Cisco IOS Certificate Server

The Cisco IOS Certificate Server feature embeds a simple certificate server, with limited certification authority (CA) functionality, into the Cisco IOS software. Thus, the following benefits are provided to the user:

- Easier public key infrastructure (PKI) deployment by defining default behavior. The user interface is simpler because default behaviors are predefined. That is, you can leverage the scaling advantages of PKI without all of the certificate extensions that a CA provides, thereby allowing you to easily enable a basic PKI-secured network.
- Direct integration with Cisco IOS software.

Feature History for Cisco IOS Certificate Server

Release	Modification
12.3(4)T	This feature was introduced.
12.3 (7)T	Certificate server Registration Authority (RA) mode was added.
12.3(11)T	The Certificate Server Auto Archive and Clear Certificate Server Enrollment Request Database enhancements were added.

Finding Support Information for Platforms and Cisco IOS Software Images

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

Contents

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

Prerequisites for Cisco IOS Certificate Server

- The certificate server supports Simple Certificate Enrollment Protocol (SCEP) over HTTP. The HTTP server must be enabled on the router for the certificate server to use SCEP. (To enable the HTTP server, use the **ip http server** command.) The certificate server will automatically enable or disable SCEP services after the HTTP server is enabled or disabled. If the HTTP server is not enabled, only manual PKCS10 enrollment is supported.
- Time services must be running on the router because the certificate server must have reliable time knowledge. If a hardware clock is unavailable, the certificate server will depend on manually configured clock settings, such as Network Time Protocol (NTP). If there is not a hardware clock or the clock is invalid, the following message will be displayed at bootup:

  ```
  % Time has not been set. Cannot start the Certificate server.
  ```

  After the clock has been set, the certificate server will automatically switch to running status. (See the **show crypto pki server** command output example in the section Certificate Server Enabled on a Router: Example.)

  For information on manually configuring clock settings, see the section “Setting Time and Calendar Services” in the chapter “Performing Basic System Management” of the *Cisco IOS Configuration Fundamentals and Network Management Configuration Guide*.

Restrictions for Cisco IOS Certificate Server

- The certificate server does not provide a mechanism for modifying the certificate request that is received from the client; that is, the certificate that is issued from the certificate server matches the requested certificate without modifications. If a specific certificate policy, such as name constraints, must be issued, the policy must be reflected in the certificate request.
- When the Cisco IOS certificate server is acting as an RA, the issuing CA should be a Cisco IOS certificate server.
- Only one certificate server in the hierarchy is supported.
- The certificate server automatically generates a 1024-bit Rivest, Shamir, and Adelman (RSA) key pair. You must manually generate an RSA key pair if you prefer a larger key pair. (For information on completing this task, see the section “Generating and Exporting a Certificate Server RSA Key Pair” later in this document.)
- Autoarchiving will not occur if you generate the CA key manually and mark it “nonexportable.”

Information About Cisco IOS Certificate Server

To configure a certificate server, you should understand the following concepts:

Key Pair, Certificate, and Trustpoint of the Certificate Server

The certificate server will use a regular Cisco IOS key pair as its CA key. This key pair must have the same name as the certificate server. If you do not generate the key pair before the certificate server is created on the router, a general-purpose key pair will be automatically generated during the configuration of the certificate server. If the key pair is automatically generated, it will not be marked as exportable. Thus, you must manually generate the key pair as exportable if you want to back up the CA key. (For more information, see the section "Generating and Exporting a Certificate Server RSA Key Pair.")

Note The Certificate Server Auto Archive enhancement was introduced in Cisco IOS Release 12.3(11)T. Effective with that enhancement, the CA certificate and CA key will be backed up automatically just once after they are are generated by the certificate server. As a result, it is not necessary to generate an exportable CA key if CA backup is desirable. For more information, see the section "Certificate Server Auto Archive."

The certificate server will also have an automatically generated trustpoint of the same name; the trustpoint will store the certificate of the certificate server. After the router detects that a trustpoint is being used to store the certificate of the certificate server, the trustpoint will be locked so that it cannot be modified. (Before configuring the certificate server, you can manually create and set up this trustpoint, which allows you to specify an alternative RSA key pair [using the **rsakeypair** command]).

If the server is a root certificate server, it will use the RSA key pairs and several other attributes to generate a self-signed certificate. The associated CA certificate will have the following key usage extensions—Digital Signature, Certificate Sign, and CRL (certificate revocation list) Sign.

Note After the CA certificate is generated, attributes can be changed only if the certificate server is destroyed.

Certificate Server Auto Archive

The Certificate Server Auto Archive enhancement allows you, at initial setup, to archive the CA certificate and the CA key so that they may later be restored if either the original copy or the original configuration is lost.

When the certificate server is turned on the first time, the CA certificate and CA key will be generated. If the Certificate Server Auto Archive enhancement is also enabled, they will be exported (archived) to the server database. The archive can be in PKCS12 or privacy-enhanced mail (PEM) format.

Note

- This archiving action occurs only one time. Only the CA key that is (1) manually generated and marked exportable or (2) automatically generated by the certificate server will be archived (this key will be marked nonexportable).
- This CA key backup file is extremely important and should be moved immediately to another secured place.
- In addition to the CA certificate and CA key archive file, you should also regularly back up the serial file (.ser) and the CRL file (.crl). The serial file and the CRL file are both critical for CA operation if you need to restore your certificate server.

If you later want to back up your certificate server, you may not be able to export the CA key that is necessary to perform the backup if it is not marked “exportable” (that is, if the key was generated automatically).

Certificate Revocation Lists

CRLs are issued once every specified time period via the **lifetime crl** command. Thereafter, the CRL is written to the specified database location as *ca-label*.crl (where *ca-label* is the name of the certificate server). 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. If the **cdp-url** command is not specified, the CRL distribution point (CDP) certificate extension will not be included in the certificates that are issued by the certificate server. Thus, Cisco IOS PKI clients will automatically use SCEP to retrieve a CRL from the certificate server, which puts an additional load on the certificate server because it must provide SCEP server support for each CRL request.

Note

- The CRL will always be available via SCEP, which is enabled by default, if the HTTP server is enabled.
- For large PKI deployments, it is recommended that you configure an HTTP-based CDP; for example, CDP URL http://myhttpserver.company.com/mycs.crl.

The CDP URL may be changed after the certificate server is running, but existing certificates will not be reissued with the new CDP that is specified via the **cdp-url** command.

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

The certificate server supports only one CDP; thus, all certificates that are issued include the same CDP.

Certificate Server States

At startup, the certificate server must check the current configuration before issuing any certificates. As it starts up, the certificate server transitions through the states defined in Table 1. To view the certificate server state, use the **show crypto pki server** command.

Table 1 Certificate Server Startup State Descriptions

Certificate Server State	Description
configured	The server is available and has generated the certificate server certificates.
storage configuration incomplete	The server is verifying that the configured storage location is available.
waiting for HTTP server	The server is verifying that the HTTP server is running.
waiting for time setting	The server is verifying that the time has been set.

The certificate server is enabled after it has successfully gone through each of the startup states. If the certificate server experiences a critical failure at any time, such as failing to publish a CRL, the certificate server will automatically enter a disabled state. This state allows the network administrator to fix the condition; thereafter, the certificate server will return to the previous normal state.

Certificate Enrollment

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 Table 2 for a complete list of certificate enrollment request states.)
  - The certificate server refers to the 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 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 the request to the 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, which 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 the client to request another certificate.

All enrollment requests transition through the certificate enrollment states that are defined in Table 2. To see current enrollment requests, use the **crypto pki server request pkcs10** command.

Table 2 Certificate Enrollment Request State Descriptions

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

SCEP Reenrollment

All SCEP requests are treated as new certificate enrollment requests, even if the request specifies a duplicate subject-name or public key pair as a previous certificate request. When servicing an enrollment request, there are no extended database look-ups.

Clear Certificate Server Enrollment Request Database

After the certificate server receives an enrollment request, the server can leave the request in a pending state, reject it, or grant it. The request stays in the Enrollment Request Database for 1 week until the client polls the certficate server for the result of the request. The Clear Certificate Server Enrollment Request Database enhancement allows you to remove either individual or all requests from the database, especially useful if the client exits and never polls the certificate server.

In addition, this enhancement, which uses the **crypto pki server remove** command, allows the server to be returned to a clean slate with respect to the keys and transaction IDs. Thus, the **crypto pki server remove** command is a useful command to use during troubleshooting with a SCEP client that may not be behaving properly.

RA Mode

RA is the authority charged with recording or verifying some or all of the data required for the CA to issue certificates. In many cases the CA will undertake all of the RA functions itself, but where a CA operates over a wide geographical area or when there is security concern over exposing the CA at the edge of the network, it may be administratively advisable to delegate some of the tasks to an RA and leave the CA to concentrate on its primary tasks of signing certificates and CRLs.

A Cisco IOS certificate server can be configured to run in RA mode. When the RA receives a SCEP or manual enrollment request, the adminstrator can either reject or grant it on the basis of local policy. If the request is granted, it will be forwarded to the issuing CA, and the CA will automatically generate the certificate and return it to the RA. The client can later retrieve the granted certificate from the RA.

Informational Messages During Bootup

If the certificate server is part of your start-up configuration, you may see the following messages during the boot procedure:

```
% Failed to find Certificate Server’s trustpoint at startup

% Failed to find Certificate Server’s cert.
```

The above are informational messages and indicate a temporary inability to configure the certificate server because the start-up configuration has not been fully parsed yet. The messages are useful for debugging in case the start-up configuration has been corrupted.

You can verify the status of the certificate server after the boot procedure using the **show crypto pki server** command.

How to Configure Certificate Server Functionality

This section contains the following procedures:

Generating and Exporting a Certificate Server RSA Key Pair

Use this task to manually generate an RSA key pair as exportable for the certificate server. If this task is not performed, the certificate sever will automatically generate a key pair, which will not be marked as exportable.

Note

- The Certificate Server Auto Archive enhancement was introduced in Cisco IOS Release 12.3(11)T. As a result, this task (“Generating and Exporting a Certificate Server RSA Key Pair”) is no longer necessary if you keep the archive file in a safe place.
- In addition to keeping the private key in a secure location, it is recommended that you regularly archive the certificate server database.

