF instance to be used for CRL retrieval.

crypto pki trustpool policy

To configure a public key infrastructure (PKI) trustpool policy parameters, use the **crypto pki trustpool policy** command in global configuration mode.

crypto pki trustpool policy

Syntax Description

This command has no arguments or keywords.

Command Default

The default PKI trustpool policy is used.

Command Modes

Global configuration mode (config)

Command History

Release	Modification
15.2(2)T	This command was introduced.
15.1(1)SY	This command was integrated into Cisco IOS Release 15.1(1)SY.

Usage Guidelines

The **crypto pki trustpool policy** command enters ca-trustpool configuration mode where commands can be accessed to configure certificate authority (CA) PKI trustpool policy parameters.

Examples

```
Router(config)# crypto pki trustpool policy
```

Related Commands

Command	Description
cabundle url	Configures the URL from which the PKI trustpool CA bundle is downloaded.
chain-validation	Enables chain validation from the peer's certificate to the root CA certificate in the PKI trustpool.
crl	Specifes the CRL query and cache options for the PKI trustpool.
crypto pki trustpool import	Manually imports (downloads) the CA certificate bundle into the PKI trustpool to update or replace the existing CA bundle.
default	Resets the value of a ca-trustpool configuration command to its default.
match	Enables the use of certificate maps for the PKI trustpool.

Command	Description
ocsp	Specifies OCSP settings for the PKI trustpool.
revocation-check	Disables revocation checking when the PKI trustpool policy is being used.
show	Displays the PKI trustpool policy of the router in ca-trustpool configuration mode.
show crypto pki trustpool	Displays the PKI trustpool certificates of the router and optionally shows the PKI trustpool policy.
source interface	Specifies the source interface to be used for CRL retrieval, OCSP status, or the downloading of a CA certificate bundle for the PKI trustpool.
storage	Specifies a file system location where PKI trustpool certificates are stored on the router.
vrf	Specifies the VRF instance to be used for CRL retrieval.

crypto provisioning petitioner

To configure a device to become an easy secure device provisioning (SDP) petitioner and enter tti-petitioner configuration mode, use the **crypto provisioning petitioner**command in global configuration mode. To disable petitioner support, use the **no** form of this command.

crypto provisioning petitioner
no crypto provisioning petitioner

Syntax Description

This command has no arguments or keywords.

Command Default

A device (with a crypto image) is configured to be an SDP petitioner.

Command Modes

Global configuration

Command History

Release	Modification
12.3(8)T	The **crypto wui tti petitioner**command was introduced.
12.3(14)T	This command replaced the **crypto wui tti petitioner** command.

Usage Guidelines

SDP uses Trusted Transitive Introduction (TTI) to easily deploy public key infrastructure (PKI) between two end devices. TTI, which is a communication protocol that provides a bidirectional introduction between two end entities, involves the following three entities:

- Introducer--A mutually trusted device that introduces the petitioner to the registrar. The introducer can be a device user, such as a system administrator.
- Petitioner--A new device that is joined to the secure domain.
- Registrar--A server that authorizes the petitioner. The registrar can be a certificate server.

Note Because the petitioner is enabled by default on the device, you only have to issue the **crypto provisioning petitioner** command if you have previously disabled the petitioner or if you want to use an existing trustpoint instead of the automatically generated trustpoint.

Examples

After the SDP exchange is complete, the petitioner will automatically enroll with the registrar and obtain a certificate. The following sample output from the **show running-config** command shows an automatically generated configuration at the petitioner.

Note The petitioner will not have any TTI-specific configuration in the beginning except that the IP HTTP server will be turned on and the Domain Name System (DNS) server needs to be properly configured.)

```
crypto pki trustpoint tti
```

```
! Enrollment url contains the registrar CS details
 enrollment url http://pki1-36a.cisco.com:80
 revocation-check crl
 rsakeypair tti 1024
 auto-enroll 70
```

Related Commands

Command	Description
crypto provisioning registrar	Configures a device to become an SDP registrar and enters tti-registrar configuration mode.
trustpoint (tti-petitioner)	Specifies the trustpoint that is to be associated with the TTI exchange between the SDP petitioner and the SDP registrar.

crypto provisioning registrar

To configure a device to become an easy secure device provisioning (SDP) registrar and enter tti-registrar configuration mode, use the **crypto provisioning registrar**command in global configuration mode. To disable registrar support, use the **no** form of this command.

crypto provisioning registrar
no crypto provisioning registrar

Syntax Description

This command has no arguments or keywords.

Command Default

The registrar is not enabled.

Command Modes

Global configuration

Command History

Release	Modification
12.3(8)T	The **crypto wui tti registrar**command was introduced.
12.3(14)T	This command replaced the **crypto wui tti registrar** command.

Usage Guidelines

SDP uses Trusted Transitive Introduction (TTI) to easily deploy public key infrastructure (PKI) between two end devices. TTI, which is a communication protocol that provides a bidirectional introduction between two end entities, involves the following three entities:

- Introducer--A mutually trusted device that introduces the petitioner to the registrar. The introducer can be a device user, such as a system administrator.
- Petitioner--A new device that is joined to the secure domain.
- Registrar--A server that authorizes the petitioner.

Although any device that contains a crypto image can be the registrar, it is recommended that the registrar be either a Cisco IOS certificate server registration authority (RA) or a Cisco IOS certificate server root.

Examples

The following sample output from the **show running-config**command verifies that the certificate server "cs1" was configured and associated with the TTI exchange between the registrar and petitioner:

```
crypto pki server cs1
 issuer-name CN = ioscs,L = Santa Cruz,C =US
 lifetime crl 336
 lifetime certificate 730
!
crypto pki trustpoint pki-36a
 enrollment url http://pki-36a:80
 ip-address FastEthernet0/0
 revocation-check none
!
crypto pki trustpoint cs1
 revocation-check crl
 rsakeypair cs1
```

```
!
!
crypto pki certificate chain pki-36a
 certificate 03
  308201D0 30820139 A0030201 02020103 300D0609 2A864886 F70D0101 04050030
  34310B30 09060355 04061302 55533114 30120603 55040713 0B205361 6E746120
  4372757A 310F300D 06035504 03130620 696F7363 73301E17 0D303430 31333130
  39333334 345A170D 30363031 33303039 33333434 5A303A31 38301606 092A8648
  86F70D01 09081309 31302E32 332E322E 32301E06 092A8648 86F70D01 09021611
  706B692D 3336612E 63697363 6F2E636F 6D305C30 0D06092A 864886F7 0D010101
  0500034B 00304802 4100AFFA 8F429618 112FAB9D 01F3352E 59DD3D2D AE67E31D
  370AC4DA 619735DF 9CF4EA13 64E4B563 C239C5F0 1578B773 07BED641 A18CA629
  191884B5 61B66ECF 4D110203 010001A3 30302E30 0B060355 1D0F0404 030205A0
  301F0603 551D2304 18301680 141DA8B1 71652961 3F7D69F0 02903AC3 2BADB137
  C6300D06 092A8648 86F70D01 01040500 03818100 67BAE186 327CED31 D642CB39
  AD585731 95868683 B950DF14 3BCB155A 2B63CFAD B34B579C 79128AD9 296922E9
  4DEDFCAF A7B5A412 AB1FC081 09951CE3 08BFFDD9 9FB1B9DA E9AA42C8 D1049268
  C524E58F 11C6BA7F C750320C 03DFB6D4 CBB3E739 C8C76359 CE939A97 B51B3F7F
  3FF;A9D82 9CFDB6CF E2503A14 36D0A236 A1CCFEAE
  quit
 certificate ca 01
  30820241 308201AA A0030201 02020101 300D0609 2A864886 F70D0101 04050030
  34310B30 09060355 04061302 55533114 30120603 55040713 0B205361 6E746120
  4372757A 310F300D 06035504 03130620 696F7363 73301E17 0D303430 31333130
  39333132 315A170D 30373031 33303039 33313231 5A303431 0B300906 03550406
  13025553 31143012 06035504 07130B20 53616E74 61204372 757A310F 300D0603
  55040313 0620696F 73637330 819F300D 06092A86 4886F70D 01010105 0003818D
  00308189 02818100 FC0695AF 181CE90A 1B34B348 BA957178 680C8B51 07802AC3
  BF77B9C6 CB45092E 3C22292D C7D5FFC1 899185A1 FD8F37D5 C44FC206 6D1FA581
  E2264C83 1CC7453E 548C89C6 F3CD25BC 9BFFE7C5 E6653A06 62133950 78BED51B
  49128428 AB237F80 83A530EA 6F896193 F2134B54 D181F059 348AA84B 21EE6D80
  727BF668 EB004341 02030100 01A36330 61300F06 03551D13 0101FF04 05300301
  01FF300E 0603551D 0F0101FF 04040302 0186301D 0603551D 0E041604 141DA8B1
  71652961 3F7D69F0 02903AC3 2BADB137 C6301F06 03551D23 04183016 80141DA8
  B1716529 613F7D69 F002903A C32BADB1 37C6300D 06092A86 4886F70D 01010405
  00038181 00885895 A0141169 3D754EB2 E6FEC293 5BF0A80B E424AA2F A3F59765
  3463AAD1 55E71F0F B5D1A35B 9EA79DAC DDB40721 1344C01E 015BAB73 1E148E03
  9DD01431 A5E2887B 4AEC8EF4 48ACDB66 A6F9401E 8F7CA588 8A4199BB F8A437A0
  F25064E7 112805D3 074A154F 650D09B9 8FA19347 ED359EAD 4181D9ED 0C667C10
  8A7BCFB0 FB
  quit
crypto pki certificate chain cs1
 certificate ca 01
  30820241 308201AA A0030201 02020101 300D0609 2A864886 F70D0101 04050030
  34310B30 09060355 04061302 55533114 30120603 55040713 0B205361 6E746120
  4372757A 310F300D 06035504 03130620 696F7363 73301E17 0D303430 31333130
  39333132 315A170D 30373031 33303039 33313231 5A303431 0B300906 03550406
  13025553 31143012 06035504 07130B20 53616E74 61204372 757A310F 300D0603
  55040313 0620696F 73637330 819F300D 06092A86 4886F70D 01010105 0003818D
  00308189 02818100 FC0695AF 181CE90A 1B34B348 BA957178 680C8B51 07802AC3
  BF77B9C6 CB45092E 3C22292D C7D5FFC1 899185A1 FD8F37D5 C44FC206 6D1FA581
  E2264C83 1CC7453E 548C89C6 F3CD25BC 9BFFE7C5 E6653A06 62133950 78BED51B
  49128428 AB237F80 83A530EA 6F896193 F2134B54 D181F059 348AA84B 21EE6D80
  727BF668 EB004341 02030100 01A36330 61300F06 03551D13 0101FF04 05300301
  01FF300E 0603551D 0F0101FF 04040302 0186301D 0603551D 0E041604 141DA8B1
  71652961 3F7D69F0 02903AC3 2BADB137 C6301F06 03551D23 04183016 80141DA8
  B1716529 613F7D69 F002903A C32BADB1 37C6300D 06092A86 4886F70D 01010405
  00038181 00885895 A0141169 3D754EB2 E6FEC293 5BF0A80B E424AA2F A3F59765
  3463AAD1 55E71F0F B5D1A35B 9EA79DAC DDB40721 1344C01E 015BAB73 1E148E03
  9DD01431 A5E2887B 4AEC8EF4 48ACDB66 A6F9401E 8F7CA588 8A4199BB F8A437A02;
  F25064E7 112805D3 074A154F 650D09B9 8FA19347 ED359EAD 4181D9ED 0C667C10
  8A7BCFB0 FB
  quit
!
```

```
crypto provisioning registrar
 pki-server cs1
!
!
!
crypto isakmp policy 1
 hash md5
!
!
crypto ipsec transform-set test_transformset esp-3des
!
crypto map test_cryptomap 10 ipsec-isakmp
 set peer 10.23.1.10
 set security-association lifetime seconds 1800
 set transform-set test_transformset
 match address 170
```

Related Commands

Command	Description
crypto pki server	Enables a Cisco IOS certificate server and enters certificate server configuration mode.
crypto provisioning **petitioner**	Configures a device to become an SDP petitioner and enters tti-petitioner configuration mode.

crypto skip-client

To configure the Secure Key Integration Protocol (SKIP) client that specifies parameters to securely connect to and import PPKs from an external key source, use the **crypto skip-client** command in global configuration mode. To delete a SKIP client configuration, use the **no** form of this command in the global configuration mode.

crypto skip-client *skip-client-name*
no crypto skip-client *skip-client-name*

Syntax Description

skip-client-name	The name of the SKIP client.