SUMMARY STEPS

1. **enable**

2. **configure terminal**
3. **crypto key generate rsa general-keys label** *key-label* **exportable**
4. **crypto key export rsa** *key-label* **pem** {**terminal** | **url** *url*} {**3des** | **des**} *passphrase*
5. **crypto key import rsa** *key-label* **pem** [**usage-keys**] {t**erminal** | **url** *url*} [**exportable**] *passphrase*
6. **exit**
7. **show crypto key mypubkey rsa**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Router> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Router# configure terminal`	Enters global configuration mode.
Step 3	**crypto key generate rsa general-keys label** *key-label* **exportable** **Example:** `Router (config)# crypto key generate rsa general-keys label mycs exportable`	Generates the RSA key pair for the certificate server. **Note** You must use the same name for the key pair (*key-label*) that you plan to use for the certificate server (via the **crypto pki server** *cs-label* command). **Note** If you manually generate the exportable RSA key pair but wait until after the certificate server has been generated before issuing the **no shutdown** command, you can use the **crypto ca export pkcs12** command to export a PKCS12 file that contains the certificate server certificate as well as the private key.
Step 4	**crypto key export rsa** *key-label* **pem** {**terminal**	**url** *url*} {**3des**	**des**} *passphrase* **Example:** `Router (config)# crypto key export rsa mycs pem url nvram: 3des PASSWORD`	Exports the generated RSA key pair.
Step 5	**crypto key import rsa** *key-label* **pem** [**usage-keys**] {**terminal**	**url** *url*} [**exportable**] *passphrase* **Example:** `Router (config)# crypto key import rsa mycs2 pem url nvram: mycs PASSWORD`	(Optional) Imports the RSA key pair. **Note** If you do not want the key to be exportable from your certificate server, import it back to the certificate server after it has been exported as a nonexportable key pair. Thus, the key cannot be taken off again.

	Command or Action	Purpose
Step 6	**exit** **Example:** `Router (config)# exit`	Exits global configuration.
Step 7	**show crypto key mypubkey rsa** **Example:** `Router# show crypto key mypubkey rsa`	Displays the RSA public keys of your router.

For more information on exporting and importing RSA key pairs, see the feature *Import of RSA Key Pair and Certificates in PEM Format*, Cisco IOS Release 12.3(4)T.

Examples

The following example shows how to generate, export, bring the key back (import), and verify the status of the RSA key pair:

```
! Generate the key pair
!
Router(config)# crypto key generate rsa general-keys label mycs exportable
The name for the keys will be: mycs
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]: 1024
% Generating 1024 bit RSA keys ...[OK]
!
! Archive the key pair to a remote location, and use a good password.
!
Router(config)# crypto key export rsa mycs pem url nvram: 3des PASSWORD
% Key name: mycs
Usage: General Purpose Key
Exporting public key...
Destination filename [mycs.pub]?
Writing file to nvram:mycs.pub
Exporting private key...
Destination filename [mycs.prv]?
Writing file to nvram:mycs.prv
!
! Import the key as a different name.
!
Router(config)# crypto key import rsa mycs2 pem url nvram:mycs PASSWORD
% Importing public key or certificate PEM file...
Source filename [mycs.pub]?
Reading file from nvram:mycs.pub
% Importing private key PEM file...
Source filename [mycs.prv]?
Reading file from nvram:mycs.prv% Key pair import succeeded.
!
! After the key has been imported, it is no longer exportable.
!
! Verify the status of the key.
!
Router# show crypto key mypubkey rsa

% Key pair was generated at: 18:04:56 GMT Jun 6 2003
Key name: mycs
```

```
Usage: General Purpose Key
Key is exportable.
Key Data:
  30819F30 0D06092A 864886F7 0D010101 05000381 8D003081 89028181 00E65253
  9C30C12E 295AB73F B1DF9FAD 86F88192 7D4FA4D2 8BA7FB49 9045BAB9 373A31CB
  A6B1B8F4 329F2E7E 8A50997E AADBCFAA 23C29E19 C45F4F05 DBB2FA51 4B7E9F79
  A1095115 759D6BC3 5DFB5D7F BCF655BF 6317DB12 A8287795 7D8DC6A3 D31B2486
  C9C96D2C 2F70B50D 3B4CDDAE F661041A 445AE11D 002EEF08 F2A627A0 5B020301 0001
% Key pair was generated at: 18:17:25 GMT Jun 6 2003
Key name: mycs2
Usage: General Purpose Key
Key is not exportable.
Key Data:
  30819F30 0D06092A 864886F7 0D010101 05000381 8D003081 89028181 00E65253
  9C30C12E 295AB73F B1DF9FAD 86F88192 7D4FA4D2 8BA7FB49 9045BAB9 373A31CB
  A6B1B8F4 329F2E7E 8A50997E AADBCFAA 23C29E19 C45F4F05 DBB2FA51 4B7E9F79
  A1095115 759D6BC3 5DFB5D7F BCF655BF 6317DB12 A8287795 7D8DC6A3 D31B2486
  C9C96D2C 2F70B50D 3B4CDDAE F661041A 445AE11D 002EEF08 F2A627A0 5B020301 0001
! After the key has been imported, it is no longer exportable.
!
! Verify the status of the key.
!
Router# show crypto key mypubkey rsa

% Key pair was generated at: 18:04:56 GMT Jun 6 2003
Key name: mycs
Usage: General Purpose Key
Key is exportable.
Key Data:
30819F30 0D06092A 864886F7 0D010101 05000381 8D003081 89028181 00E65253

% Key pair was generated at: 18:17:25 GMT Jun 6 2003
Key name: mycs2
Usage: General Purpose Key
Key is not exportable.
Key Data:
30819F30 0D06092A 864886F7 0D010101 05000381 8D003081 89028181 00E65253

! Keep the PEM files in a safe place.
!
! Now you can delete the exportable key "mycs" and configure a certificate server "mycs2"
! to use the non-exportable key.
!
Router# configure terminal
Router(config)# crypto key zeroize rsa mycs
```

What to Do Next

After you have manually generated an RSA key pair, you can enable the certificate server using the manually generated key pair. You will configure the certificate server as normal, except that the certificate server will not generate a new key pair because you have already manually created one. Also, the certificate server must use the same name as the key pair you just manually generated.

To enable the certificate server, see the following section, “Enabling a Certificate Server.”

Enabling a Certificate Server

Use this task to configure a Cisco IOS certificate server.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **ip http server**
4. **crypto pki server** *cs-label*

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Router> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Router# configure terminal`	Enters global configuration mode.
Step 3	**ip http server** **Example:** `Router (config)# ip http server`	Enables the HTTP server on your system.
Step 4	**crypto pki server** *cs-label* **Example:** `Router (config)# crypto pki server server-pki`	Enables the certificate server and enters certificate server configuration mode. **Note** If you manually generated an RSA key pair, the *cs-label* argument must match the name of the key pair. **Note** The *cs-label* should not exceed 13 characters.

What to Do Next

After you have enabled a certificate server, you can use the preconfigured default values or specify values via the CLI for the functionality of the certificate server. If you choose to specify values other than the defaults, see the following section, "Configuring Certificate Server Functionality."

Configuring Certificate Server Functionality

Use this task to configure basic certificate server functionality values other than the default values.

SUMMARY STEPS

1. **database url** *root-url*
2. **database level** {**minimal** | **names** | **complete**}
3. **database username** *username* [**password** *password*]
4. **database archive** {**pkcs12** | **pem**} [**password** *password*]
5. **issuer-name** *DN-string*
6. **lifetime** {**ca-certificate** | **certificate**} *time*
7. **lifetime crl** *time*
8. **lifetime enrollment-request** *time*
9. **cdp-url** *url*
10. **no shutdown**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**database url** *root-url* **Example:** `Router (cs-server)# database url tftp://mytftp`	Specifies the location where all database entries for the certificate server will be written out. If this command is not specified, all database entries will be written to NVRAM.
Step 2	**database level** {**minimal**	**names**	**complete**} **Example:** `Router (cs-server)# database level complete`	Controls what type of data is stored in the certificate enrollment database. • **minimal**—Enough information is stored only to continue issuing new certificates without conflict; the default value. • **names**—In addition to the information given in the minimal level, the serial number and subject name of each certificate. • **complete**—In addition to the information given in the minimal and names levels, each issued certificate is written to the database. **Note** The **complete** keyword produces a large amount of information; if it is issued, you should also specify an external TFTP server in which to store the data via the **database url** command.
Step 3	**database username** *username* [**password** *password*] **Example:** `Router (cs-server)# database username user1 password cisco123`	Requires a username or password to be issued when accessing a certificate enrollment database storage location. • If the password is configured, it will be encrypted.

	Command or Action	Purpose
Step 4	**database archive** {**pkcs12** | **pem**} [**password** *password*] **Example:** `Router (cs-server)# database archive pem password cisco123`	(Optional) Sets the CA key and CA certificate archive format and password to encrypt the file. • The default is database archive pkcs12, so if this subcommand is not configured, autoarchiving will still be done, and the PKCS12 format will be used. • The password is optional. If it is not configured, you will be prompted for the password when the server is turned on for the first time. **Note** It is recommended that you remove the password from the configuration after the archive is finished.
Step 5	**issuer-name** *DN-string* **Example:** `Router (cs-server)# issuer-name CN = ipsec_cs,L = Santa Cruz,C = US`	Sets the CA issuer name to the specified distinguished name (DN-string). The default value is as follows: **issuer-name cn**={*cs-label*}.
Step 6	**lifetime** {**ca-certificate** | **certificate**} *time* **Example:** `Router (cs-server)# lifetime ca-certificate 30`	(Optional) Specifies the lifetime, in days, of a CA certificate or a certificate. Valid values range from 1 day to 1825 days. The default CA certificate lifetime is 3 years; the default certificate lifetime is 1 year. The maximum certificate lifetime is 1 month less than the lifetime of the CA certificate.
Step 7	**lifetime crl** *time* **Example:** `Router (cs-server)# lifetime crl 24`	(Optional) Defines the lifetime, in hours, of the CRL that is used by the certificate server. Maximum lifetime value is 336 hours (2 weeks). The default value is 168 hours (1 week).
Step 8	**lifetime enrollment-request** *time* **Example:** `Router (cs-server)# lifetime enrollment-request 24`	(Optional) Specifies how long an enrollment request should stay in the enrollment database before being removed.
Step 9	**cdp-url** *url* **Example:** `Router (cs-server)# cdp-url http://myhttpserver/mycdp.mycs.crl`	Defines a CDP to be used in the certificates that are issued by the certificate server. URL must be an HTTP URL.
Step 10	**no shutdown** **Example:** `Router (cs-server)# no shutdown`	Disables a certificate server without removing the configuration. You should issue this command only after you have completely configured your certificate server.

Specifying Enrollment Processing Parameters

SCEP, which is the only supported enrollment protocol, supports two client authentication mechanisms—manual and preshared key. Manual enrollment requires the administrator at the 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. Use any of these optional steps to help specify enrollment processing parameters that are to be used by SCEP.

SUMMARY STEPS

1. **enable**
2. **crypto pki server** *cs-label* **grant** {**all** | *req-id*}
3. **crypto pki server** *cs-label* **reject** {**all** | *req-id*}
4. **crypto pki server** *cs-label* **password generate** [*minutes*]
5. **crypto pki server** *cs-label* **revoke** *certificate-serial-number*
6. **crypto pki server** *cs-label* **request pkcs10** {*url* | **terminal**} [**pem**]
7. **crypto pki server** *cs-label* **info crl**
8. **crypto pki server** *cs-label* **info requests**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Router> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**crypto pki server** *cs-label* **grant** {**all** | *req-id*} **Example:** `Router# crypto pki server mycs grant all`	Grants all or specific SCEP requests.
Step 3	**crypto pki server** *cs-label* **reject** {**all** | *req-id*} `Router# crypto pki server mycs reject all`	Rejects all or specific SCEP requests.
Step 4	**crypto pki server** *cs-label* **password generate** [*minutes*] **Example:** `Router# crypto pki server mycs password generate 75`	Generates a one-time password (OTP) for SCEP requests. • *minutes*—Length of time, in minutes, that the password is valid. Valid values range from 1 to 1440 minutes. The default is 60 minutes. **Note** Only one OTP is valid at a time; if a second OTP is generated, the previous OTP is not valid.
Step 5	**crypto pki server** *cs-label* **revoke** *certificate-serial-number* **Example:** `Router# crypto pki server mycs revoke 3`	Revokes a certificate on the basis of its serial number. • The serial number can be a hexadecimal number with the prefix “0x” (for example, 0x4c) or a decimal number (for example, 76).

	Command or Action	Purpose
Step 6	**crypto pki server** *cs-label* **request pkcs10** {*url* | **terminal**} [**pem**] **Example:** `Router# crypto pki server mycs request pkcs10 terminal pem`	Manually adds either a base64-encoded or PEM-formatted PKCS10 certificate enrollment request to the request database. After the certificate is granted, it will be displayed on the console terminal using base64 encoding; if the **pem** keyword is specified, PEM headers are also added to the certificate.
Step 7	**crypto pki server** *cs-label* **info crl** **Example:** `Router# crypto pki server mycs info crl`	Displays information regarding the status of the current CRL.
Step 8	**crypto pki server** *cs-label* **info requests** **Example:** `Router# crypto pki server mycs info requests`	Displays all outstanding certificate enrollment requests.

Removing Requests from the Enrollment Request Database

To remove requests from the Enrollment Request Database, perform the following steps.

SUMMARY STEPS

1. **enable**
2. **crypto pki server** *cs-label* **remove** {**all** | *req-id*}

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Router> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**crypto pki server** *cs-label* **remove** {**all** | *req-id*} **Example:** `Router# crypto pki server mycs remove 15`	Removes enrollment requests from the Enrollment Request Database.

Verifying and Troubleshooting Certificate Server, Certificate, and CA Status

Use any of the following optional steps to verify the status of the certificate server, the certificate, or the CA.

SUMMARY STEPS

1. **enable**
2. **show crypto pki server**
3. **show crypto ca certificates**
4. **debug crypto pki server**
5. **dir** *filesystem***:**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Router> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**show crypto pki server** **Example:** `Router# show crypto pki server`	Displays the current state and configuration of the certificate server.
Step 3	**show crypto ca certificates** **Example:** `Router# show crypto ca certificates`	Displays information about your certificate, the CA certificate, and any registration authority certificates.
Step 4	**debug crypto pki server** **Example:** `Router# debug crypto pki server`	Enables debugging for a crypto PKI certificate server. • This command can be used for monitoring the progress of an enrollment and for troubleshooting if the certificate server fails to respond or if the certificate server has trouble handling the request that has been configured.
Step 5	**dir** *filesystem*: **Example:** `Router# dir slot0:`	Displays a list of files on a file system. • This command can be used to verify the certificate server autoarchived file if the **database url** command was entered to point to a local file system. You should be able to at least see "*cs-label*.ser" and "*cs-label*.crl" files in the database.

Examples

The following sample output for the **show crypto pki server** command shows the status of the certificate server as well as server parameters:

```
Router# show crypto pki server

    Certificate Server status:enabled, configured
    Granting mode is:manual
    Last certificate issued serial number:0x1
    CA certificate expiration timer:19:31:15 PST Nov 17 2006
    CRL NextUpdate timer:19:31:15 PST Nov 25 2003
```

```
    Current storage dir:nvram:
    Database Level:Minimum - no cert data written to storage
```

The following sample output for the **debug crypto pki server** command is typical of output that could be used to troubleshoot or check the progress of a certificate enrollment:

```
! Received client "crypto ca authenticate" request.
Aug 23 21:25:12.893: CRYPTO_CS: received a SCEP GetCACert request
Aug 23 21:25:12.897: CRYPTO_CS: CA certificate sent

Aug 23 21:25:25.449: CRYPTO_CS: enter FSM: input state enabled, input signal time set
Aug 23 21:25:25.449: CRYPTO_CS: exit FSM: new state enabled

! Received client "crypto ca enroll" request.
Aug 23 21:25:35.585: CRYPTO_CS: received a SCEP request
Aug 23 21:25:35.589: CRYPTO_CS: read SCEP: registered and bound service SCEP_READ_DB_1
Aug 23 21:25:35.653: CRYPTO_CS: scep msg type - 19
Aug 23 21:25:35.653: CRYPTO_CS: trans id - 080D7BFB739AD103B9C99F1A9BCFD2B7
Aug 23 21:25:36.325: CRYPTO_CS: read SCEP: unregistered and unbound service SCEP_READ_DB_1
Aug 23 21:25:36.325: CRYPTO_CS: received an enrollment request
! Pending request ID is 1.
Aug 23 21:25:36.329: CRYPTO_CS: reqID = 1
Aug 23 21:25:36.333: CRYPTO_CS: write SCEP: registered and bound service SCEP_WRTE_DB_1
Aug 23 21:25:37.045: CRYPTO_CS: write SCEP: unregistered and unbound service
SCEP_WRTE_DB_1
Aug 23 21:25:37.049: CRYPTO_CS: sent SCEP pending reply
Aug 23 21:25:38.589: CRYPTO_CS: received a SCEP request
Aug 23 21:25:38.593: CRYPTO_CS: read SCEP: registered and bound service SCEP_READ_DB_2
Aug 23 21:25:38.653: CRYPTO_CS: scep msg type - 20
Aug 23 21:25:38.653: CRYPTO_CS: trans id - 080D7BFB739AD103B9C99F1A9BCFD2B7
Aug 23 21:25:39.325: CRYPTO_CS: read SCEP: unregistered and unbound service SCEP_READ_DB_2
Aug 23 21:25:39.325: CRYPTO_CS: received an enrollment request
Aug 23 21:25:39.325: CRYPTO_CS: reqID = 1
Aug 23 21:25:39.329: CRYPTO_CS: write SCEP: registered and bound service SCEP_WRTE_DB_2
Aug 23 21:25:40.037: CRYPTO_CS: write SCEP: unregistered and unbound service
SCEP_WRTE_DB_2
Aug 23 21:25:40.041: CRYPTO_CS: sent SCEP pending reply
Router#

! Manually grant the request.
Router# crypto pki server sub grant 1

Aug 23 21:25:53.833: CRYPTO_CS: Granting enrollment request 1
Aug 23 21:25:53.837: CRYPTO_CS: added key usage extension
Aug 23 21:25:54.533: CRYPTO_CS: serial number 0x2 written.
Aug 23 21:25:56.725: CRYPTO_CS: reqID=1 granted,
fingerprint=919929D8C9B89607AB8FF53876F8E770
Aug 23 21:26:49.761: CRYPTO_CS: received a SCEP request
Aug 23 21:26:49.765: CRYPTO_CS: read SCEP: registered and bound service SCEP_READ_DB_3
Aug 23 21:26:49.825: CRYPTO_CS: scep msg type - 20
Aug 23 21:26:49.829: CRYPTO_CS: trans id - 080D7BFB739AD103B9C99F1A9BCFD2B7
Aug 23 21:26:50.497: CRYPTO_CS: read SCEP: unregistered and unbound service SCEP_READ_DB_3

! Received client polling.
Aug 23 21:26:50.497: CRYPTO_CS: received an enrollment request
Aug 23 21:26:50.501: CRYPTO_CS: write SCEP: registered and bound service SCEP_WRTE_DB_3
Aug 23 21:26:51.293: CRYPTO_CS: write SCEP: unregistered and unbound service
SCEP_WRTE_DB_3

! The granted certificate is returned to the client.
Aug 23 21:26:51.305: CRYPTO_CS: Certificate sent to requestor.
```

Restoring a Certificate Server from Certificate Server Backup Files

If your certificate server configuration somehow gets corrupted, you can restore the certificate server using the backup files. You will need the serial file (.ser), CRL file (.crl), and the CA certificate and CA key archive (.p12 or .pem). You will also need the encryption password for the archive file. If any of those files have been lost or you forget the password, you will not be able to recover the certificate server configuration. (For output examples in which certificate servers have been restored using backup files, see the section "Restoring a Certificate Server from Certificate Server Backup Files: Examples.")

Configuring a Certificate Server to Run in RA Mode

After a certificate server is running as a CA, you may want to configure an RA mode certificate server on another device and delegate the task of enrollment request handling to that device. To configure a certificate server to run in RA mode, perform the following steps.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **crypto pki trustpoint** *name*
4. **enrollment url** *url*
5. **subject-name** *x.500-name*
6. **exit**
7. **crypto pki server** *cs-label*
8. **mode ra**
9. **no shutdown**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Router> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Router# configure terminal`	Enters global configuration mode.
Step 3	**crypto pki trustpoint** *name* **Example:** `Router (config)# crypto pki trustpoint myra`	Declares the trustpoint that your RA mode certificate server should use and enters ca-trustpoint configuration mode.