Command Default

There is no default SKIP client configuration.

Command Modes

Global configuration (config)

Command History

Release	Modification
Cisco IOS XE Release 17.11.1a	This command was introduced.

Usage Guidelines

The SKIP client configuration specifies the parameters that are required to securely communicate with and request PPKs from an external SKIP-compliant key source, for quantum-safe encryption.

After you enter the **crypto skip-client** command, the prompt changes to the following:

```
Router(config-crypto-skip-client)#
```

The following **crypto skip-client** submode commands are available:

- **exit** - Exits from crypto ssl policy submode.
- **no** - Negates a command or set its defaults.
- **psk** - Specifies the preshared key for the SKIP TLS session.

  psk id *identity* **key** { **0** | **6** | **hex** } *key-value*

identity	PSK identity.
0	An unencrypted password will follow.
6	An encrypted password will follow.
hex	A hexadecimal string will follow.
key-value	Encrypted or unencrypted PSK.

- **server** - Specifies the SKIP server.

  server *identity* **key** { **fqdn** *domain-name* **port** *port-number* | **ipv4** *ipv4-address* **port** *port-number* | **ipv6** *ipv6-address* **port** *port-number* }

domain-name	Fully Qualified Domain Name (FQDN).
port-number	Port number.
ipv4-address	IPv4 address.
ipv6-address	IPv6 address.

Examples

The following example shows how to configure an SKIP client with an IPv4 server address and an unencrypted PSK in plain text:

```
Router(config-crypto-skip-client)#crypto skip-client skip-client-cfg
Router(config-crypto-skip-client)#server ipv4 10.10.0.3 port 9991
Router(config-crypto-skip-client)#psk id psk-id key 0 cisco123
Router(config-crypto-skip-client)#end
```

The following example shows how to configure an SKIP client with an IPv6 server address and an encrypted PSK:

```
Router(config-crypto-skip-client)#crypto skip-client skip-client-cfg
Router(config-crypto-skip-client)#server ipv6 2001::1:1 port 443
Router(config-crypto-skip-client)#psk id psk-id key 6 [XO[J\fAbOhiLUC]^ZRIEQNTefDAAB
Router(config-crypto-skip-client)#end
```

Related Commands

Command	Description
crypto ikev2 keyring	Specifies a manual or dynamic PPK in a keyring.
crypto ikev2 profile	Configures a PPK keyring in an IKEv2 profile.

crypto vpn

To install a Cisco Secure Desktop (CSD) or Cisco AnyConnect VPN Client package file on a Secure Socket Layer VPN (SSL VPN) gateway for distribution to end users, use the **crypto vpn** command in global configuration mode. To remove a package file from the SSL VPN gateway, use the **no** form of this command.

crypto vpn {**anyconnect** *file name* **sequence** *sequence-number* | **profile** *profile-name device* **:** *file name* | **csd** *file name*}
no crypto vpn {**anyconnect** *file name* **sequence** *sequence-number* | **profile** *profile-name device* **:** *file name* | **csd** *file name*}

Syntax Description

anyconnect *file name*	Installs the specified file from the Cisco AnyConnect VPN Client package.
sequence *sequence-number*	Allows for multiple packages to be installed on one gateway. If the **sequence** keyword and the *sequence-number* argument are not configured, a sequence number of 1 is applied to the package.
profile *profile-name device:file name*	Installs the profile of the Cisco AnyConnect VPN Client and the device into which the profile is imported.
csd	Installs the CSD package.

Command Default

Neither a CSD nor a Cisco AnyConnect VPN Client package file is installed on an SSL VPN gateway.

Command Modes

Global configuration (config)

Command History

Release	Modification
15.2(1)T	This command was introduced.

Usage Guidelines

The CSD and Cisco AnyConnect VPN Client installation packages must first be copied to a local file system, such as disk, flash, or USB flash. The CSD and Cisco AnyConnect VPN Client software packages are pushed to end users as access is needed. The end user must have administrative privileges, and the Java Runtime Environment (JRE) for Windows version 1.4 or a later version must be installed before a CSD or Cisco AnyConnect VPN Client package can be installed.

Note SSL VPN Client (SVC) is the predecessor of Cisco AnyConnect VPN Client software.

If you have not entered the **sequence** keyword and the *sequence-number* argument and you want to install another package, you can remove the previous package (using the **no** form of the command) or you can provide another sequence number.

If you try to install a package with a sequence number that is being used, you will get an error message.

Examples

The following example shows how to install the Cisco AnyConnect VPN Client package on an SSL VPN gateway:

```
Device(config)# crypto vpn anyconnect filea sequence 5
```

Related Commands

Command	Description
csd enable	Enables CSD support for SSL VPN sessions.

crypto wui tti petitioner

Note This command was replaced by the **crypto provisioning petitioner** command effective with Cisco IOS Release 12.3(14)T.

To configure a device to become an easy secure device deployment (EzSDD) petitioner and enter tti-petitioner configuration mode, use the **crypto wui tti petitioner**command in global configuration mode. To disable petitioner support, use the **no** form of this command.

crypto wui tti petitioner
no crypto wui tti petitioner

Syntax Description

This command has no arguments or keywords.

Command Default

A device (with a crypto image) is configured to be an EzSDD petitioner.

Command Modes

Global configuration

Command History

Release	Modification
12.3(8)T	This command was introduced.

Usage Guidelines

EzSDD uses Trusted Transitive Introduction (TTI) to easily deploy public key infrastructure (PKI) between two end devices. TTI, which is a communication protocol that provides a bidirectional introduction between two end entities, involves the following three entities:

- Introducer--A mutually trusted device that introduces the petitioner to the registrar. The introducer can be a device user, such as a system administrator.
- Petitioner--A new device that is joined to the secure domain.
- Registrar--A server that authorizes the petitioner. The registrar can be a certificate server.

Note Because the petitioner is enabled by default on the device, you only have to issue the **crypto wui tti petitioner** command if you have previously disabled the petitioner or if you want to use an existing trustpoint instead of the automatically generated trustpoint.

Examples

After the EzSDD exchange is complete, the petitioner will automatically enroll with the registrar and obtain a certificate. The following sample output from the **show running-config** command shows an automatically generated configuration at the petitioner. (Note that petitioner will not have any TTI-specific configuration in the beginning except that the http server will be turned on and the Domain Name System (DNS) server needs to be properly configured.)

```
crypto pki trustpoint tti
! Enrollment url contains the registrar CS details
 enrollment url http://pki1-36a.cisco.com:80
 revocation-check crl
 rsakeypair tti 1024
 auto-enroll 70
```

Related Commands

Command	Description
crypto wui tti registrar	Configures a device to become an EzSDD registrar and enters tti-registrar configuration mode.
trustpoint (tti-petitioner)	Specifies the trustpoint that is to be associated with the TTI exchange between the EzSDD petitioner and the EzSDD registrar.

crypto wui tti registrar

Note This command was replaced by the **crypto provisioning registrar** command effective with Cisco IOS Release 12.3(14)T.

To configure a device to become an easy secure device deployment (EzSDD) registrar and enter tti-registrar configuration mode, use the **crypto wui tti registrar**command in global configuration mode. To disable registrar support, use the **no** form of this command.

crypto wui tti registrar
no crypto wui tti registrar

Syntax Description

This command has no arguments or keywords.

Command Default

The registrar is not enabled.

Command Modes

Global configuration

Command History

Release	Modification
12.3(8)T	This command was introduced.

Usage Guidelines

EzSDD uses Trusted Transitive Introduction (TTI) to easily deploy public key infrastructure (PKI) between two end devices. TTI, which is a communication protocol that provides a bidirectional introduction between two end entities, involves the following three entities:

- Introducer--A mutually trusted device that introduces the petitioner to the registrar. The introducer can be a device user, such as a system administrator.
- Petitioner--A new device that is joined to the secure domain.
- Registrar--A server that authorizes the petitioner.

Although any device that contains a crypto image can be the registrar, it is recommended that the registrar be either a Cisco IOS certificate server registration authority (RA) or a Cisco IOS certificate server root.

Examples

The following sample output from the **show running-config**command verifies that the certificate server "cs1" was configured and associated with the TTI exchange between the registrar and petitioner:

```
crypto pki server cs1
 issuer-name CN = ioscs,L = Santa Cruz,C =US
 lifetime crl 336
 lifetime certificate 730
!
crypto pki trustpoint pki-36a
 enrollment url http://pki-36a:80
 ip-address FastEthernet0/0
 revocation-check none
```

```
!
crypto pki trustpoint cs1
 revocation-check crl
 rsakeypair cs1
!
!
crypto pki certificate chain pki-36a
 certificate 03
  308201D0 30820139 A0030201 02020103 300D0609 2A864886 F70D0101 04050030
  34310B30 09060355 04061302 55533114 30120603 55040713 0B205361 6E746120
  4372757A 310F300D 06035504 03130620 696F7363 73301E17 0D303430 31333130
  39333334 345A170D 30363031 33303039 33333434 5A303A31 38301606 092A8648
  86F70D01 09081309 31302E32 332E322E 32301E06 092A8648 86F70D01 09021611
  706B692D 3336612E 63697363 6F2E636F 6D305C30 0D06092A 864886F7 0D010101
  0500034B 00304802 4100AFFA 8F429618 112FAB9D 01F3352E 59DD3D2D AE67E31D
  370AC4DA 619735DF 9CF4EA13 64E4B563 C239C5F0 1578B773 07BED641 A18CA629
  191884B5 61B66ECF 4D110203 010001A3 30302E30 0B060355 1D0F0404 030205A0
  301F0603 551D2304 18301680 141DA8B1 71652961 3F7D69F0 02903AC3 2BADB137
  C6300D06 092A8648 86F70D01 01040500 03818100 67BAE186 327CED31 D642CB39
  AD585731 95868683 B950DF14 3BCB155A 2B63CFAD B34B579C 79128AD9 296922E9
  4DEDFCAF A7B5A412 AB1FC081 09951CE3 08BFFDD9 9FB1B9DA E9AA42C8 D1049268
  C524E58F 11C6BA7F C750320C 03DFB6D4 CBB3E739 C8C76359 CE939A97 B51B3F7F
  3FF;A9D82 9CFDB6CF E2503A14 36D0A236 A1CCFEAE
  quit
 certificate ca 01
  30820241 308201AA A0030201 02020101 300D0609 2A864886 F70D0101 04050030
  34310B30 09060355 04061302 55533114 30120603 55040713 0B205361 6E746120
  4372757A 310F300D 06035504 03130620 696F7363 73301E17 0D303430 31333130
  39333132 315A170D 30373031 33303039 33313231 5A303431 0B300906 03550406
  13025553 31143012 06035504 07130B20 53616E74 61204372 757A310F 300D0603
  55040313 0620696F 73637330 819F300D 06092A86 4886F70D 01010105 0003818D
  00308189 02818100 FC0695AF 181CE90A 1B34B348 BA957178 680C8B51 07802AC3
  BF77B9C6 CB45092E 3C22292D C7D5FFC1 899185A1 FD8F37D5 C44FC206 6D1FA581
  E2264C83 1CC7453E 548C89C6 F3CD25BC 9BFFE7C5 E6653A06 62133950 78BED51B
  49128428 AB237F80 83A530EA 6F896193 F2134B54 D181F059 348AA84B 21EE6D80
  727BF668 EB004341 02030100 01A36330 61300F06 03551D13 0101FF04 05300301
  01FF300E 0603551D 0F0101FF 04040302 0186301D 0603551D 0E041604 141DA8B1
  71652961 3F7D69F0 02903AC3 2BADB137 C6301F06 03551D23 04183016 80141DA8
  B1716529 613F7D69 F002903A C32BADB1 37C6300D 06092A86 4886F70D 01010405
  00038181 00885895 A0141169 3D754EB2 E6FEC293 5BF0A80B E424AA2F A3F59765
  3463AAD1 55E71F0F B5D1A35B 9EA79DAC DDB40721 1344C01E 015BAB73 1E148E03
  9DD01431 A5E2887B 4AEC8EF4 48ACDB66 A6F9401E 8F7CA588 8A4199BB F8A437A0
  F25064E7 112805D3 074A154F 650D09B9 8FA19347 ED359EAD 4181D9ED 0C667C10
  8A7BCFB0 FB
  quit
crypto pki certificate chain cs1
 certificate ca 01
  30820241 308201AA A0030201 02020101 300D0609 2A864886 F70D0101 04050030
  34310B30 09060355 04061302 55533114 30120603 55040713 0B205361 6E746120
  4372757A 310F300D 06035504 03130620 696F7363 73301E17 0D303430 31333130
  39333132 315A170D 30373031 33303039 33313231 5A303431 0B300906 03550406
  13025553 31143012 06035504 07130B20 53616E74 61204372 757A310F 300D0603
  55040313 0620696F 73637330 819F300D 06092A86 4886F70D 01010105 0003818D
  00308189 02818100 FC0695AF 181CE90A 1B34B348 BA957178 680C8B51 07802AC3
  BF77B9C6 CB45092E 3C22292D C7D5FFC1 899185A1 FD8F37D5 C44FC206 6D1FA581
  E2264C83 1CC7453E 548C89C6 F3CD25BC 9BFFE7C5 E6653A06 62133950 78BED51B
  49128428 AB237F80 83A530EA 6F896193 F2134B54 D181F059 348AA84B 21EE6D80
  727BF668 EB004341 02030100 01A36330 61300F06 03551D13 0101FF04 05300301
  01FF300E 0603551D 0F0101FF 04040302 0186301D 0603551D 0E041604 141DA8B1
  71652961 3F7D69F0 02903AC3 2BADB137 C6301F06 03551D23 04183016 80141DA8
  B1716529 613F7D69 F002903A C32BADB1 37C6300D 06092A86 4886F70D 01010405
  00038181 00885895 A0141169 3D754EB2 E6FEC293 5BF0A80B E424AA2F A3F59765
  3463AAD1 55E71F0F B5D1A35B 9EA79DAC DDB40721 1344C01E 015BAB73 1E148E03
  9DD01431 A5E2887B 4AEC8EF4 48ACDB66 A6F9401E 8F7CA588 8A4199BB F8A437A02;
```

```
  F25064E7 112805D3 074A154F 650D09B9 8FA19347 ED359EAD 4181D9ED 0C667C10
  8A7BCFB0 FB
  quit
!
crypto wui tti registrar
 pki-server cs1
!
!
!
crypto isakmp policy 1
 hash md5
!
!
crypto ipsec transform-set test_transformset esp-3des
!
crypto map test_cryptomap 10 ipsec-isakmp
 set peer 10.23.1.10
 set security-association lifetime seconds 1800
 set transform-set test_transformset
 match address 170
```