	Command or Action	Purpose
Step 4	**enrollment url** *url* **Example:** `Router (ca-trustpoint)# enrollment url http://10.3.0.6`	Specifies the enrollment URL of the issuing CA certificate server.
Step 5	**subject-name** *x.500-name* **Example:** `Router (ca-trustpoint)# subject-name cn=myra, ou=ioscs RA, o=cisco, c=us`	Specifies the subject name the RA will use. **Note** Include "cn=ioscs RA" or "ou=ioscs RA" in the subject name so that the issuing CA certificate server can recognize the RA (see step 7 below).
Step 6	**exit** **Example:** `Router (ca-trustpoint)# exit`	Exits ca-trustpoint configuration mode.
Step 7	**crypto pki server** *cs-label* **Example:** `Router(config)# crypto pki server myra`	Enables a Cisco IOS certificate server and enters cs-server configuration mode. • *cs-label*—Should not exceed 13 characters. **Note** The certificate server must have the same name as the trustpoint that was created in Step 3 above.
Step 8	**mode ra** **Example:** `Router(cs-server)# mode ra`	Places the PKI server into RA certificate server mode.
Step 9	**no shutdown** **Example:** `Router(cs-server)# no shutdown`	Reenables the certificate server.

The following steps should be configured on the router that is running the issuing certificate server.

SUMMARY STEPS

1. **enable**
2. **crypto pki server** *cs-label* **info requests**
3. **crypto pki server** *cs-label* **grant** *req-id*
4. **configure terminal**
5. **crypto pki server** *cs-label*
6. **grant ra-auto**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Router# enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**crypto pki server** *cs-label* **info requests** **Example:** `Router# crypto pki server mycs info requests`	Displays the outstanding RA certificate request. **Note** This command is configured on the router that is running the issuing certificate server. **Note** The RA certificate request can only be identified if it has "cn=ioscs RA" or "ou=ioscs RA" in the subject name.
Step 3	**crypto pki server** *cs-label* **grant** *req-id* **Example:** `Router# crypto pki server myc grant 12`	Grants the pending RA certificate request. **Note** Because the issuing certificate server will delegate the enrollment request verification task to the RA, you must pay extra attention to the RA certificate request before granting it.
Step 4	**configure terminal** **Example:** `Router# configure terminal`	Enters global configuration mode.
Step 5	**crypto pki server** *cs-label* **Example:** `Router (config)# crypto pki server mycs`	Enables a Cisco IOS certificate server and enters cs-server configuration mode. *cs-label*—Should not exceed 13 characters.
Step 6	**grant ra-auto** **Example:** `Router(cs-server)# grant ra-auto`	Specifies that all enrollment requests from an RA are to be granted automatically. **Note** For the **grant ra-auto** command to work, you have to include “cn=ioscs RA” or “ou=iosc RA” in the subject name of the RA certificate. (See Step 5 above.)

Deleting a Certificate Server

To delete a certificate server, perform the following steps.

Note When a certificate server is deleted, the trustpoint and key are automatically deleted.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **no crypto pki server** *cs-label*

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Router# enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Router# configure terminal`	Enters global configuration mode.
Step 3	**no crypto pki server** *cs-label* **Example:** `Router (config)# no crypto pki server mycs`	Deletes a certificate server.

Configuration Examples for Enabling a Certificate Server

This section provides the following examples:

- Certificate Server Enabled on a Router: Example, page 21
- Basic Certificate Server Configuration: Example, page 21
- Removing Enrollment Requests from the Enrollment Request Database: Examples, page 22
- Autoarchiving the Certificate Server Root Keys: Examples, page 23
- Restoring a Certificate Server from Certificate Server Backup Files: Examples, page 25
- RA Mode Certificate Server: Examples, page 27

Certificate Server Enabled on a Router: Example

After a certificate server has been enabled on a router, the **show crypto pki server** command displays the following output:

```
Router# show crypto pki server

Certificate Server status:enabled, configured
Granting mode is:manual
Last certificate issued serial number:0x1
CA certificate expiration timer:19:31:15 PST Nov 17 2006
CRL NextUpdate timer:19:31:15 PST Nov 25 2003
Current storage dir:nvram:
Database Level:Minimum - no cert data written to storage
```

Basic Certificate Server Configuration: Example

The following example shows how to configure the certificate server “ca”:

```
Router(config)# crypto pki server ca
Router(cs-server)# no shutdown
```

```
% Once you start the server, you can no longer change some of
% the configuration.
Are you sure you want to do this? [yes/no]: yes
% Generating 1024 bit RSA keys ...[OK]

% Certificate Server enabled.
Router(cs-server)# end
!
Router# show crypto pki server

Certificate Server ca:
    Status: enabled, configured
    CA cert fingerprint: 5A856122 4051347F 55E8C246 866D0AC3
    Granting mode is: manual
    Last certificate issued serial number: 0x1
    CA certificate expiration timer: 19:44:57 GMT Oct 14 2006
    CRL NextUpdate timer: 19:45:25 GMT Oct 22 2003
    Current storage dir: nvram:
    Database Level: Complete - all issued certs written as <serialnum>.cer
```

Removing Enrollment Requests from the Enrollment Request Database: Examples

The following examples show the enrollment requests that are currently in the Enrollment Request Database and the result after one of the enrollment requests has been removed from the database.

Enrollment Request Currently in the Enrollment Request Database

The following example shows that the **crypto pki server info requests** command has been used to display the enrollment requests that are currently in the Enrollment Request Database:

```
Router# crypto pki server myserver info requests

Enrollment Request Database:

RA certificate requests:
ReqID    State      Fingerprint                              SubjectName
--------------------------------------------------------------------------

Router certificates requests:
ReqID    State      Fingerprint                              SubjectName
--------------------------------------------------------------------------
2        pending    1B07F3021DAAB0F19F35DA25D01D8567      hostname=host1.cisco.com
1        denied     5322459D2DC70B3F8EF3D03A795CF636      hostname=host2.cisco.com
```

crypto pki server remove Command Used to Remove One Enrollment Request

The following example shows that the **crypto pki server remove** command has been used to remove Enrollment Request 1:

```
Router# crypto pki server myserver remove 1
```

Enrollment Request Database After the Removal of One Enrollment Request

The following example shows the result of the removal of Enrollment Request 1 from the Enrollment Request Database:

```
Router# crypto pki server mycs info requests

Enrollment Request Database:
```

```
RA certificate requests:
ReqID    State     Fingerprint                         SubjectName
--------------------------------------------------------------------

Router certificates requests:
ReqID    State     Fingerprint                         SubjectName
--------------------------------------------------------------------
2        pending   1B07F3021DAAB0F19F35DA25D01D8567   hostname=host1.cisco.com
```

Autoarchiving the Certificate Server Root Keys: Examples

The following output configurations and examples show what you might see if the **database archive** command has not been configured (that is, using the default value); if the **database archive** command has been configured to set the CA certificate and CA key archive format as PEM, without configuring a password; and if the **database archive** command has been configured to set the CA certificate and CA key archive format as PKCS12, with a password configured. The last example is sample content of a PEM-formatted archive file.

database archive Command Not Configured

Note The default is PKCS12, and the prompt for the password is after "no shutdown."

```
Router (config)# crypto pki server myserver
Router (cs-server)# no shutdown

% Generating 1024 bit RSA keys ...[OK]
% Ready to generate the CA certificate.
%Some server settings cannot be changed after CA certificate generation.

Are you sure you want to do this? [yes/no]: y
% Exporting Certificate Server signing certificate and keys...
! Note the next two lines, which are asking for a password.
% Please enter a passphrase to protect the private key.
Password:
% Certificate Server enabled.
Router (cs-server)# end

Router# dir nvram:

Directory of nvram:/

  125  -rw-        1693              <no date>  startup-config
  126  ----           5              <no date>  private-config
    1  -rw-          32              <no date>  myserver.ser
    2  -rw-         214              <no date>  myserver.crl
! Note the next line, which indicates PKCS12 format.
    3  -rw-        1499              <no date>  myserver.p12
```

database archive Command and pem Keyword Configured

Note The prompt for the password is after "no shutdown."

```
Router (config)# crypto pki server myserver
Router (cs-server)# database archive pem
Router (cs-server)# no shutdown
```

```
% Generating 1024 bit RSA keys ...[OK]
% Ready to generate the CA certificate.
%Some server settings cannot be changed after CA certificate generation.

Are you sure you want to do this? [yes/no]: y
% Exporting Certificate Server signing certificate and keys...
!Note the next two lines, which are asking for a password.
% Please enter a passphrase to protect the private key.
Password:
% Certificate Server enabled.
Router (cs-server)# end

Router# dir nvram

Directory of nvram:/

  125  -rw-        1693            <no date>  startup-config
  126  ----           5            <no date>  private-config
    1  -rw-          32            <no date>  myserver.ser
    2  -rw-         214            <no date>  myserver.crl
! Note the next line showing that the format is PEM.
    3  -rw-        1705            <no date>  myserver.pem
```

database archive Command and pkcs12 Keyword (and Password) Configured

Note When the password is entered, it will be encrypted. However, it is recommended that you remove the password from the configuration after the archive has finished.

```
Router (config)# crypto pki server myserver
Router (cs-server)# database archive pkcs12 password cisco123
Router (cs-server)# no shutdown

% Generating 1024 bit RSA keys ...[OK]
% Ready to generate the CA certificate.
% Some server settings cannot be changed after CA certificate generation.

Are you sure you want to do this? [yes/no]: y
% Exporting Certificate Server signing certificate and keys...
! Note that you are not being prompted for a password.
% Certificate Server enabled.
Router (cs-server)# end
Router# dir nvram:
Directory of nvram:/

   125   -rw-         1693             <no date>   startup-config
   126   ----            5             <no date>   private-config
     1   -rw-           32             <no date>   myserver.ser
     2   -rw-          214             <no date>   myserver.crl
! Note that the next line indicates that the format is PKCS12.
     3   -rw-         1499             <no date>   myserver.p12
```

PEM-Formatted Archive

The following sample output shows that autoarchiving has been configured in PEM file format. The archive consists of the CA certificate and the CA private key. To restore the certificate server using the backup, you would have to import the PEM-formatted CA certificate and CA key individually. For more information about RSA key pairs and certificates in PEM format, see the document *Import of RSA Key Pair and Certificates in PEM Format.* (See the section "Related Documents.")