Related Commands

Command	Description
crypto pki server	Enables a Cisco IOS certificate server and enters certificate server configuration mode.
crypto wui tti petitioner	Configures a device to become an EzSDD petitioner and enters tti-petitioner configuration mode.

crypto xauth

To configure crypto Extended Authentication (xauth) parameters globally on a per-interface basis, use the **crypto xauth**command in global configuration mode. To disable the xauth parameters, use the **no** form of this command.

crypto xauth *interface-name interface-number*
no crypto xauth *interface-name interface-number*

Syntax Description

interface-name	Name of the interface.
interface-number	Number of the related interface. Each interface has a related range of numbers. For example, the asynchronous interface has a range of interface numbers from 1 to 5 and the BVI interface has a range of interface numbers from 1 to 255.

Command Default

Crypto xauth parameters are not configured on any interface.

Command Modes

Global configuration (config)

Command History

Release	Modification
15.0(1)M	This command was introduced in a release earlier than Cisco IOS 15.0(1)M.

Usage Guidelines

This command is mainly used on responders.

This command is used to disable the negotiation of xauth capabilities during proposals for a session that is terminating on a specific interface.

The **no crypto xauth** command enables the negotiation of xauth capabilities.

Examples

The following example shows how to enable crypto xauth parameters globally on a per-interface basis:

```
Router> enable
Router# configure terminal
Router(config)# crypto xauth fastethernet 0/1
```

The following example shows how the **no crypto xauth** command uses the nonvolatile generation (NVGEN) process to perform a configuration state retrieval operation when you specify the **show run**command:

```
Router> enable
Router# configure terminal
Router(config)# no crypto xauth fastethernet 0/1

Router# show run
archive
 log config
  hidekeys
!
```

```
redundancy
!
!
!
no crypto xauth Ethernet0/0
```

Related Commands

Command	Description
crypto key decrypt rsa	Deletes the encrypted RSA key and leaves only the unencrypted key on the running router.

csd enable

To enable Cisco Secure Desktop (CSD) support for SSL VPN sessions, use the **csd enable** command in webvpn context configuration mode. To remove CSD support from the SSL VPN context configuration, use the **no** form of this command.

csd enable
no csd enable

Syntax Description

This command has no keywords or arguments.

Command Default

CSD support is not enabled.

Command Modes

Webvpn context configuration

Command History

Release	Modification
12.4(6)T	This command was introduced.

Usage Guidelines

The CSD software installation package must be present in a local file system, such as flash memory, and it must be cached for distribution to end users (remote PC or networking device). The **webvpn install**command is used to install the software installation package to the distribution cache.

Examples

The following example enables CSD support for SSL VPN sessions:

```
Router(config)# webvpn install csd flash:/securedesktop_3_1_0_9.pkg

SSLVPN Package Cisco-Secure-Desktop : installed successfully
Router(config)# webvpn context context1

Router(config-webvpn-context)# csd enable
```

Related Commands

Command	Description
webvpn context	Enters webvpn context configuration mode to configure the SSL VPN context.
webvpn install	Installs a CSD or SSL VPN client package file to a SSL VPN gateway for distribution to end users.

ctcp port

To set the port number for Cisco Tunneling Control Protocol (cTCP) encapsulation for Easy VPN, use the **ctcp port** command in crypto ipsec client ezvpn configuration mode. To disable the port that was configured, use the **no** form of this command.

ctcp port *port-number*
no ctcp port

Syntax Description

port-number	Port number. Value = 1 through 65535.

Command Default

If a port is not specified, the default port is the port on which the cTCP server listens.

Command Modes

Crypto ipsec client ezvpn configuration (config-crypto-ezvpn)

Command History

Release	Modification
12.4(20)T	This command was introduced.

Usage Guidelines

This command is used only on the Easy VPN remote device.

Examples

The following example shows that the cTCP port number has been set to 10:

```
Router (config)# crypto ipsec client ezvpn client1
Router (config-crypto-ezvpn)# ctcp port 10
```

Related Commands

Command	Description
crypto ctcp	Configures cTCP encapsulation for Easy VPN.

ctype

To preauthenticate calls on the basis of the call type, use the **ctype** command in AAA preauthentication configuration mode. To remove the **ctype** command from your configuration, use the **no** form of this command.

ctype [{**if-avail** | **required**}] [**accept-stop**] [**password** *password*] [{**digital** | **speech** | **v.110** | **v.120**}]
no ctype [{**if-avail** | **required**}] [**accept-stop**] [**password** *password*] [{**digital** | **speech** | **v.110** | **v.120**}]

Syntax Description

if-avail	(Optional) Implies that if the switch provides the data, RADIUS must be reachable and must accept the string in order for preauthentication to pass. If the switch does not provide the data, preauthentication passes.
required	(Optional) Implies that the switch must provide the associated data, that RADIUS must be reachable, and that RADIUS must accept the string in order for preauthentication to pass. If these three conditions are not met, preauthentication fails.
accept-stop	(Optional) Prevents subsequent preauthentication elements such as clid or dnis from being tried once preauthentication has succeeded for a call element.
password *password*	(Optional) Defines the password for the preauthentication element.
digital	(Optional) Specifies “digital” as the call type for preauthentication.
speech	(Optional) Specifies “speech” as the call type for preauthentication.
v.110	(Optional) Specifies “v.110” as the call type for preauthentication.
v.120	(Optional) Specifies “v.120” as the call type for preauthentication.

Command Default

The **if-avail** and **required** keywords are mutually exclusive. If the **if-avail** keyword is not configured, the preauthentication setting defaults to **required**.

The default password string is cisco.

Command Modes

AAA preauthentication configuration

Command History

Release	Modification
12.1(2)T	This command was introduced.
12.2(33)SRA	This command was integrated into Cisco IOS Release 12.2(33)SRA.
12.2SX	This command is supported in the Cisco IOS Release 12.2SX train. Support in a specific 12.2SX release of this train depends on your feature set, platform, and platform hardware.

Usage Guidelines

You may configure more than one of the AAA preauthentication commands (**clid**, **ctype**, **dnis**) to set conditions for preauthentication. The sequence of the command configuration decides the sequence of the preauthentication conditions. For example, if you configure **dnis**, then **clid**, then **ctype**, in this order, then this is the order of the conditions considered in the preauthentication process.

In addition to using the preauthentication commands to configure preauthentication on the Cisco router, you must set up the preauthentication profiles on the RADIUS server.

Set up the RADIUS preauthentication profile with the call type string as the username and with the password that is defined in the **ctype** command as the password. The table below shows the call types that you may use in the preauthentication profile.

Table 8: Preauthentication Call Types

Call Type String	ISDN Bearer Capabilities
digital	Unrestricted digital, restricted digital.
speech	Speech, 3.1 kHz audio, 7 kHz audio.
v.110	Anything with V.110 user information layer.
v.120	Anything with V.120 user information layer.

Examples

The following example specifies that incoming calls be preauthenticated on the basis of the call type:

```
aaa preauth
 group radius
 ctype required
```

Related Commands

Command	Description
clid	Preauthenticates calls on the basis of the CLID number.
dnis (RADIUS)	Preauthenticates calls on the basis of the DNIS number.
dnis bypass (AAA preauthentication configuration)	Specifies a group of DNIS numbers that will be bypassed for preauthentication.
group (RADIUS)	Specifies the AAA RADIUS server group to use for preauthentication.

cts authorization list network

To specify a list of AAA servers for the Cisco TrustSec (CTS) seed device to use, use the **cts authorization list network** command in global configuration mode. To stop using the list during authentication, use the **no** form of this command.

cts authorization list network *server_list*
no cts authorization list network *list-name*

Syntax Description

list-name	Specifies a Cisco TrustSec AAA server group.

Command Default

No CTS AAA server list is specified.

Command Modes

Global configuration (config)

Command History

Release	Modification
12.2(33)SXI3	This command was introduced on the Catalyst 6500 series switches.
15.2(2)T	This command was integrated into Cisco IOS Release 15.2(2)T.

Usage Guidelines

A CTS AAA server list is specified in order to establish CTS credentials so that CTS works on your router that is acting as a seed device.

This command is only for the seed device. Non-seed devices obtain the CTS AAA server list from their CTS authenticator peer as a component of their TrustSec environment data. This server list is created by the **aaa authorization network** *list-name* **group radius** command.

Examples

The following example shows how to specify a list of AAA servers for a CTS seed device:

```
Router# cts credentials id Router password Cisco123

Router# configure terminal
Router(config)# aaa new-model
Router(config)# aaa authentication dot1x default group radius
Router(config)# aaa authorization network cts-mlist group radius
Router(config)# cts authorization list cts-mlist
Router(config)# aaa accounting dot1x default start-stop group radius
Router(config)# radius-server host 10.20.3.1 auth-port 1812 acct-port 1813 pac key AbCe1234
Router(config)# radius-server vsa send authentication
Router(config)# dot1x system-auth-control
Router(config)# exit
```

Related Commands

Command	Description
show cts server-list	Displays RADIUS server configurations for CTS seed devices.

cts credentials

To specify the Cisco TrustSec (CTS) ID and password of the network device, use the **cts credentials** command in privileged EXEC mode. To delete the CTS credentials, use the **clear cts credentials** command.

cts credentials id *cts-id* **password** *cts-pwd*

Syntax Description

cts-id	The CTS device ID for this device used when authenticating with other CTS devices with EAP-FAST. This argument has a maximum length of 32 characters and is case sensitive.
password *cts-pwd*	Specifies the password for this device to use when authenticating with other CTS devices with EAP-FAST.

Command Default

No CTS credentials are specified.

Command Modes

Privileged EXEC (#)

Command History

Release	Modification
12.2(33)SXI	This command was introduced on the Catalyst 6500 series switches.
15.2(2)T	This command was integrated into Cisco IOS Release 15.2(2)T.

Usage Guidelines

CTS requires each device in the network to identity itself uniquely. For use in TrustSec Network Device Admission Control (NDAC) authentication, the **cts credentials** command specifies the Cisco TrustSec device ID and password for this switch to use when authenticating with other Cisco TrustSec devices with EAP-FAST. The CTS credentials state retrieval is not performed by the nonvolatile generation process (NVGEN) because the CTS credential information is saved in the keystore, not in the startup-config. The device can be assigned a CTS identity by the Cisco Secure Access Control Server (ACS), or auto-generate a new password when prompted to do so by the ACS. Those credentials are stored in the keystore, eliminating the need to save the running-config. To display the CTS device ID, use the **show cts credentials** command. The stored password is never displayed.

To change the device ID or the password, reenter the command. To clear the keystore, use the **clear cts credentials** command.

Note When the CTS device ID is changed, all Protected Access Credentials (PACs) are flushed from the keystore because the PACs are associated with the old device ID and are not valid for a new identity.

Examples

The following example configures himalaya and cisco as the CTS device ID and password:

```
Router# cts credentials id himalaya password cisco

CTS device ID and password have been inserted in the local keystore. Please make sure that
 the same ID and password are configured in the server database.
```

The following example changes the CTS device ID and password to atlas and cisco123:

```
Router# cts credentials id atlas password cisco123

A different device ID is being configured.
This may disrupt connectivity on your CTS links.
Are you sure you want to change the Device ID? [confirm] y
TS device ID and password have been inserted in the local keystore. Please make sure that
the same ID and password are configured in the server database.
```

The following example displays the CTS device ID and password state:

```
Router# show cts credentials

CTS password is defined in keystore, device-id = atlas
```

Related Commands

Command	Description
clear cts credentials	Clears the CTS device ID and password.
show cts credentials	Displays the state of the current CTS device ID and password.
show cts keystore	Displays contents of the hardware and software keystores.

cts dot1x

Use the **cts dot1x** command in interface configuration mode to enable Network Device Admission Control (NDAC) and configure NDAC authentication parameters. Use the **no** form of the command to disable NDAC authentication on the interface.

cts dot1x
no cts dot1x

Syntax Description

This command has no arguments or keywords.

Command Default

CTS dot1x configuration on the interface is disabled by default.

Command Modes

Interface configuration (config-if)

Command History

Release	Modification
12.2(33)SXI3	This command was introduced on the Catalyst 6500 series switches.
15.1(1)SY	This command was integrated into Cisco IOS Release 15.1(1)SY.

Usage Guidelines

Once the **cts dot1x** command is specified, CTS dot1x interface configuration mode (config-if-cts-dot1x) is entered where Cisco TrustSec NDAC parameters can be configured. Cisco TrustSec NDAC is enabled when the interface is enabled. Cisco TrustSec NDAC must be enabled with 802.1X on each uplink interface that connects to another Cisco TrustSec device.

Examples

```
Device# configure terminal
Device(config)# interface gigabitethernet 3/1
Device(config-if)# cts dot1x
Device(config-if-cts-dot1x)# sap mode-list gcm null no-encap
Device(config-if-cts-dot1x)# timer reauthentication 43200
Device(config-if-cts-dot1x)# exit
Device(config-if)# no shutdown
Device(config-if)# end
Device#
```

Related Commands

Command	Description
propagate sgt (config-if-cts-dot1x)	Enables Security Group Tag (SGT) propagation on a Cisco TrustSec (CTS) 802.1X interface.
sap mode-list (config-if-cts-dot1x)	Configures CTS Security Association Protocol (SAP) authentication.
show cts interface	Displays CTS interface status and configurations.
show dot1x interface	Displays IEEE 802.1x configurations and statistics.
timer reauthentication (config-if-cts-dot1x)	Configures the reauthentication timer for a CTS device.

cts manual

To manually enable an interface for Cisco TrustSec Security (CTS), use the **cts manual** command in interface configuration mode.

cts manual

Syntax Description

This command has no arguments or keywords.

Command Default

Disabled

Command Modes

Interface configuration (config-if)

Command History

Release	Modification
4.1(2)	This command was introduced on the Cisco Nexus 7000 series switches.
Cisco IOS XE Release 3.4S	This command was integrated into Cisco IOS XE Release 3.4S.
15.1(3)S	This command was integrated into Cisco IOS Release 15.1(3)S.

Usage Guidelines

When the **cts manual** command is entered, CTS is enabled on the interface and CTS manual interface configuration mode is entered where CTS parameters can be configured.

All CTS configuration commands with VRF parameters require that the named VRF exists. If the VRF is removed, then the associated CTS configuration is also removed.

Examples

The following example shows how to enter CTS manual interface configuration mode on an interface:

```
Router# configure terminal
Router(config)# interface gigabitethernet 0
Router(config-if)# cts manual
Router(config-if-cts-manual))#
```

The following example shows how to remove the CTS manual configuration from an interface:

```
Router# configure terminal
Router(config)# interface gigabitethernet 0
Router(config-if)# no cts manual
```

Related Commands

Command	Description
propagate sgt	Enables Security Group Tag (SGT) propagation at Layer 2 on Cisco TrustSec Security (CTS) interfaces.
show cts interface	Displays information about CTS interfaces.

cts role-based enforcement

To enable role-based access control globally and on specific Layer 3 interfaces using Cisco TrustSec, use the **cts role-based enoforcement** command in global configuration mode and interface configuration mode respectively. To disable the enforcement of role-based access control at an interface level, use the **no** form of this command.

cts role-based enforcement
no cts role-based enforcement

Syntax Description

This command has no keywords or arguments.

Command Default

Enforcement of role-based access control at an interface level is disabled globally.

Command Modes

Global configuration (config)

Interface configuration (config-if)

Command History

Release	Modification
15.1(2)SY	This command was introduced.

Usage Guidelines

The **cts role-based enforcement** command in global configuration mode enables role-based access control globally. Once role-based access control is enabled globally, it is automatically enabled on every Layer 3 interface on the device. To disable role-based access control on specific Layer 3 interfaces, use the **no** form of the command in interface configuration mode. The **cts role-based enforcement** command in interface configuration mode enables enforcement of role-based access control on specific Layer 3 interfaces.

The attribute-based access control list organizes and manages the Cisco TrustSec access control on a network device. The security group access control list (SGACL) is a Layer 3-4 access control list to filter access based on the value of the security group tag (SGT). The filtering usually occurs at an egress port of the Cisco TrustSec domain. The terms role-based access control list (RBACL) and SGACL can be used interchangeably, and they refer to a topology-independent ACL used in an attribute-based access control (ABAC) policy model.

The following example shows how to enable role-based access control on a Gigabit Ethernet interface:

```
Device> enable
Device# configure terminal
Device(config)# interface gigabitethernet 1/1/3
Device(config-if)# cts role-based enforcement
Device(config-if)# end
```

cts role-based sgt-cache

To enable Security Group Tag (SGT) caching on an interface, use the **cts role-based sgt-cache** command in interface configuration mode. To disable SGT caching on an interface, use the **no** form of this command.

cts role-based sgt-cache {**egress** | **ingress**}

no cts role-based sgt-cache {**egress** | **ingress**}

Syntax Description

egress	Enables SGT caching at the egress point of an interface.
ingress	Enables SGT caching at the ingress point of an interface.

Command Default

SGT caching is enabled on the interface.

Command Modes

Interface configuration (config-if)

Command History

Release	Modification
Cisco IOS 15.5(2)T	This command was introduced.
Cisco IOS XE Release 3.15S	This command was integrated into Cisco IOS XE Release 3.15S.

Usage Guidelines

The global SGT caching configuration and the interface-specific ingress configuration are mutually exclusive. If an interface has ingress SGT caching enabled using the **cts role-based sgt-cache ingress** command in interface configuration mode, and a global configuration is attempted using the **cts role-based sgt-caching** command, the following message is displayed:

```
There is at least one interface that has ingress sgt caching configured. Please remove all
 interface ingress sgt caching configuration(s) before attempting global enable.
```

When an interface is configured to be on a Virtual Routing and Forwarding (VRF) network, the IP-SGT bindings identified on that interface are added under the specific VRF.

Example

The following example shows how to configure SGT caching on an interface:

```
Device> enable
Device# configure terminal
Device(config)# interface gigabitEthernet 0/1/0
Device(config-if)# cts role-based sgt-cache ingress
Device(config-if)# end
```

The following example shows how to disable SGT caching on an interface when SGT caching is enabled globally:

```
Device> enable
Device# configure terminal
Device(config)# cts role-based sgt-caching
Device(config)# interface gigabitEthernet 0/1/0
Device(config-if)# no cts role-based sgt-cache ingress
Device(config-if)# end
```

Related Commands

Command	Description
cts role-based sgt-caching	Enables SGT caching in ingress direction for all interfaces.
interface	Specifies the interface for network traffic.
show cts role-based sgt-map all	Displays the IP-SGT binding table.

cts role-based sgt-caching

To enable Security Group Tag (SGT) caching in ingress direction for all interfaces, use the **cts role-based sgt-caching** command in global configuration mode. To disable SGT caching, use the **no** form of this command.

cts role-based sgt-caching

no cts role-based sgt-caching

Syntax Description

This command has no arguments or keywords.

Command Default

SGT caching is enabled globally.

Command Modes

Global configuration (config)

Command History

Release	Modification
Cisco IOS 15.5(2)T	This command was introduced.
Cisco IOS XE Release 3.15S	This command was integrated into Cisco IOS XE Release 3.15S.

Usage Guidelines

Cisco TrustSec uses SGT caching to ensure that the network traffic tagged with SGT can pass through services that cannot propagate SGTs.

SGT caching can be enabled globally or on an interface. The global SGT caching configuration and the interface-specific ingress configuration are mutually exclusive. If global configuration is enabled using the **cts role-based sgt-caching** command, and an interface configuration is attempted using the **cts role-based sgt-cache ingress** command in interface configuration mode, the following message is displayed:

```
Note that ingress sgt caching is already active on this interface due to global sgt-caching
 enable.
```

Example

The following example shows how to configure SGT caching globally:

```
Device> enable
Device# configure terminal
Device(config)# cts role-based sgt-caching
Device(config)# end
```

Related Commands

Command	Description
cts role-based sgt-cache	Enables SGT caching on an interface.
show cts role-based sgt-map all	Displays the IP-SGT binding table.

cts role-based sgt-map (config)

To assign an Security Group Tag (SGT) value to hosts of an IPv4 or IPv6 network, a VLAN instance, or a VRF instance, use the **cts role-based sgt-map** command in global configuration mode. To remove the SGT value, use the **no** form of this command.

cts role-based sgt-map {[**vrf** *vrf-name*] {*ipv4-address ipv4-address/prefix ipv6-address ipv6-address/prefix* | **host** {*ipv4-address ipv6-address*}} | **vlan-list** {*vlan-id* | **all**}} **sgt** *sgt-value*

no cts role-based sgt-map {[**vrf** *vrf-name*] {*ipv4-address ipv4-address/prefix ipv6-address ipv6-address/prefix* | **host** {*ipv4-address ipv6-address*}} | **vlan-list** {*vlan-id* | **all**}} **sgt** *sgt-value*

Syntax Description

vrf *vrf-name*	Specifies a VRF instance.
ipv4-address	The IPv4 address for a single host.
ipv4-address/prefix	The IPv4 address for all hosts within the specified subnet.
ipv6-address	The IPv6 address for a single host.
ipv6-address/prefix	The IPv6 address for all hosts within the specified subnet.
host {*ipv4-address* | *ipv6-address*}	Specifies the IPv4 or IPv6 address for the host IP-SGT binding.
vlan-list *vlan-id*	Specifies a VLAN ID. The VLAN ID values range from 1 to 4094. Individual VLAN IDs are separated by commas, a range of IDs is specified with a hyphen.
all	Specifies all VLAN instances.
sgt *sgt-value*	Specifies the SGT. The SGT values range from 2 to 65519.