Note In addition to the CA certificate and CA key archive file, you should also back up the serial file (.ser) and the CRL file (.crl) regularly. The serial file and the CRL file are both critical for CA operation if you need to restore your certificate server.

```
Router# more nvram:mycs.pem

-----BEGIN CERTIFICATE-----
MIIB9zCCAWCgAwIBAgIBATANBgkqhkiG9w0BAQQFADAPMQ0wCwYDVQQDEwRteWNz
MB4XDTA0MDgyNzAyMzI0NloXDTA3MDgyNzAyMzI0NlowDzENMAsGA1UEAxMEbXlj
czCBnzANBgkqhkiG9w0BAQEFAAOBjQAwgYkCgYEA1lZpKP4nGDJHgPkpYSkix7lD
nr23aMlZ9Kz5oo/qTBxeZ8mujpjYcZ0T8AZvoOiCuDnYMl796ZwpkMgjz1aZZbL+
BtuVvllsEOfhC+u/Ol/vxfGG5xpshoz/F5J3xdg5ZZuWWuIDAUYu9+QbI5feuG04
Z/BiPIb4AmGTP4B2MM0CAwEAAaNjMGEwDwYDVR0TAQH/BAUwAwEB/zAOBgNVHQ8B
Af8EBAMCAYYwHwYDVR0jBBgwFoAUKi/cuK6wkz+ZswVtb06vUJboEeEwHQYDVR0O
BBYEFCov3LiusJM/mbMFbW9Or1CW6BHhMA0GCSqGSIb3DQEBBAUAA4GBAKLOmoE2
4+NeOKEXMCXG1jcohK7O2HrkFfl/vpK0+q92PTnMUFhxLOqI8pWIq5CCgC7heace
OrTv2zcUAoH4rzx3Rc2USIxkDokWWQMLujsMm/SLIeHit0G5uj//GCcbgK20MAW6
ymf7+TmblSFljWzstoUXC2hLnsJIMq/KffaD
-----END CERTIFICATE-----

!The private key is protected by the password that is
configured in “database archive pem password pwd” or that
is entered when you are prompted for the password.
-----BEGIN RSA PRIVATE KEY-----
Proc-Type: 4,ENCRYPTED
DEK-Info: DES-EDE3-CBC,106CE91FFD0A075E

zyiFC8rKv8Cs+IKsQG2QpsVpvDBHqZqBSM4D528bvZv7jzr6WuHj8E6zO+6G8R/A
zjsfTALo+e+ZDg7KMzbryHARvjskbqFdOMLlVIYBhCeSElKsskWB6chOuyPHJInW
JwC5YzZdZwOqcyLBP/xOYXcvjzzNfPAXZzN12VR8vWDNq/kHT+3Lplc8hY++ABMI
M+C9FB3dpNZzu5O1BZCJg46bqbkulaCCmScIDaVt0zDFZwWTSufiemmNxZBG4xS8
t5t+FEhmSfv8DAmwg4f/KVRFTm10phUArcLxQO38Al0W5YHHORdACnuzVUvHgco7
VT4XUTjO7qMhmJgFNWy1pu49fbdS2NnOn5IoiyAq5lk1KUPrz/WABWiCvLMylGnZ
kyMCWoaMtgS/vdx74BBCj09yRZJnLMlIi6SDofjCNTDHfmFEVg4LsSWCd4lP9OP8
0MqhP1D5VIx6PbMNwkWW12lpBbCCdesFRGHjZD2dOu96kHD7ItErx34CC8W04aG4
b7DLktUu6WNV6M8g3CAqJiC0V8ATlp+kvdHZVkXovgND5IU0OJpsj0HhGzKAGpOY
KTGTUekUboISjVVkI6efp1vO6temVL3Txg3KGhzWMJGrq1snghE0KnV8tkddv/9N
d/t1l+we9mrccTq50WNDnkEi/cwHI/0PKXg+NDNH3k3QGpAprsqGQmMPdqc5ut0P
86i4cF9078QwWg4Tpay3uqNH1Zz6UN0tcarVVNmDupFESUxYw10qJrrEYVRadu74
rKAU4Ey4xkAftB2kuqvr21Av/L+jne4kkGIoZYdB+p/M98pQRgkYyg==
-----END RSA PRIVATE KEY-----
```

Restoring a Certificate Server from Certificate Server Backup Files: Examples

The following example shows that restoration is from a PKCS12 archive and that the database URL is NVRAM (the default).

```
Router# copy tftp://172.71.71.71/backup.ser nvram:mycs.ser
Destination filename [mycs.ser]?

32 bytes copied in 1.320 secs (24 bytes/sec)
Router# copy tftp://172.71.71.71/backup.crl nvram:mycs.crl
Destination filename [mycs.crl]?

214 bytes copied in 1.324 secs (162 bytes/sec)
Router# configure terminal
Router (config)# crypto pki import mycs pkcs12 tftp://172.71.71.71/backup.p12 cisco123
Source filename [backup.p12]?
CRYPTO_PKI: Imported PKCS12 file successfully.
```

```
Router (config)# crypto pki server mycs
! fill in any CS configuration here
Router (cs-server)# no shutdown
% Certificate Server enabled.
Router (cs-server)# end
Router# show crypto pki server
Certificate Server mycs:
    Status: enabled
    Server's current state: enabled
    Issuer name: CN=mycs
    CA cert fingerprint: 34885330 B13EAD45 196DA461 B43E813F
    Granting mode is: manual
    Last certificate issued serial number: 0x1
    CA certificate expiration timer: 01:49:13 GMT Aug 28 2007
    CRL NextUpdate timer: 01:49:16 GMT Sep 4 2004
    Current storage dir: nvram:
    Database Level: Minimum - no cert data written to storage
```

The following example shows that restoration is from a PEM archive and that the database URL is flash:

```
Router# copy tftp://172.71.71.71/backup.ser flash:mycs.ser
Destination filename [mycs.ser]?
32 bytes copied in 1.320 secs (24 bytes/sec)
Router# copy tftp://172.71.71.71/backup.crl flash:mycs.crl
Destination filename [mycs.crl]?
214 bytes copied in 1.324 secs (162 bytes/sec)
Router# configure terminal
! Because CA cert has Digital Signature usage, you need to import using the "usage-keys"
keyword
Router (config)# crypto ca import mycs pem usage-keys terminal cisco123
% Enter PEM-formatted CA certificate.
% End with a blank line or "quit" on a line by itself.
! Paste the CA cert from .pem archive.
-----BEGIN CERTIFICATE-----
MIIB9zCCAWCgAwIBAgIBATANBgkqhkiG9w0BAQQFADAPMQ0wCwYDVQQDEwRteWNz
MB4XDTA0MDkwMjIxMDI1NloXDTA3MDkwMjIxMDI1NlowDzENMAsGA1UEAxMEbXlj
czCBnzANBgkqhkiG9w0BAQEFAAOBjQAwgYkCgYEAuGnnDXJbpDDQwCuKGs5Zg2rc
K7ZJauSUotTmWYQvNx+ZmWrUs5/j9Ee5FV2YonirGBQ9mc6u163kNlrIPFck062L
GpahBhNmKDgod1o2PHTnRlZpEZNDIqU2D3hACgByxPjrY4vUnccV36ewLnQnYpp8
szEu7PYTJr5dU5ltAekCAwEAAaNjMGEwDwYDVR0TAQH/BAUwAwEB/zAOBgNVHQ8B
Af8EBAMCAYYwHwYDVR0jBBgwFoAUaEEQwYKCQ1dm9+wLYBKRTlzxaDIwHQYDVR0O
BBYEFGhBEMGCgkNXZvfsC2ASkU5c8WgyMA0GCSqGSIb3DQEBBAUAA4GBAHyhiv2C
mH+vswkBjRA1Fzzk8ttu9s5kwqG0dXp25QRUWsGlr9nsKPNdVKt3P7p0A/KochHe
eNiygiv+hDQ3FVnzsNv983le6O5jvAPxc17RO1BbfNhqvEWMsXdnjHOcUy7XerCo
+bdPcUf/eCiZueH/BEy/SZhD7yovzn2cdzBN
-----END CERTIFICATE-----

% Enter PEM-formatted encrypted private SIGNATURE key.
% End with "quit" on a line by itself.
! Paste the CA private key from .pem archive.
-----BEGIN RSA PRIVATE KEY-----
Proc-Type: 4,ENCRYPTED
DEK-Info: DES-EDE3-CBC,5053DC842B04612A

1CnlF5Pqvd0zp2NLZ7iosxzTy6nDeXPpNyJpxB5q+V29IuY8Apb6TlJCU7YrsEB/
nBTK7K76DCeGPlLpcuyEI171QmkQJ2gA0QhC0LrRo09WrINVH+b4So/y7nffZkVb
p2yDpZwqoJ8cmRH94Tie0YmzBtEh6ayOud11z53qbrsCnfSEwszt1xrW1MKrFZrk
/fTy6loHzGFzl3BDj4r5gBecExwcPp74ldHO+Ld4Nc9egG8BYkeBCsZZOQNVhXLN
I0tODOs6hP915zb6OrZFYv0NK6grTBO9D8hjNZ3U79jJzsSP7UNzIYHNTzRJiAyu
i56Oy/iHvkCSNUIK6zeIJQnW4bSoM1BqrbVPwHU6QaXUqlNzZ8SDtw7ZRZ/rHuiD
RTJMPbKquAzeuBss1132OaAUJRStjPXgyZTUbc+cWb6zATNws2yijPDTR6sRHoQL
47wHMr2Yj80VZGgkCSLAkL88ACz9TfUiVFhtfl6xMC2yuFl+WRk1XfF5VtWe5Zer
3Fn1DcBmlF7O86XUkiSHP4EV0cI6n5ZMzVLx0XAUtdAl1gD94y1V+6p9PcQHLyQA
pGRmj5IlSFw90aLafgCTbRbmC0ChIqHy91UFa1ub0130+yu7LsLGRlPmJ9NE61JR
```

```
bjRhlUXItRYWY7C4M3m/0wz6fmVQNSumJM08RHq6lUB3olzIgGIZlZkoaESrLG0p
qq2AENFemCPF0uhyVS2humMHjWuRr+jedfc/IMl7sLEgAdqCVCfV3RZVEaNXBud1
4QjkuTrwaTcRXVFbtrVioT/puyVUlpA7+k7w+F5TZwUV08mwvUEqDw==
-----END RSA PRIVATE KEY-----
quit
% Enter PEM-formatted SIGNATURE certificate.
% End with a blank line or "quit" on a line by itself.
! Paste the CA cert from .pem archive again.
-----BEGIN CERTIFICATE-----
MIIB9zCCAWCgAwIBAgIBATANBgkqhkiG9w0BAQQFADAPMQ0wCwYDVQQDEwRteWNz
MB4XDTA0MDkwMjIxMDI1NloXDTA3MDkwMjIxMDI1NlowDzENMAsGA1UEAxMEbXlj
czCBnzANBgkqhkiG9w0BAQEFAAOBjQAwgYkCgYEAuGnnDXJbpDDQwCuKGs5Zg2rc
K7ZJauSUotTmWYQvNx+ZmWrUs5/j9Ee5FV2YonirGBQ9mc6u163kNlrIPFck062L
GpahBhNmKDgod1o2PHTnRlZpEZNDIqU2D3hACgByxPjrY4vUnccV36ewLnQnYpp8
szEu7PYTJr5dU5ltAekCAwEAAaNjMGEwDwYDVR0TAQH/BAUwAwEB/zAOBgNVHQ8B
Af8EBAMCAYYwHwYDVR0jBBgwFoAUaEEQwYKCQ1dm9+wLYBKRTlzxaDIwHQYDVR0O
BBYEFGhBEMGCgkNXZvfsC2ASkU5c8WgyMA0GCSqGSIb3DQEBBAUAA4GBAHyhiv2C
mH+vswkBjRA1Fzzk8ttu9s5kwqG0dXp25QRUWsGlr9nsKPNdVKt3P7p0A/KochHe
eNiygiv+hDQ3FVnzsNv983le6O5jvAPxc17RO1BbfNhqvEWMsXdnjHOcUy7XerCo
+bdPcUf/eCiZueH/BEy/SZhD7yovzn2cdzBN
-----END CERTIFICATE-----

% Enter PEM-formatted encrypted private ENCRYPTION key.
% End with "quit" on a line by itself.
! Because the CA cert only has Digital Signature usage, skip the encryption part.
quit
% PEM files import succeeded.
Router (config)# crypto pki server mycs
Router (cs-server)# database url flash:
! Fill in any CS configuration here.
Router (cs-server)# no shutdown
% Certificate Server enabled.
Router (cs-server)# end

Router # show crypto pki server

Certificate Server mycs:
    Status: enabled
    Server's current state: enabled
    Issuer name: CN=mycs
    CA cert fingerprint: F04C2B75 E0243FBC 19806219 B1D77412
    Granting mode is: manual
    Last certificate issued serial number: 0x2
    CA certificate expiration timer: 21:02:55 GMT Sep 2 2007
    CRL NextUpdate timer: 21:02:58 GMT Sep 9 2004
    Current storage dir: flash:
    Database Level: Minimum - no cert data written to storage
```