Command Default

SGT value is not assigned.

Command Modes

Global configuration (config)

Command History

Release	Modification
15.0(2)SE	This command was introduced.
15.2(2)E	This command was modified. The *ipv6-address* and *ipv6-address/prefix* arguments were added.

Usage Guidelines

If you do not have a Cisco Identity Services Engine (ISE), Cisco Secure ACS, dynamic ARP inspection, DHCP snooping, or Host Tracking available to your device to automatically map SGTs to source IP addresses, you can manually map an SGT to the following with the **cts role-based sgt-map** command:

- A single host IPv4 or IPv6 address
- All hosts of an IPv4 or IPv6 network or subnetwork
- VRFs
- Single or multiple VLANs

The **cts role-based sgt-map host** command binds the specified SGT with incoming packets when the IP source address is matched by the specified host address. This IP-SGT binding has the lowest priority and is ignored in the presence of any other dynamically discovered bindings from other sources (such as, SXP or locally authenticated hosts).

The **cts role-based sgt-map** *ipv4-address ipv4-address/prefix* and **cts role-based sgt-map** *ipv6-address ipv6-address/prefix* commands bind the specified SGT with packets that fall within the specified network address.

SXP exports an exhaustive expansion of all possible individual IP-SGT bindings within the specified network or subnetwork. IPv6 bindings and subnet bindings are exported only to SXP listener peers of SXP version 2 or later.

The **vrf** keyword specifies a Virtual Routing and Forwarding table previously defined with the **vrf definition** global configuration command. The configuration of VRF contexts is outside the scope of this document. The IP-SGT binding specified with the **cts role-based sgt-map vrf** global configuration command command is entered into the IP-SGT table associated with the specified VRF and the IP protocol version which is implied by the type of IP address entered.

The **cts role-based** sgt-map vlan-list command binds an SGT with a specified VLAN or a set of VLANs. The keyword **all** is equivalent to the full range of VLANs supported by the device and is not preserved in the nonvolatile generation (NVGEN) process. The specified SGT is bound to incoming packets received in any of the specified VLANs.

Example

The following example shows how to assign an SGT value of 5 to an IPv6 address:

```
Device> enable
Device# configure terminal
Device(config)# cts role-based sgt-map sgt-map 2001:DB8:: sgt 5
Device(config)# end
```

The following example shows how to assign an SGT value of 5 to an IP address that falls within an IPv4 network of 10.0.0.0/8:

```
Device> enable
Device# configure terminal
Device(config)# cts role-based sgt-map sgt-map 10.0.0.0/8 sgt 5
Device(config)# end
```

The following example shows how to assign an SGT value of 5 to a VRF instance:

```
Device> enable
Device# configure terminal
Device(config)# cts role-based sgt-map vrf vrfname 10.2.2.3 sgt 5
Device(config)# end
```

The following example shows how to assign an SGT value of 5 to a VLAN:

```
Device> enable
Device# configure terminal
Device(config)# cts role-based sgt-map vlan-list 2 sgt 5
Device(config)# end
```

Related Commands

Command	Description
show cts role-based sgt-map	Displays the IP-SGT binding table.

cts role-based sgt-map interface

To manually map a source IP address to a Security Group Tag (SGT) on either a host or a VRF, use the **cts role-based sgt-map interface** command in global configuration mode. Use the **no** form of the command to remove the mapping.

cts role-based sgt-map *interface-type slot/port* {**security-group** | **sgt**} *sgt-number*
no cts role-based sgt-map interface *interface-type slot/port* {**security-group** | **sgt**} *sgt-number*

Syntax Description

interface-type	Specifies the type of interface. For example, ethernet. The specified SGT is mapped to traffic from this logical or physical Layer 3 interface.
slot/port	Specifies the interface slot and port number.
sgt *sgt-number*	Specifies the SGT number from 0-65535.

Command Default

None

Command Modes

Global configuration (config)

Command History

Release	Modification
15.0(0)SY	This command was introduced on the Catalyst 6500 series switches.
15.1(1)SY	This command was integrated into Cisco IOS Release 15.1(1)Y.
15.3(2)T	This command was integrated into Cisco IOS Release 15.3(2)T.

Usage Guidelines

The **cts role-based sgt-map interface** command binds a specified Layer 3 logical interface to a security group name or to an SGT. A security group information table that maps SGTs to security group names is downloaded from the authentication server with the TrustSec environment data. The **cts role-based sgt-map interface security-group** command is rejected if a security group name table is not available.

Whenever a security group table is downloaded for the first time or refreshed, all L3IF mappings are reprocessed. IP-SGT bindings are added, updated, or deleted for all network prefixes that have output paths through the specified interface.

Note The **interface** keyword is not supported on the Cisco ASR 1000 series routers.

When configuring this command on a Cisco ASR 1000 series router, use the following syntax: **cts role-based sgt-map** {*ipv4-address* | *ipv6-address* | *host-ip-address* | *vrf*} {**security-group** | **sgt**} *sgt-number*.

Examples

The following example shows how to manually map a source IP address to an SGT on a Catalyst 6500 series switch:

```
Device(config)# cts role-based sgt-map interface gigabitEthernet 1/1 sgt 77
```

The following example shows how to manually map a source IP address to an SGT on a Cisco ASR 1000 series router:

```
Device(config)# cts role-based sgt-map 10.10.1.1 sgt 77
```

Related Commands

Command	Description
cts sxp	Configures SXP on a network device.
cts sgt	Configures local device security group tag.
show cts role-based sgt-map	Displays role-based access control information.

cts role-based sgt-map sgt

To bind all traffic on a Layer 3 ingress interface to a security group tag (SGT), use the **cts role-based sgt-map sgt** command in interface configuration mode. To remove the mapping, use the **no** form of this command.

cts role-based sgt-map sgt *sgt-number*
no cts role-based sgt-map sgt *sgt-number*

Syntax Description

sgt-number	SGT number from 2 to 65519.

Command Default

The traffic on a Layer 3 interface is not mapped to an SGT.

Command Modes

Interface configuration (config-if)

Command History

Release	Modification
15.4(2)T	This command was introduced on Cisco Integrated Services Router Generation 2 (Cisco ISR G2).
Cisco IOS XE Release 3.12S	This command was implemented on the Cisco ASR 1000 Series Routers.

Usage Guidelines

The **cts role-based sgt-map sgt** command binds a logical Layer 3 ingress interface to an SGT. Once the mapping is implemented, Cisco TrustSec uses the SGT to segregate traffic from various Layer 3 ingress interfaces.

The SGT is assigned to all traffic on the Layer 3 ingress interface and can be used for inline tagging and policy enforcement.

Examples

The following example shows how to map a Layer 3 ingress interface to an SGT:

```
Device> enable
Device# configure terminal
Device(config)# interface gigabitEthernet 0/0
Device(config-if)# cts role-based sgt-map sgt 77
Device(config-if)# end
```

Related Commands

Command	Description
show cts role-based sgt-map	Displays the IP-SGT binding table.

cts sxp connection peer

Use the **cts sxp connection peer** command in global configuration mode to specify

- the Cisco TrustSec Security Group Tag (SGT) Exchange Protocol (CTS-SXP) peer IP address
- if a password is used for the peer connection or a TCP key-chain should be used to provide TCP-AO authentication
- the global hold-time period for a listener or speaker device
- if the connection is bidirectional.

To remove these configurations for a peer connection, use the **no** form of this command.

cts sxp connection peer *ipv4-address* {**source** | **password**} {**default** | **key-chain** | **none**} **mode** {**local** | **peer**} {{**listener** | **speaker**} [**hold-time** *minimum-time maximum-time*] [**vrf** *vrf-name*] | **both** [**vrf** *vrf-name*]}

no cts sxp connection peer *ipv4-address* {**source** | **password**} {**default** | **key-chain** | **none**} **mode** {**local** | **peer**} {{**listener** | **speaker**} [**hold-time** *minimum-time maximum-time*] [**vrf** *vrf-name*] | **both** [**vrf** *vrf-name*]}

Syntax Description

ipv4-address	SXP peer IPv4 address.
source	Specifies the source IPv4 address.
password	Specifies that an SXP password is used for the peer connection.
default	Specifies that the default SXP password is used.
key-chain	Specifies that the TCP-AO key-chain should be used to authenticate TCP segments.
none	Specifies no password is used.
mode	Specifies either the local or peer SXP connection mode.
local	Specifies that the SXP connection mode refers to the local device.
peer	Specifies that the SXP connection mode refers to the peer device.
listener	(Optional) Specifies that the device is the listener in the connection.
speaker	(Optional) Specifies that the device is the speaker in the connection.

hold-time *minimum-time maximum-time*	(Optional) Specifies the hold-time period, in seconds, for the device. The range for minimum and maximum time is from 0 to 65535. A *maximum-time* value is required only when you use the following keywords: **peer speaker** and **local listener**. In other instances, only a *minimum-time* value is required. **Note** If both minimum and maximum times are required, the *maximum-time* value must be greater than or equal to the *minimum-time* value.
vrf *vrf-name*	(Optional) Specifies the virtual routing and forwarding (VRF) instance name to the peer.
both	(Optional) Specifies that the device is both the speaker and the listener in the bidirectional SXP connection.

Command Default

The CTS-SXP peer IP address is not configured and no CTS-SXP peer password is used for the peer connection.

The default setting for a CTS-SXP connection password is **none**.

Command Modes

Global configuration (config)

Command History

Release	Modification
12.2(33)SXI3	This command was introduced on the Catalyst 6500 series switches.
12.2(50)SG7	This command was integrated on the Catalyst 4000 series switches.
12.2(53)SE2	This command was integrated into Cisco IOS Release 12.2(53)SG7 on the Catalyst 3750(E) and 3560(E) series switches.
Cisco IOS XE Release 3.4S	This command was integrated into Cisco IOS XE Release 3.4S.
15.1(3)S	This command was integrated into Cisco IOS Release 15.1(3)S.
15.3(2)T	This command was modified. The **hold-time** keyword and *minimum-time* and *maximum-time* arguments were added.
Cisco IOS XE Release 3.11S	This command was modified. The **both** keyword was added.
15.4(1)T	This command was modified. The **both** keyword was added.
16.12.1	This command was modified. The **key-chain** keyword was added.

Usage Guidelines

When a CTS-SXP connection to a peer is configured with the **cts sxp connection peer** command, only the connection mode can be changed. The **vrf** keyword is optional. If a VRF name is not provided or a VRF name is provided with the **default** keyword, then the connection is set up in the default routing or forwarding domain.

A **hold-time** *maximum-period* value is required only when you use the following keywords: **peer speaker** and **local listener**. In other instances, only a **hold-time** *minimum-period* value is required.

Note The *maximum-period* value must be greater than or equal to the *minimum-period* value.

Use the **both** keyword to configure a bidirectional SXP connection. With the support for bidirectional SXP configuration, a peer can act as both a speaker and a listener and propagate SXP bindings in both directions using a single connection.

Use the **key-chain** keyword to specify that TCP-AO should be used to authenticate the TCP segments exchanged by the SXP peers. You must define the default key-chain to use for TCP-AO using **cts sxp default key-chain**.

Examples

The following example shows how to enable CTS-SXP and configure the CTS-SXP peer connection on Device_A, a speaker, for connection to Device_B, a listener with the password option for TCP MD5 authentication: :

```
Device_A> enable
Device_A# configure terminal
Device_A#(config)# cts sxp enable
Device_A#(config)# cts sxp default password Cisco123
Device_A#(config)# cts sxp default source-ip 10.10.1.1
Device_A#(config)# cts sxp connection peer 10.20.2.2 password default mode local speaker
```

The following example shows how to configure the CTS-SXP peer connection on Device_B, a listener, for connection to Device_A, a speaker:

```
Device_B> enable
Device_B# configure terminal
Device_B(config)# cts sxp enable
Device_B(config)# cts sxp default password Cisco123
Device_B(config)# cts sxp default source-ip 10.20.2.2
Device_B(config)# cts sxp connection peer 10.10.1.1 password default mode local listener
```

You can also configure both peer and source IP addresses for an SXP connection. The source IP address specified in the **cts sxp connection** command overwrites the default value.

The following example shows how to configure the CTS-SXP peer connection on Device_A, a speaker, for connection to Device_B, a listener without a password or key chain option:

```
Device_A(config)# cts sxp connection peer 51.51.51.1 source 51.51.51.2 password none mode
local speaker
```

```
Device_B(config)# cts sxp connection peer 51.51.51.2 source 51.51.51.1 password none mode
local listener
```

The following example shows how to enable bidirectional CTS-SXP and configure the SXP peer connection on Device_A to connect to Device_B:

```
Device_A> enable
Device_A# configure terminal
Device_A#(config)# cts sxp enable
Device_A#(config)# cts sxp default password Cisco123
Device_A#(config)# cts sxp default source-ip 10.10.1.1
Device_A#(config)# cts sxp connection peer 10.20.2.2 password default mode local both
```

The following example shows how to enable CTS-SXP and configure a CTS-SXP peer connection with TCP-AO authentication on Device_A, a speaker, for connection to Device_B, a listener:

```
Device_A> enable
Device_A# configure terminal
Device_A#(config)# cts sxp enable
Device_A#(config)# cts sxp default key-chain sxp_1
Device_A#(config)# cts sxp connection peer 2.2.2.2 password key-chain mode local speaker
hold-time 0
```

Related Commands

Command	Description
cts sxp default password	Configures the Cisco TrustSec SXP default password.
cts sxp default key-chain	Configures the default key-chain to use for TCP-AO authentication.
cts sxp default source-ip	Configures the Cisco TrustSec SXP source IPv4 address.
cts sxp enable	Enables Cisco TrustSec SXP on a device.
cts sxp log	Enables logging for IP-to-SGT binding changes.
cts sxp reconciliation	Changes the Cisco TrustSec SXP reconciliation period.
cts sxp retry	Changes the Cisco TrustSec SXP retry period timer.
cts sxp speaker hold-time	Configures the global hold-time period of a speaker device in a Cisco TrustSec SGT SXPv4 network.
cts sxp listener hold-time	Configures the global hold-time period of a listener device in a Cisco TrustSec SGT SXPv4 network.
show cts sxp	Displays the status of all Cisco TrustSec SXP configurations.

cts sxp default key-chain

To specify the Cisco TrustSec Security Group Tag (SGT) Exchange Protocol (CTS-SXP) default key-chain for TCP-AO, use the **cts sxp default key-chain** command in global configuration mode. To remove the CTS-SXP default key-chain, use the **no** form of this command.

cts sxp default key-chain *key-chain-name*
no cts sxp default key-chain *key-chain-name*

Syntax Description

key-chain-name	Name of the TCP key-chain that must be used by default to provide TCP-AO authentication for CTS SXP sessions.

Command Default

A default key chain is not configured for CTS SXP.

Command Modes

Privileged EXEC (#)

Command History

Release	Modification
16.12.1	Command introduced

Usage Guidelines

Use this command to specify the key-chain that must be used by default to provide TCP-AO authentication for CTS SXP sessions.

Define the key-chain using the **key chain** *key-chain-name* **tcp** command.

Example

In the following example, a TCP-AO key chain named sxp_1 is configured as the default key chain for CTS SXP sessions using TCP-AO.

```
Device> enable
Device# configure terminal
Device(config)# cts sxp default key-chain sxp_1
```

Related Commands

Command	Description
key chain *key-chain-name* **tcp**	Use this command to define a key-chain for TCP-AO.

cts sxp default password

To specify the Cisco TrustSec Security Group Tag (SGT) Exchange Protocol (CTS-SXP) default password, use the **cts sxp default password** command in global configuration mode. To remove the CTS-SXP default password, use the **no** form of this command.

cts sxp default password {**0** *unencrypted-pwd* | **6** *encrypted-key* | **7** *encrypted-keycleartext-pwd*}
no cts sxp default password {**0** *unencrypted-pwd* | **6** *encrypted-key* | **7** *encrypted-keycleartext-pwd*}

Syntax Description

0 *unencrypted-pwd*	Specifies that an unencrypted CTS-SXP default password follows. The maximum password length is 32 characters.
6 *encrypted-key*	Specifies that a 6 encryption type password is used as the CTS-SXP default password. The maximum password length is 32 characters.
7 *encrypted-key*	Specifies that a 7 encryption type password is used as the CTS-SXP default password. The maximum password length is 32 characters.
cleartext-pwd	Specifies a cleartext CTS-SXP default password. The maximum password length is 32 characters.

Command Default

Type **0** (cleartext)

Command Modes

Global configuration (config)

Command History

Release	Modification
12.2(33)SXI3	This command was introduced on the Catalyst 6500 series switches.
12.2(50)SG7	This command was integrated on the Catalyst 4000 series switches.
12.2(53)SE2	This command was integrated into Cisco IOS Release 12.2(53)SG7 on the Catalyst 3750(E) and 3560(E) series switches.
Cisco IOS XE Release 3.4S	This command was integrated into Cisco IOS XE Release 3.4S.
15.1(3)S	This command was integrated into Cisco IOS Release 15.1(3)S.

Usage Guidelines

The **cts sxp default password** command sets the CTS-SXP default password to be optionally used for all CTS-SXP connections configured on the device. The CTS-SXP password can be cleartext, or encrypted with the **0**, **7**, **6** encryption type keywords. If the encryption type is 0, then an unencrypted cleartext password follows.

Examples

The following example shows how to enable CTS-SXP and configure the CTS-SXP peer connection on Router_A, a speaker, for connection to Router_B, a listener:

```
Router_A# configure terminal
Router_A#(config)# cts sxp enable
```

```
Router_A#(config)# cts sxp default password Cisco123
Router_A#(config)# cts sxp default source-ip 10.10.1.1
Router_A#(config)# cts sxp connection peer 10.20.2.2 password default mode local speaker
```

The following example shows how to configure the CTS-SXP peer connection on Router_B, a listener, for connection to Router_A, a speaker:

```
Router_B# configure terminal
Router_B(config)# cts sxp enable
Router_B(config)# cts sxp default password Cisco123
Router_B(config)# cts sxp default source-ip 10.20.2.2
Router_B(config)# cts sxp connection peer 10.10.1.1 password default mode local listener
```

Related Commands

Command	Description
cts sxp connection peer	Enters the CTS-SXP peer IP address and specifies if a password is used for the peer connection.
cts sxp default source-ip	Configures the CTS-SXP source IPv4 address.
cts sxp enable	Enables CTS-SXP on a device.
cts sxp log	Enables logging for IP-to-SGT binding changes.
cts sxp reconciliation	Changes the CTS-SXP reconciliation period.
cts sxp retry	Changes the CTS-SXP retry period timer.
show cts sxp	Displays the status of all SXP configurations.

cts sxp default source-ip

To configure the Cisco TrustSec Security Group Tag (SGT) Exchange Protocol (CTS-SXP) source IPv4 address, use the **cts sxp default source-ip** command in global configuration mode. To remove the CTS-SXP default source IP address, use the **no** form of this command.

cts sxp default source-ip *ipv4-address*
no cts sxp default source-ip *ipv4-address*

Syntax Description

ip-address	Default source CTS-SXP IPv4 address.

Command Default

The CTS-SXP source IP address is not configured.

Command Modes

Global configuration (config)

Command History

Release	Modification
12.2(33)SXI3	This command was introduced on the Catalyst 6500 series switches.
12.2(50)SG7	This command was integrated on the Catalyst 4000 series switches.
12.2(53)SE2	This command was integrated into Cisco IOS Release 12.2(53)SG7 on the Catalyst 3750(E) and 3560(E) series switches.
Cisco IOS XE Release 3.4S	This command was integrated into Cisco IOS XE Release 3.4S.
15.1(3)S	This command was integrated into Cisco IOS Release 15.1(3)S.

Usage Guidelines

The **cts sxp default source-ip** command sets the default source IP address that CTS-SXP uses for all new TCP connections where a source IP address is not specified. Preexisting TCP connections are not affected when this command is entered. CTS-SXP connections are governed by three timers:

- Retry timer
- Delete Hold Down timer
- Reconciliation timer

Examples

The following example shows how to enable CTS-SXP and configure the CTS-SXP peer connection on Router_A, a speaker, for connection to Router_B, a listener:

```
Router_A# configure terminal
Router_A#(config)# cts sxp enable
Router_A#(config)# cts sxp default password Cisco123
Router_A#(config)# cts sxp default source-ip 10.10.1.1
Router_A#(config)# cts sxp connection peer 10.20.2.2 password default mode local speaker
```

The following example shows how to configure the CTS-SXP peer connection on Router_B, a listener, for connection to Router_A, a speaker:

```
Router_B# configure terminal
Router_B(config)# cts sxp enable
Router_B(config)# cts sxp default password Cisco123
Router_B(config)# cts sxp default source-ip 10.20.2.2
Router_B(config)# cts sxp connection peer 10.10.1.1 password default mode local listener
```

Related Commands

Command	Description
cts sxp connectionpeer	Enters the CTS-SXP peer IP address and specifies if a password is used for the peer connection.
cts sxp default password	Configures the CTS-SXP default password.
cts sxp enable	Enables CTS-SXP on a device.
cts sxp log	Enables logging for IP-to-SGT binding changes.
cts sxp reconciliation	Changes the CTS-SXP reconciliation period.
cts sxp retry	Changes the CTS-SXP retry period timer.
show cts sxp	Displays the status of all SXP configurations.

cts sxp enable

To enable the Cisco TrustSec Security Group Tag (SGT) Exchange Protocol (CTS-SXP) on a device, use the **cts sxp enable** command in global configuration mode. To disable the CTS-SXP on a device, use the **no** form of this command.

cts sxp enable
no cts sxp enable

Syntax Description

This command has no arguments or keywords.

Command Default

Disabled

Command Modes

Global configuration (config)

Command History

Release	Modification
12.2(33)SXI3	This command was introduced on the Catalyst 6500 series switches.
12.2(50)SG7	This command was integrated on the Catalyst 4000 series switches.
12.2(53)SE2	This command was integrated into Cisco IOS Release 12.2(53)SG7 on the Catalyst 3750(E) and 3560(E) series switches.
Cisco IOS XE Release 3.4S	This command was integrated into Cisco IOS XE Release 3.4S.
15.1(3)S	This command was integrated into Cisco IOS Release 15.1(3)S.

Usage Guidelines

The **cts sxp enable** command enables CTS-SXP over a TCP (SXP) connection. CTS-SXP propagates IP-to-SGT binding information across network devices that do not have the capability to tag packets, which allows security services on switches, routers or firewalls to learn identity information from devices that access the network.

Examples

The following example shows how to enable CTS-SXP and configure the SXP peer connection on Router_A, a speaker, for connection to Router_B, a listener:

```
Router_A# configure terminal
Router_A#(config)# cts sxp enable
Router_A#(config)# cts sxp default password Cisco123
Router_A#(config)# cts sxp default source-ip 10.10.1.1
Router_A#(config)# cts sxp connection peer 10.20.2.2 password default mode local speaker
```

The following example shows how to configure the CTS-SXP peer connection on Router_B, a listener, for connection to Router_A, a speaker:

```
Router_B# configure terminal
Router_B(config)# cts sxp enable
Router_B(config)# cts sxp default password Cisco123
Router_B(config)# cts sxp default source-ip 10.20.2.2
Router_B(config)# cts sxp connection peer 10.10.1.1 password default mode local listener
```

Related Commands

Command	Description
cts sxp sxp connectionpeer	Enters the CTS-SXP peer IP address and specifies if a password is used for the peer connection.
cts sxp default password	Configures the CTS-SXP default password.
cts sxp default source-ip	Configures the CTS-SXP source IPv4 address.
cts sxp log	Enables logging for IP-to-SGT binding changes.
cts sxp reconciliation	Changes the CTS-SXP reconciliation period.
show cts sxp	Displays the status of all CTS-SXP configurations.
cts sxp retry	Changes the CTS-SXP retry period timer.

cts sxp filter-enable

To enable filtering after creating filter lists and filter groups, use the **cts sxp filter-enable** command in global configuration mode. To disable filtering, use the **no** form of the command.

cts sxp filter-enable
no cts sxp filter-enable

Syntax Description

This command has no keywords or arguments.

Command Modes

Global configuration (config)

Command History

Release	Modification
16.6.1	This command was introduced.