RA Mode Certificate Server: Examples

The following output is typical of what you might see after having configured an RA mode certificate server:

```
Router-ra (config)# crypto pki trustpoint myra
Router-ra (ca-trustpoint)# enrollment url http://10.3.0.17
! Include "cn=ioscs RA" or "ou=ioscs RA" in the subject-name.
Router-ra (ca-trustpoint)# subject-name cn=myra, ou=ioscs RA, o=cisco, c=us
Router-ra (ca-trustpoint)# exit
Router-ra (config)# crypto pki server myra
Router-ra (cs-server)# mode ra
Router-ra (cs-server)# no shutdown
% Generating 1024 bit RSA keys ...[OK]
```

```
Certificate has the following attributes:
Fingerprint MD5: 32661452 0DDA3CE5 8723B469 09AB9E85
Fingerprint SHA1: 9785BBCD 6C67D27C C950E8D0 718C7A14 C0FE9C38
% Do you accept this certificate? [yes/no]: yes
Trustpoint CA certificate accepted.
% Ready to request the CA certificate.
%Some server settings cannot be changed after the CA certificate has been requested.

Are you sure you want to do this? [yes/no]: yes
%
% Start certificate enrollment ..
% Create a challenge password. You will need to verbally provide this
   password to the CA administrator in order to revoke your certificate.
   For security reasons your password will not be saved in the configuration.
   Please make a note of it.

Password:
Re-enter password:

% The subject name in the certificate will include: cn=myra, ou=ioscs RA, o=cisco, c=us
% The subject name in the certificate will include: Router-ra.cisco.com
% Include the router serial number in the subject name? [yes/no]: no
% Include an IP address in the subject name? [no]: no
Request certificate from CA? [yes/no]: yes
% Certificate request sent to Certificate Authority
% The certificate request fingerprint will be displayed.
% The 'show crypto pki certificate' command will also show the fingerprint.

% Enrollment in progress...
Router-ra (cs-server)#
Sep 15 22:32:40.197: CRYPTO_PKI:  Certificate Request Fingerprint MD5: 82B41A76 AF4EC87D
AAF093CD 07747D3A
Sep 15 22:32:40.201: CRYPTO_PKI:  Certificate Request Fingerprint SHA1: 897CDF40 C6563EAA
0FED05F7 0115FD3A 4FFC5231
Sep 15 22:34:00.366: %PKI-6-CERTRET: Certificate received from Certificate Authority
Router-ra (cs-server)#
Router-ra(cs-server)# end

Router-ra# show crypto pki server

Certificate Server myra:
    Status: enabled
    Issuer name: CN=myra
    CA cert fingerprint: 32661452 0DDA3CE5 8723B469 09AB9E85
    ! Note that the certificate server is running in RA mode
    Server configured in RA mode
    RA cert fingerprint: C65F5724 0E63B3CC BE7AE016 BE0D34FE
    Granting mode is: manual
    Current storage dir: nvram:
    Database Level: Minimum - no cert data written to storage
```

The following output shows the enrollment request database of the issuing certificate server after the RA has been enabled:

Note The RA certificate request is recognized by the issuing certificate server because "ou=ioscs RA" is listed in the subject name.

```
Router-ca# crypto pki server mycs info request
Enrollment Request Database:
```

```
Subordinate CA certificate requests:
ReqID  State      Fingerprint                      SubjectName
--------------------------------------------------------------

! The request is identified as RA certificate request.
RA certificate requests:
ReqID  State      Fingerprint                      SubjectName
--------------------------------------------------------------
12     pending    88F547A407FA0C90F97CDE8900A30CB0
hostname=Router-ra.cisco.com,cn=myra,ou=ioscs RA,o=cisco,c=us

Router certificates requests:
ReqID   State     Fingerprint                      SubjectName
--------------------------------------------------------------

! Issue the RA certificate.
Router-ca# crypto pki server mycs grant 12
```

The following output shows that the issuing certificate server is configured to issue a certificate automatically if the request comes from an RA:

```
Router-ca(config)# crypto pki server mycs
Router-ca (cs-server)# grant ra-auto
% This will cause all certificate requests already authorized by known RAs to be
automatically granted.

Are you sure you want to do this? [yes/no]: yes
Router-ca (cs-server)# end
Router-ca# show crypto pki server
Certificate Server mycs:
    Status: enabled
    Server's current state: enabled
    Issuer name: CN=mycs
    CA cert fingerprint: 32661452 0DDA3CE5 8723B469 09AB9E85
    ! Note that the certificate server will issue certificate for requests from the RA.
    Granting mode is: auto for RA-authorized requests, manual otherwise
    Last certificate issued serial number: 0x2
    CA certificate expiration timer: 22:29:37 GMT Sep 15 2007
    CRL NextUpdate timer: 22:29:39 GMT Sep 22 2004
    Current storage dir: nvram:
    Database Level: Minimum - no cert data written to storage
```

Additional References

The following sections provide references related to the Cisco IOS Certificate Server feature.

Related Documents

Related Topic	Document Title
Additional certificate enrollment commands	*Cisco IOS Security Command Reference*, Release 12.3T
Additional certificate enrollment tasks	*Cisco IOS Security Configuration Guide*, Release 12.3

Related Topic	Document Title
Enabling HTTP servers and configuring time clocks	*Cisco IOS Configuration Fundamentals and Network Management Configuration Guide*, Release 12.3T
Restoring a certificate server using backup for a PEM-formatted CA certificate and CA key	*Import of RSA Key Pair and Certificates in PEM Format*, Release 12.3(4)T

Standards

Standards	Title
None	—

MIBs

MIBs	MIBs Link
None	To locate and download MIBs for selected platforms, Cisco IOS releases, and feature sets, use Cisco MIB Locator found at the following URL: http://www.cisco.com/go/mibs

RFCs

RFCs	Title
RFC 2459	*Internet X.509 Public Key Infrastructure Certificate and CRL Profile*

Technical Assistance

Description	Link
Technical Assistance Center (TAC) home page, containing 30,000 pages of searchable technical content, including links to products, technologies, solutions, technical tips, and tools. Registered Cisco.com users can log in from this page to access even more content.	http://www.cisco.com/public/support/tac/home.shtml

Command Reference

This section documents new commands. All other commands used with this feature are documented in the Cisco IOS Release 12.3 T command reference publications.

Global Configuration Command

- **crypto pki server**

Certificate Server Configuration Commands

- cdp-url
- database (certificate server)
- database archive
- database level
- database url
- grant auto
- grant none
- grant ra-auto
- issuer-name
- lifetime (certificate server)
- lifetime crl
- lifetime enrollment-request
- shutdown

Privileged EXEC Commands

- crypto pki server grant
- crypto pki server info crl
- crypto pki server info requests
- crypto pki server password generate
- crypto pki server reject
- crypto pki server remove
- crypto pki server request pkcs10
- crypto pki server revoke
- debug crypto pki server
- show crypto pki server

cdp-url

To specify a certificate revocation list (CRL) distribution point (CDP) to be used in certificates that are issued by the certificate server, use the **cdp-url** command in certificate server configuration mode. To remove a CDP from your configuration, use the **no** form of this command.

cdp-url *url*

no cdp-url *url*

Syntax Description

url	HTTP URL where CRLs are published.

Defaults

When verifying a certificate that does not have a specified CDP, Cisco IOS public key infrastructure (PKI) clients will use Simple Certificate Enrollment Protocol (SCEP) to retrieve the CRL directly from their configured certificate server.

Command Modes

Certificate server configuration

Command History

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

Usage Guidelines

CRLs are issued once every specified time period via the **lifetime crl** command. Thereafter, the CRL is written to the specified database location as *ca-label*.crl (where *ca-label* is the name of the certificate server). 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. If the **cdp-url** command is not specified, the CDP certificate extension will not be included in the certificates that are issued by the certificate server. Thus, Cisco IOS public key infrastructure (PKI)l clients will automatically use SCEP to retrieve a CRL from the certificate server, which puts an additional load on the certificate server because it must provide SCEP server support to for each CRL request.

Note The CRL will always be available via SCEP, which is enabled by default, if the HTTP server is enabled.

Note For large PKI deployments, it is recommended that you configure an HTTP-based CDP; for example, cdp-url http://myhttpserver.company.com/mycs.crl.

The CDP URL may be changed after the certificate server is running, but existing certificates will not be reissued with the new CDP that is specified via the **cdp-url** command.

The certificate server supports only one CDP; thus, all certificates that are issued include the same CDP.