Usage Guidelines

This command can be used at any time to enable or disable filtering. Configured filter lists and filter groups can be used to implement filtering only after filtering is enabled. The filter action will only filter bindings that are exchanged after filtering is enabled; there won't be any effect on the bindings that were exchanged before filtering was enabled.

Examples

```
Device(config)# cts sxp filter-enable
```

Related Commands

Command	Description
cts sxp filter-list	Creates a SXP filter list to filter IP-SGT bindings based on IP prefixes, SGT or a combination of both.
cts sxp filter-group	Creates a filter group for grouping a set of peers and applying a filter list to them.
show cts sxp filter-group	Displays information about the configured filter groups..
show cts sxp filter-list	Displays information about the configured filter lists.
debug cts sxp filter events	Logs events related to the creation, deletion and update of filter-lists and filter-groups

cts sxp filter-group

To create a filter group for grouping a set of peers and applying a filter list to them, use the **cts sxp filter-group** command in global configuration mode. To delete a filter group, use the **no** form of this command.

cts sxp filter-group {**listener** | **speaker**} [**global**] {*filter-group-name*}
no cts sxp filter-group {**listener** | **speaker**} [**global**] {*filter-group-name*}

Syntax Description

listener	Creates a filter group for a set of listeners.
speaker	Creates a filter group for a set of speakers.
global	Groups all speakers or listeners on the device.
filter-group-name	Name of the filter group.

Command Modes

Global configuration (config)

Command History

Release	Modification
16.6.1	This command was introduced.

Usage Guidelines

Issuing this command, places the device in the filter group configuration mode. From this mode, you can specify the devices to be grouped and apply a filter list to the filter group.

The command format to add devices or peers to the group is a follows:

peer ipv4 *peer-IP*

In a single command, you can add one peer. To add more peers, repeat the command as many times as required.

The command format to apply a filter list to the group is as follows:

filter *filter-list-name*

You cannot specify a peer list for the global listener and global speaker filter-group options because in this case the filter is applied to all SXP connections

When both the global filter group and peer-based filter groups are applied, the global filter takes priority. If only a global listener or global speaker filter group is configured, then the global filtering takes precendence only in that specific direction. For the other direction, the peer-based filter group is implemented.

Examples

The following example shows how to create a listener group called **group_1**, and assign peers and a filter list to this group:

```
Device# configure terminal
Device(config)# cts sxp filter-group listener group_1
Device(config-filter-group)# filter filter_1
Device(config-filter-group)# peer ipv4 10.0.0.1
Device(config-filter-group)# peer ipv4 10.10.10.1
```

The following example shows how to create a global listener group called **group_2**

```
Device# configure terminal
Device(config)# cts sxp filter-group listener global group_2
```

Related Commands

Command	Description
cts sxp filter-list	Creates a SXP filter list to filter IP-SGT bindings based on IP prefixes, SGT or a combination of both.
cts sxp filter-enable	Enables filtering.
show cts sxp filter-group	Displays information about the configured filter groups.
show cts sxp filter-list	Displays information about the configured filter lists.
debug cts sxp filter events	Logs events related to the creation, deletion and update of filter-lists and filter-groups

cts sxp filter-list

To create a SXP filter list to hold a set of filter rules for filtering IP-SGT bindings, use the **cts sxp filter-list** command in global configuration mode. To delete a filter list, use the **no** form of the command.

cts sxp filter-list *filter-list-name*
no cts sxp filter-list *filter-list-name*

Syntax Description

filter-list-name	Name of the filter-list.

Command Modes

Global configuration (config)

Command History

Release	Modification
16.6.1	This command was introduced.

Usage Guidelines

Issuing this command, places the device in the filter list configuration mode. From this mode, you can specify rules for the filter lists.

A filter rule can be based on SGT or IP Prefixes or a combination of both SGT and IP Prefixes.

The command format to add rules to the group is a follows:

sequence-number **action(permit/deny) filter-type(ipv4/ipv6/sgt)** *value/values*

For example, to permit SGT-IP bindings whose SGT value is 20, the rule is as follows:

```
30 permit sgt 20
```

Note that the sequence number is optional. If you do not specify a sequence number, it is generated by the system. Sequence numbers are automatically incremented by a value of 10 from the last used/configured sequence number. A new rule can be inserted by specifying a sequence number in between two existing rules.

The range of valid SGT values is between 2 and 65519. To provide multiple SGT values in a rule, seperate the values using a space. A maximum of 8 SGT values are allowed in a rule.

In a SGT and IP prefix combination rule, if there is a match for the binding in both the parts of the rule, then the action specified in the second part of the rule takes precedence. For example, in the following rule, if the SGT value of the IP prefix 10.0.0.1 is 20, the corresponding binding will be denied even if the first part of the rule permits the binding.

```
Router(config-filter-list)# 10 permit sgt 30 20 deny 10.0.0.1/24
```

Similarly, in the rule below the binding with the sgt value 20 will be permitted even if the sgt of the IP prefix 10.0.0.1 is 20, and the first action does not permit the binding.

```
Router(config-filter-list)# 10 deny 10.0.0.1/24 permit sgt 30 20
```

Examples

The following example shows how to create a filter list and add some rules to the list:

```
Device# configure terminal
Device(config)# cts sxp filter-list filter_1
Device (config-filter-list)# 10 deny ipv4 10.0.0.1/24 permit sgt 100
Device(config-filter-list)# 20 permit sgt 60 61 62 63
```

Related Commands

Command	Description
cts sxp filter-enable	Enable SXP IP-prefix and SGT-based filtering.
cts sxp filter-group	Creates a filter group for grouping a set of peers and applying a filter list to them.
show cts sxp filter-group	Displays information about the configured filter groups.
show cts sxp filter-list	Displays information about the configured filter lists.
debug cts sxp filter events	Logs events related to the creation, deletion and update of filter-lists and filter-groups.

cts sxp listener hold-time

To configure the global hold-time period of a listener network device in a Cisco TrustSec Security Group Tag (SGT) Exchange Protocol version 4 (SXPv4) network, use the **cts sxp listener hold-time** command in global configuration mode. To remove the hold time from the listener device, use the **no** form of this command.

cts sxp listener hold-time *minimum-period maximum-period*
no cts sxp listener hold-time

Syntax Description

minimum-period	Minimum allowed hold time in seconds. The range is from 1 to 65534.
maximum-period	Specifies the maximum allowed hold-time in seconds. The range is from 1 to 65534 seconds. **Note** The *maximum-period* specified must be greater than or equal to the *minimum-period*.

Command Default

The default hold time range for a listener device is 90 seconds to 180 seconds.

Command Modes

Global configuration (config)

Command History

Release	Modification
15.3(2)T	This command was introduced.
Cisco IOS Release 3.9S	This command was modified. Support was added for the Cisco ASR 1000 Series Routers.

Usage Guidelines

SXP uses a TCP-based, keepalive mechanism to determine if a connection is live. SXPv4 adds an optional negotiated keepalive mechanism, the hold-time period, in order to provide more predictable and timely detection of connection loss.

Hold time can be configured globally on a network device. This global configuration will apply the configuration to all SXP connections configured on the device.

You may configure a hold-time period locally on a listener device or a default of 90 seconds to 180 seconds is used. A value of “0xFFFF..0xFFFF” indicates that the keepalive mechanism is not used.

The hold-time negotiation between the speaker device and the listener device succeeds when the speaker’s minimum acceptable hold-time falls below or within the desirable hold-time range of the listener. (Use the **cts sxp speaker hold-time** command to configure the hold-time of the speaker device.) If one end turns off the keepalive mechanism, the other end should also turn it off to make the negotiation successful.

The negotiation fails when the speaker's minimum acceptable hold-time is greater than the upper bound of the listener's hold-time range.

The selected hold-time period of a successful negotiation is the maximum of the speaker’s minimum acceptable hold-time and the lower bound of the listener’s hold-time range.

The speaker calculates the keepalive time to one-third of the selected hold time by default, unless a different keepalive time is locally configured.

The following example shows how to configure the hold time period of a listener device for a minimum of 300 seconds and a maximum of 500 seconds:

```
Device> enable
Device# configure terminal
Device(config)# cts sxp listener hold-time 300 500
```

Related Commands

Command	Description
cts sxp enable	Enables Cisco TrustSec SXP on a device.
cts sxp speaker hold-time	Configures the hold time of a speaker device in an SXPv4 network.
show cts sxp	Displays the status of all Cisco TrustSec SXP configurations.

cts sxp log binding-changes

To enable logging for IP-to-Cisco TrustSec Security Group Tag (SGT) Exchange Protocol (CTS-SXP) binding changes, use the **cts sxp log binding-changes** command in global configuration mode. To disable logging, use the **no** form of this command.

cts sxp log binding-changes
no cts sxp log binding-changes

Command Default

Logging disabled.

Command Modes

Global configuration (config)

Command History

Release	Modification
12.2(33)SXI3	This command was introduced on the Catalyst 6500 series switches.
12.2(50)SG7	This command was integrated on the Catalyst 4000 series switches.
12.2(53)SE2	This command was integrated into Cisco IOS Release 12.2(53)SG7 on the Catalyst 3750(E) and 3560(E) series switches.
Cisco IOS XE Release 3.4S	This command was integrated into Cisco IOS XE Release 3.4S.
15.1(3)S	This command was integrated into Cisco IOS Release 15.1(3)S.

Usage Guidelines

The **cts sxp log binding-changes** command enables logging for IP-to-SGT binding changes. SXP syslogs (sev 5 syslogs) are generated whenever IP address-to-SGT binding occurs (add, delete, change). These changes are learned and propagated on the SXP connection.

Related Commands

Command	Description
cts sxp connectionpeer	Enters the CTS-SXP peer IP address and specifies if a password is used for the peer connection
cts sxp default password	Configures the CTS-SXP default password.
cts sxp default source-ip	Configures the CTS-SXP source IPv4 address.
cts sxp enable	Enables CTS-SXP on a device.
cts sxp reconciliation	Changes the CTS-SXP reconciliation period.
cts sxp retry	Changes the CTS-SXP retry period timer.
show cts sxp	Displays status of all SXP configurations.

cts sxp mapping network-map

To configure the subnet to Security Group Tag (SGT) mapping host count constraint to limit the number of subnet bindings SXPv3 can export, use the **cts sxp mapping network-map** command in global configuration mode. To return to the default, use the **no** form of this command.

cts sxp mapping network-map *bindings*

Syntax Description

bindings	Specifies the maximum number of subnet IP hosts from 0 to 65,535 that can be bound to SGTs and exported to the SXP listener.

Command Default

The default is 0 (no expansions performed).

Command Modes

Global configuration (config)

Command History

Release	Modification
15.1(1)SY	This command was introduced.

Usage Guidelines

In IPv4 networks, SXPv3, and more recent versions, can receive and parse subnet network address/prefix strings from SXPv3 peers. Earlier SXP versions convert the subnet prefix into its set of host bindings before exporting them to an SXP listener peer.

For example, the IPv4 subnet 198.1.1.0/29 is expanded as follows (only 3 bits for host addresses):

- Host addresses 198.1.1.1 to 198.1.1.7 are tagged and propagated to SXP peer.
- Network and broadcast addresses 198.1.1.0 and 198.1.1.8 are not tagged and not propagated.

Subnet bindings are static, which means that active hosts are not learned. They can be used locally for SGT imposition and SGACL enforcement. Packets tagged by subnet to SGT mapping can be propagated on Layer 2 or Layer 3 TrustSec links.

Note For IPv6 networks, SXPv3 cannot export subnet bindings to SXPv2 or SXPv1 peers.

Examples

```
Device(config)# cts role-based sgt-map 10.10.10.10/29 sgt 1234
```

Related Commands

Command	Description
cts role-based	Manually maps a source IP address to a SGT on either a host or a VRF.

cts sxp node-id

To configure the node ID of a network device for Cisco TrustSec (CTS) Security Group Tag (SGT) Exchange Protocol version 4 (SXPv4), use the **cts sxp node-id** command in global configuration mode. To remove the node ID, use the **no** form of this command.

cts sxp node-id {*node-id* | **interface** *interface-type* | *ipv4-address*}
no cts sxp node-id

Syntax Description

node-id	Specifies the node ID of the device. Enter the node ID in hexadecimal format.
interface *interface-type*	Specifies the type of interface.
ipv4-address	Specifies the SXP peer IPv4 address.

Command Default

A node ID is not configured.

Command Modes

Global configuration (config)

Command History

Release	Modification
15.3(2)T	This command was introduced.
Cisco IOS Release 3.9S	This command was modified. Support was added for the Cisco ASR 1000 Series Routers.

Usage Guidelines

The **cts sxp node-id** command configures the node ID of a network device.

An SXP node ID is used to identify the individual devices within the network. The node ID is a four-octet integer that can be configured by the user. If it is not configured by the user, SXP picks a node ID itself using the highest IPv4 address in the default VRF domain, in the same manner that EIGRP generates its node ID.

The node ID has to be unique in the network that SXP connections traverse to enable SXP loop prevention.

The SXP loop detection mechanism drops the binding propagation packets based on finding its own node ID in the peer sequence attribute. Changing a node ID in a loop detection running SXP network could break SXP loop detection functionality and therefore needs to be handled carefully.

Wait until the SXP bindings that are propagated with the particular node ID in the path attribute are deleted, before you change the node ID.

Note A syslog is generated when you change the node ID.

```
Device(config)# cts sxp node-id 172.16.1.3
```

Related Commands

Command	Description
cts sxp enable	Enables CTS-SXP on a device.
show cts sxp	Displays the status of all CTS-SXP configurations.

cts sxp reconciliation period

To change the Cisco TrustSec Security Group Tag (SGT) Exchange Protocol (CTS-SXP) reconciliation period, use the **cts sxp reconciliation period** command in global configuration mode. To return the CTS-SXP reconciliation period to its default value, use the **no** form of this command.

cts sxp reconciliation period *seconds*
no cts sxp reconciliation period *seconds*

Syntax Description

seconds	CTS-SXP reconciliation timer in seconds. The range is from 0 to 64000. The default is 120.

Command Default

120 seconds (2 minutes)

Command Modes

Global configuration (config)

Command History

Release	Modification
12.2(33)SXI3	This command was introduced on the Catalyst 6500 series switches.
12.2(50)SG7	This command was integrated on the Catalyst 4000 series switches.
12.2(53)SE2	This command was integrated into Cisco IOS Release 12.2(53)SG7 on the Catalyst 3750(E) and 3560(E) series switches.
Cisco IOS XE Release 3.4S	This command was integrated into Cisco IOS XE Release 3.4S.
15.1(3)S	This command was integrated into Cisco IOS Release 15.1(3)S.

Usage Guidelines

After a peer terminates a CTS-SXP connection, an internal Delete Hold-down timer starts. If the peer reconnects before the Delete Hold-down timer expires, then the CTS-SXP Reconciliation timer starts. While the CTS-SXP Reconciliation period timer is active, the CTS-SXP software retains the SGT mapping entries learned from the previous connection and removes invalid entries. Setting the SXP reconciliation period to 0 seconds disables the timer and causes all entries from the previous connection to be removed.

Related Commands

Command	Description
cts sxp connection peer	Enters the CTS-SXP peer IP address and specifies if a password is used for the peer connection.
cts sxp default password	Configures the CTS-SXP default password.
cts sxp default source-ip	Configures the CTS-SXP source IPv4 address.
cts sxp enable	Enables CTS-SXP on a device.
cts sxp log	Turns on logging for IP to SGT binding changes.
cts sxp retry	Changes the CTS-SXP retry period timer.

Command	Description
show cts sxp	Displays status of all CTS-SXP configurations.

cts sxp retry period

To change the Cisco TrustSec Security Group Tag (SGT) Exchange Protocol (CTS-SXP) retry period timer, use the **cts sxp retry period** command in global configuration mode. To return the CTS-SXP retry period timer to its default value, use the **no** form of this command.

cts sxpretry period *seconds*
no cts sxpretry period *seconds*

Syntax Description

seconds	CTS-SXP retry timer in seconds. The range is from 0 to 64000. The default is 120.

Command Default

120 seconds (2 minutes)

Command Modes

Global configuration (config)

Command History

Release	Modification
12.2(33)SXI3	This command was introduced on the Catalyst 6500 series switches.
12.2(50)SG7	This command was integrated on the Catalyst 4000 series switches.
12.2(53)SE2	This command was integrated into Cisco IOS Release 12.2(53)SG7 on the Catalyst 3750(E) and 3560(E) series switches.
Cisco IOS XE Release 3.4S	This command was integrated into Cisco IOS XE Release 3.4S.
15.1(3)S	This command was integrated into Cisco IOS Release 15.1(3)S.

Usage Guidelines

The retry timer is triggered if there is at least one CTS-SXP connection that is not up. A new CTS-SXP connection is attempted when this timer expires. A zero value results in no retry being attempted.

Related Commands

Command	Description
cts sxp connectionpeer	Enters the CTS-SXP peer IP address and specifies if a password is used for the peer connection.
cts sxp default password	Configures the CTS-SXP default password.
cts sxp default source-ip	Configures the CTS-SXP source IPv4 address.
cts sxp enable	Enables CTS-SXP on a device.
cts sxp log	Enables logging for IP-to-SGT binding changes.
cts sxp reconciliation	Changes the CTS-SXP reconciliation period.
show cts sxp	Displays the status of all CTS-SXP configurations.

cts sxp speaker hold-time

To configure the global hold-time period of a speaker network device in a Cisco TrustSec Security Group Tag (SGT) Exchange Protocol version 4 (SXPv4) network, use the **cts sxp speaker hold-time** command in global configuration mode. To remove the hold time from the speaker device, use the **no** form of this command.

cts sxp speaker hold-time *minimum-period*
no cts sxp speaker hold-time

Syntax Description

minimum-period	Minimum allowed hold time in seconds. The range is from 1 to 65534.

Command Default

The default hold time for a speaker device is 120 seconds.

Command Modes

Global configuration (config)

Command History

Release	Modification
15.3(2)T	This command was introduced.
Cisco IOS Release 3.9S	This command was modified. Support was added for the Cisco ASR 1000 Series Routers.

Usage Guidelines

The Security Group Tag Exchange Protocol (SXP) uses a TCP-based, keepalive mechanism to determine if a connection is live. SXPv4 adds an optional negotiated keepalive mechanism, the hold-time period, in order to provide more predictable and timely detection of connection loss.

Hold time can be configured globally on a network device. This global configuration will apply the configuration to all SXP connections configured on the device.

You may configure a hold-time period locally on a speaker device or a default of 120 seconds is used. This is the shortest period of time a speaker is willing to send keepalive messages for keeping the connection active. Any shorter hold-time period would require a faster keepalive rate than the rate the speaker is ready to support. A value of 0xFFFF indicates that the keepalive mechanism is not used.

The hold-time negotiation between the speaker device and the listener device succeeds when the speaker's minimum acceptable hold time falls below or within the desirable hold-time range of the listener. (Use the **cts sxp listener hold-time** command to configure the hold time of the listener device.) If one end turns off the keepalive mechanism, the other end should also turn it off to make the negotiation successful.

The negotiation fails when the speaker's minimum acceptable hold-time is greater than the upper bound of the listener's hold-time range.

The selected hold-time period of a successful negotiation is the maximum of the speaker's minimum acceptable hold time and the lower bound of the listener's hold-time range.

The speaker calculates the keepalive time to one-third of the selected hold time by default, unless a different keepalive time is locally configured.

The following example shows how to configure the minimum hold time period of a speaker device for 300 seconds:

```
Device(config)# cts sxp speaker hold-time 300
```

Related Commands

Command	Description
cts sxp enable	Enables Cisco TrustSec SXP on a device.
cts sxp listener hold-time	Configures the hold time of a listener device in an SXPv4 network.
show cts sxp	Displays the status of all Cisco TrustSec SXP configurations.

custom-page

To display custom web pages during web authentication login, use the **custom-page** command in parameter map webauth configuration mode. To disable custom web pages, use the **no** form of this command.

custom-page {**failure** | **login** [**expired**] | **success**} **device** *location***:***filename*
no custom-page {**failure** | **login** [**expired**] | **success**} **device** *location***:***filename*

Syntax Description

failure	Displays the custom web page if the login fails.
login	Displays the custom web page during login.
expired	(Optional) Displays the custom web page if the login expires.
success	Displays the custom web page when the login is successful.
location **:***filename*	Location and name of the locally stored HTML file to use in place of the default HTML file for the specified condition.

Command Default

The internal default web pages are displayed.

Command Modes

Parameter map webauth configuration (config-params-parameter-map)

Command History

Release	Modification
Cisco IOS XE Release 3.2SE	This command was introduced.

Usage Guidelines

Use the **custom-page** command to display custom web pages during web authentication login. To enable custom web pages:

- You must specify all four custom HTML files. If fewer than four files are specified, the internal default HTML pages are used.
- The four custom HTML files and any images in the custom pages must be stored in the disk or flash of the switch. The maximum size of each HTML file is 256 KB.
- Filenames must start with web_auth.
- To serve custom pages and images from an external server, you must configure a redirect portal IP address by using the **redirect** (parameter-map webauth) command instead of using local custom pages.
- Any external link from a custom page requires an intercept ACL configuration.
- Any name resolution required for external links or images requires an intercept ACL configuration.
- If the custom web pages feature is enabled, the redirection URL for successful login feature will not be available.
- Because the custom login page is a public web form, consider the following guidelines for this page:
  - The login form must accept user input for the username and password and must POST the data as uname and pwd.

- The custom login page should follow best practices for a web form, such as page timeout, hidden password, and prevention of redundant submissions.

Examples

The following example shows how to configure a named parameter map for web authentication with custom pages enabled:

```
parameter-map type webauth PMAP_WEBAUTH
 type webauth
 custom-page login device flash:webauth_login.html
 custom-page success device flash:webauth_success.html
 custom-page failure device flash:webauth_fail.html
 custom-page login expired device flash:webauth_expire.html
```

Related Commands

Command	Description
banner (parameter-map webauth)	Displays a banner on the web-authentication login web page.
consent email	Requests a user's e-mail address on the consent login web page.
redirect (parameter-map webauth)	Redirects clients to a particular URL during web-based authentication.

cws out

To enable Cloud Web Security content scanning on an egress interface, use the **cws out** command in interface configuration mode. To disable Cloud Web Security content scanning, use the **no** form of this command.

cws out
no cws out

Syntax Description

This command has no arguments or keywords.

Command Default

Cloud Web Security content scan is disabled.

Command Modes

Interface configuration (config-if)

Command History

Release	Modification
15.4(2)T	This command was introduced. This command replaces the **content-scan out** command.

Usage Guidelines

The content scanning process redirects client web traffic to Cloud Web Security. Content scanning is enabled on an Internet-facing WAN interface to protect the web traffic going out.

In case you enable content scanning on a interface that has Wide Area Application Services (WAAS) configured, you must not apply both the WAAS and content scanning feature on the same TCP session.

Examples

The following example shows how to enable Cloud Web Security content scanning on a Gigabit Ethernet interface:

```
Device(config)# interface gigabitethernet 0/0
Device(config-if)# cws out
```

Related Commands

Command	Description
cws whitelisting	Enables allowed listing of incoming traffic and enters Cloud Web Security allowed list configuration mode.
interface	Configures an interface and enters interface configuration mode.

cws whitelisting

To enable allowed listing of incoming traffic and to enter Cloud Web Security allowed listing configuration mode, use the **cws whitelisting** command in global configuration mode. To disable the allowed listing of traffic, use the **no** form of this command.

cws whitelisting
no cws whitelisting

Syntax Description

This command has no arguments or keywords.

Command Default

Allowed listing of traffic is disabled.

Command Modes

Global configuration (config)

Command History

Release	Modification
15.4(2)T	This command was introduced. This command replaces the **content-scan whitelisting** command.

Usage Guidelines

An approved list contains entities that are provided a particular privilege, service, mobility, access, or recognition. Allowed lists grant access.

The web traffic that you have configured for allowed listing will bypass the content scanning by Cloud Web Security.

Examples

The following example shows how to enable Cloud Web Security content scan allowed listing and enter Cloud Web Security allowed list configuration mode:

```
Device(config)# cws whitelisting
Device(config-cws-wl)#
```

Related Commands

Command	Description
parameter-map type cws global	Configures a global Cloud Web Security parameter map and enters parameter-map type inspect configuration mode.