Examples

The following example shows how to configure a CDP:

```
Router(config)# crypto pki server aaa
Router(cs-server)# database level minimum
Router(cs-server)# database url tftp://10.1.1.1/johndoe/
Router(cs-server)# issuer-name CN=aaa
Router(cs-server)# cdp-url http://msca-root.cisco.com/certEnroll/aaa.crl
```

Verifying a CDP Configuration

The following example is sample output from the **show crypto ca certificates** command, which allows you to verify the specified CDP. In this example, the CDP is "http://msca-root.cisco.com/certEnroll/aaa.crl."

```
Router# show crypto ca certificates

Certificate
  Status: Available
  Certificate Serial Number: 03
  Certificate Usage: General Purpose
  Issuer:
    CN = aaa
  Subject:
    Name: Router.cisco.com
    OID.1.2.840.113549.1.9.2 = Router.cisco.com
  CRL Distribution Point:
    http://msca-root.cisco.com/certEnroll/aaa.crl
  Validity Date:
    start date: 18:44:49 GMT Jun 6 2003
    end   date: 18:44:49 GMT Jun 5 2004
    renew date: 00:00:00 GMT Jan 1 1970
  Associated Trustpoints: bbb
```

Related Commands

Command	Description
crypto pki server	Enables a Cisco IOS certificate server and enters certificate server configuration mode.
crypto pki server revoke	Revokes a certificate based on its serial number.
lifetime crl	Defines the lifetime of the CRL that is used by the certificate server.
show crypto ca certificates	Displays information about your certificate, the certification authority certificate, and any registration authority certificates.

crypto pki server

To enable a Cisco IOS certificate server and enter certificate server configuration mode, use the **crypto pki server** command in global configuration mode. To disable a certificate server (which is the default functionality), use the **no** form of this command.

crypto pki server *cs-label*

no crypto pki server *cs-label*

Syntax Description

cs-label	Name of the certificate server. **Note** The certificate server name should not exceed 13 characters.

Defaults

A certificate server is not enabled.

Command Modes

Global configuration

Command History

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

Usage Guidelines

A certificate server allows you to more easily deploy public key infrastructure (PKI) by defining default behavior, which limits user interface complexity. To define the functionality of the certificate server, you can use any of the following certificate server configuration mode commands:

- **database (certificate server)**—Requires a username or password to be issued when accessing a database storage location.
- **database level**—Controls what type of data is stored in the certificate enrollment database.
- **database url—**Specifies the location where all database entries for the certificate server will be written out.
- **grant auto**—Specifies automatic certificate enrollment.

Note This command can be used for testing and building simple networks; however, it is recommended that you do not issue this command if your network is generally accessible.

- **issuer-name—**Specifies the distinguished name (DN) as the certification authority (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.

Note All of these commands are optional; thus, any basic certificate server functionality that is not specified via the command-line interface (CLI) will use the default value.

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
```

Related Commands

Command	Description
crypto pki server info requests	Displays all outstanding certificate enrollment requests.
ip http server	Enables an HTTP server on your network.

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.

Defaults

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

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.

Defaults

No default behavior or values

Command Modes

Privileged EXEC

Command History

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

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

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.

Defaults

No default behavior or values

Command Modes

Privileged EXEC

Command History

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

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 Simple Certificate Enrollment Protocol (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 Table 1.

Table 3 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.
```

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.

Defaults

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.

Defaults

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.

Defaults

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.

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

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 Table 4. **Note** The request file name should have a .req extension and the granted certificate file name will have a .crt extension (see the URL example in the section "Examples."
terminal	Certificate requests will be manually pasted from the console terminal, and the granted certificate will be displayed on the console.
pem	(Optional) Privacy-enhanced mail (PEM) headers are automatically added to the certificate after the certificate is granted.

Defaults

No default behavior or values.

Command Modes

Privileged EXEC

Command History

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

Usage Guidelines

Use the **crypto pki server request pkcs10** command to manually add either a base64-encoded or PEM-formatted 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. If the **pem** keyword is specified, PEM headers are also added to the certificate.

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. Table 4 lists available file system options.

Table 4 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

Table 4 crypto pki server request pkcs10 Options (continued)

Location	Description
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://172.69.1.129/router5
% Retrieving Base64 encoded or PEM formatted PKCS10 enrollment request...
Reading file from tftp://172.69.1.129/router5.req
Loading router5.req from 172.69.1.129 (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://172.69.1.129/router5.crt!
```

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 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).

Defaults

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.

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 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.

database (certificate server)

To require a username or password to be issued when accessing a database storage location, use the **database** command in certificate server configuration mode. To return to the default value, use the **no** form of this command.

database username *username* [**password** *password*]

no database username *username* [**password** *password*]

Syntax Description

username *username*	When prompted, a username will be used to access a storage location.
password *password*	(Optional) When prompted, a password will be used to access a storage location.

Defaults

This command is not enabled.

Command Modes

Certificate server configuration

Command History

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

Usage Guidelines

All information stored in the remote database is public: there are no private keys stored in the database location. Using a password helps to protect against a potential attacker who can change the contents of the .ser or .crl file. If the contents of the files are changed, the certificate server may shut down, refusing to either issue new certificates or respond to simple certificate enrollment protocol (SCEP) requests until the files are restored.

It is good security practice to protect all information exchanges with the database server using IP Security (IPSec). To protect your information, use a remote database to obtain the appropriate certificates and setup the necessary IPSec connections to protect all future access to the database server.

Examples

The following example shows how to specify the username “mystorage” when accessing the complete database that is stored on an external TFTP server:

```
Router (config)# ip http server
Router (config)# crypto pki server myserver
Router (cs-server)# database level complete
Router (cs-server)# database url tftp://mytftp
Router (cs-server)# database username mystorage
```

Related Commands

Command	Description
crypto pki server	Enables a Cisco IOS certificate server and enters PKI configuration mode.
database level	Controls what type of data is stored in the database.
database url	Specifies the location where all database entries for the certificate server will be written out.

database archive

To set the certification authority (CA) certificate and CA key archive format—and the password—to encrypt this CA certificate and CA key archive file, use the **database archive** command in certificate server configuration mode. To disable the autoarchive feature, use the **no** form of this command.

database archive {**pkcs12** | **pem**} [**password** *password*]

no database archive {**pkcs12** | **pem**} [**password** *password*]

Syntax Description

pkcs12	Export as a PKCS12 file. The default is PKCS12.
pem	Export as a privacy-enhanced mail (PEM) file.
password *password*	(Optional) Password to encrypt the CA certificate and CA key. The password must be at least eight characters. If a password is not specified, you will be prompted for the password after the **no shutdown** command has been issued for the first time. When the password is entered, it will be encrypted.

Defaults

The archive format is PKCS (that is, the CA certificate and CA key are exported into a PKCS12 file, and you will be prompted for the password when the certificate server is turned on the first time).

Command Modes

Certificate server configuration

Command History

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

Usage Guidelines

Use this command to configure the autoarchive format for the CA certificate and CA key. The archive can later be used to restore your certificate server.

If autoarchiving is not explicitly turned off when the certificate server is first enabled (using the **no shutdown** command), the CA certificate and CA key will be archived automatically, applying the following rule:

- The CA key must be (1) manually generated and marked “exportable” or (2) automatically generated by the certificate server (it will be marked nonexportable).

Note It is strongly recommended that if the password is included in the configuration to suppress the prompt after the **no shutdown** command, the password should be removed from the configuration after the archiving is finished.

Examples

The following example shows that certificate server autoarchiving has been enabled. The CA certificate and CA key format has been set to PEM, and the password has been set as cisco123.

```
Router (config)# crypto pki server myserver
Router (cs-server)# database archive pem password cisco123
```

Related Commands

Command	Description
crypto pki server	Enables a Cisco IOS certificate server.

database level

To control what type of data is stored in the certificate enrollment database, use the **database level** command in certificate server configuration mode. To return to the default functionality, use the **no** form of this command.

database level {**minimal** | **names** | **complete**}

no database level {**minimal** | **names** | **complete**}

Syntax Description

minimal	Enough information is stored only to continue issuing new certificates without conflict. This is the default functionality.
names	The serial number and subject name of each certificate are stored in the database, providing enough information for the administrator to find and revoke and particular certificate, if necessary.
complete	Each issued certificate is written to the database. If this keyword is used, you should enable the **database url** command; see "Usage Guidelines" for more information.

Defaults

minimal

Command Modes

Certificate server configuration

Command History

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

Usage Guidelines

The **database level** command is used to describe the database of certificates and certification authority (CA) states. After the user downgrades the database level, the old data stays the same and the new data is logged at the new level.

minimum Level

The *ca-label*.ser file is always available. It contains the previously issued certificate's serial number, which is always 1. If the .ser file is unavailable and the CA server has a self-signed certificate in the local configuration, the CA server will refuse to issue new certificates.

The file format is as follows:

```
last_serial = serial-number
```

names Level

The *serial-number*.cnm file, which is written for each issued certificate, contains the "human readable decoded subject name" of the issued certificate and the "der encoded" values. This file can also include a certificate expiration date and the current status. (The **minimum** level files are also written out.)

The file format is as follows:

```
subjectname_der = <base64 encoded der value>
subjectname_str = <human readable decode subjectname>
expiration = <expiration date>
status = valid | revoked
```

complete Level

The *serial-number*.cer file, which is written for each issued certificate, is the binary certificate without additional encoding. (The **minimum** and **names** level files are also written out.)

The **complete** level produces a large amount of information, so you may want to store all database entries on an external TFTP server via the **database url** command unless your router does one of the following:

- Issues only a small number of certificates
- Has a local file system that is designed to support a large number of write operations and has sufficient storage for the certificates that are being issued

Examples

The following example shows how configure a minimum database to be stored on the local system:

```
Router#(config) ip http server
Router#(config) crypto pki server myserver
Router#(cs-server) database level minimum
Router#(cs-server) database url nvram:
Router#(cs-server) issuer-name CN = ipsec_cs,L = Santa Cruz,C = US
```

Related Commands

Command	Description
crypto pki server	Enables a Cisco IOS certificate server and enters PKI configuration mode.
database url	Specifies the location where all database entries for the certificate server will be written out.

database url

To specify the location where all database entries for the certificate server will be written out, use the **database url** command in certificate server configuration mode. To return to the default location, use the **no** form of this command.

database url *root-url*

no database url *root-url*

Syntax Description

root-url	Location where database entries will be written out. The URL can be any URL that is supported by the Cisco IOS file system (IFS).

Defaults

The default location is flash.

Command Modes

Certificate server configuration

Command History

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

Usage Guidelines

After you create a certificate server via the **crypto pki server** command, use the **database url** command if you want to specify a combined list of all the certificates that have been issued and the current command revocation list (CRL). The CRL is written to the certificate enrollment database as *ca-label*.crl (where *ca-label* is the name of the certificate server).

Note Although issuing the **database url** command is not required, it is recommended. Unless your router has a local file system that is designed for a large number of write operations and has sufficient storage for the certificates that are issued, you should issue this command.

Cisco IOS File System

The router uses any file system that is supported by your version of Cisco IOS software (such as TFTP, FTP, flash, and NVRAM) to send a certificate request and to receive the issued certificate. A user may wish to enable IFS certificate enrollment when his or her certification authority (CA) does not support Simple Certificate Enrollment Protocol (SCEP).

Examples

The following example shows how to configure all database entries to be written out to a TFTP server:

```
Router#(config) ip http server
Router#(config) crypto pki server myserver
Router#(cs-server) database level complete
Router#(cs-server) database url tftp://mytftp
```

Verifying the Database URL

To ensure that the specified URL is working correctly, configure the **database url** command before you issue the **no shutdown** command on the certificate server for the first time. If the URL is broken, you will see output as follows:

```
Router(config)# crypto pki server mycs
Router(cs-server)# database url ftp://myftpserver
Router(cs-server)# no shutdown
% Once you start the server, you can no longer change some of
% the configuration.
Are you sure you want to do this? [yes/no]: yes
Translating "myftpserver"

% Failed to generate CA certificate - 0xFFFFFFFF
% The Certificate Server has been disabled.
```

Related Commands

Command	Description
crypto pki server	Enables a Cisco IOS certificate server and enters PKI configuration mode.
database level	Controls what type of data is stored in the database.

debug crypto pki server

To enable debugging for a crypto public key infrastructure (PKI) certificate server, use the **debug crypto pki server** command in privileged EXEC mode. To disable debugging, use the **no** form of this command.

debug crypto pki server

no debug crypto pki server

Syntax Description

This command has no arguments or keywords.

Command Modes

Privileged EXEC

Command History

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

Examples

The following example shows how to enable debugging for a certificate server. This example also contains sample debug messages, which allow users to troubleshoot the various certificate-request-related stages and tasks that are handled by the certificate server.

```
Router# debug crypto pki server

Crypto PKI Certificate Server debugging is on

Oct 15 19:50:41.003:CRYPTO_CS:old RA cert flag 0x4
Oct 15 19:50:41.003:CRYPTO_CS:new RA cert flag 0x1000C
Oct 15 19:50:41.003:CRYPTO_CS:nvram filesystem
Oct 15 19:50:41.279:CRYPTO_CS:serial number 0x1 written.
Oct 15 19:50:53.383:CRYPTO_CS:created a new serial file.
Oct 15 19:50:53.383:CRYPTO_CS:SCEP server started
Oct 15 19:50:53.419:%SYS-5-CONFIG_I:Configured from console by console
Oct 15 19:50:53.731:CRYPTO_CS:received a SCEP GetCACert request
Oct 15 19:50:53.739:CRYPTO_CS:CA certificate sent
Oct 15 19:50:54.355:CRYPTO_CS:received a SCEP GetCACert request
Oct 15 19:50:54.363:CRYPTO_CS:CA certificate sent
Oct 15 19:50:57.791:CRYPTO_CS:received a SCEP request
Oct 15 19:50:57.795:CRYPTO_CS:read SCEP:registered and bound service
SCEP_READ_DB_8
Oct 15 19:50:57.947:CRYPTO_CS:scep msg type - 19
Oct 15 19:50:57.947:CRYPTO_CS:trans id -
3673CE2FF0235A4AE6F26242B00A4BB4
Oct 15 19:50:58.679:CRYPTO_CS:read SCEP:unregistered and unbound
service SCEP_READ_DB_8
Oct 15 19:50:58.683:CRYPTO_CS:received an enrollment request
Oct 15 19:50:58.691:CRYPTO_CS:request has been authorized, transaction
id=3673CE2FF0235A4AE6F26242B00A4BB4
Oct 15 19:50:58.699:CRYPTO_CS:byte 2 in key usage in PKCS#10 is 0x7
Oct 15 19:50:58.699:CRYPTO_CS:signature
Oct 15 19:50:58.699:CRYPTO_CS:key_usage is 1
Oct 15 19:50:58.703:CRYPTO_CS:enrollment request with pendingID 1 sent
to the CA
```

```
Oct 15 19:50:58.707:CRYPTO_CS:write SCEP:registered and bound service
SCEP_WRTE_DB_8
Oct 15 19:50:59.531:CRYPTO_CS:write SCEP:unregistered and unbound
service SCEP_WRTE_DB_8

.....

Oct 15 19:53:08.403:CRYPTO_CS:CS_RA_REQUEST:save cert in dbase,
pending id = 2
Oct 15 19:53:08.403:CRYPTO_CS:enrollment request 2 granted
Oct 15 19:53:08.403:CRYPTO_PKI:All enrollment requests completed for
trustpoint ra.
Oct 15 19:53:08.403:%CRYPTO-6-CERTRET:Certificate received from
Certificate Authority
Oct 15 19:53:08.403:CRYPTO_PKI:All enrollment requests completed for
trustpoint ra.
Oct 15 19:53:08.403:CRYPTO_PKI:All enrollment requests completed for
trustpoint ra.
Oct 15 19:53:08.407:CRYPTO_PKI:All enrollment requests completed for
trustpoint ra.
Oct 15 19:53:19.623:IPSEC(key_engine):major = 1
Oct 15 19:53:19.623:IPSEC(key_engine):expired_timer:skip ...
Oct 15 19:53:35.707:CRYPTO_CS:received a SCEP request
Oct 15 19:53:35.711:CRYPTO_CS:read SCEP:registered and bound service
SCEP_READ_DB_14
Oct 15 19:53:35.859:CRYPTO_CS:scep msg type - 20
Oct 15 19:53:35.859:CRYPTO_CS:trans id -
4D774FFE2F7CA9991A7F6A785E803E77
Oct 15 19:53:36.591:CRYPTO_CS:read SCEP:unregistered and unbound
service SCEP_READ_DB_14
Oct 15 19:53:36.595:CRYPTO_CS:received an enrollment request
Oct 15 19:53:36.595:CRYPTO_CS:write SCEP:registered and bound service
SCEP_WRTE_DB_14
Oct 15 19:53:37.623:CRYPTO_CS:write SCEP:unregistered and unbound
service SCEP_WRTE_DB_14
Oct 15 19:53:37.631:CRYPTO_CS:Certificate sent to requestor
```

Related Commands

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

grant auto

To specify automatic certificate enrollment, use the **grant auto** command in certificate server configuration mode. To disable automatic certificate enrollment, use the **no** form of this command.

grant auto

no grant auto

Syntax Description

This command has no arguments or keywords.

Defaults

Certificate enrollment is manual; that is, authorization is required.

Command Modes

Certificate server configuration

Command History

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

Usage Guidelines

The **grant auto** command should be used only when testing and building simple networks. This command must be disabled before the network is accessible by the Internet.

Note This command can be used for testing and building simple networks; however, it is recommended that you do not issue this command if your network is generally accessible.

Examples

The following example shows how to enable automatic certificate enrollment for the certificate server "myserver":

```
Router#(config) ip http server
Router#(config) crypto pki server myserver
Router#(cs-server) database level minimum
Router#(cs-server)# grant auto
% This will cause all certificate requests to be automatically granted.

Are you sure you want to do this? [yes/no]: yes
```

Related Commands

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

grant none

To specify all certificate requests to be rejected, use the **grant none** command in certificate server configuration mode. To disable automatic rejection of certificate enrollment, use the **no** form of this command.

grant none

no grant none

Syntax Description

This command has no arguments or keywords.

Defaults

Certificate enrollment is manual; that is, authorization is required.

Command Modes

Certificate server configuration

Command History

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

Examples

The following example shows how to automatically reject all certificate enrollment requests for the certificate server “myserver”:

```
Router#(config) ip http server
Router#(config) crypto pki server myserver
Router#(cs-server) database level minimum
Router#(cs-server)# grant none
```

Related Commands

Command	Description
crypto pki server	Enables a Cisco IOS certificate server and enters certificate server configuration mode.
grant auto	Specifies automatic certificate enrollment.

grant ra-auto

To specify that all enrollment requests from a Registration Authority (RA) be granted automatically, use the **grant ra-auto** command in certificate server configuration mode. To disable automatic certificate enrollment, use the **no** form of this command.

grant ra-auto

no grant ra-auto

Syntax Description

This command has no arguments or keywords.

Defaults

Certificate enrollment is manual; that is, authorization is required.

Command Modes

Certificate server configuration

Command History

Release	Modification
12.3(7)T	This command was introduced.

Usage Guidelines

When grant ra-auto mode is configured on the issuing certificate server, ensure that the RA mode certificate server is running in manual grant mode so that enrollment requests are authorized individually by the RA.

Note For the **grant ra-auto** command to work, you have to include “cn=ioscs RA” or “ou=ioscs RA” in the subject name of the RA certificate.

Examples

The following output shows that the issuing certificate server is configured to issue a certificate automatically if the request comes from an RA:

```
Router (config)# crypto pki server myserver
Router-ca (cs-server)# grant ra-auto
% This will cause all certificate requests that are already authorized by known RAs to be
automatically granted.

Are you sure you want to do this? [yes/no]:yes
```

Related Commands

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

issuer-name

To specify the distinguished name (DN) as the certification authority (CA) issuer name for the certificate server, use the **issuer-name** command in certificate server configuration mode. To clear the issuer name and return to the default, use the **no** form of this command.

issuer-name *DN-string*

no issuer-name *DN-string*

Syntax Description

DN-string	Name of the DN string.

Defaults

If the issuer name is not configured, **CN** = *cs-label*

Command Modes

Certificate server configuration

Command History

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

Usage Guidelines

The DN-string value cannot be changed after the certificate server generates its signed certificate.

Examples

The following example shows how to define an issuer name for the certificate server “mycertserver”:

```
Router(config)# ip http server
Router(config)# crypto pki server mycertserver
Router(cs-server)# database level minimal
Router(cs-server)# database url nvram:
Router(cs-server)# issuer-name CN = ipsec_cs,L = My Town,C = US
```

Related Commands

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

lifetime (certificate server)

To specify the lifetime of the certification authority (CA) or a certificate, use the **lifetime** command in certificate server configuration mode. To return to the default lifetime values, use the **no** form of this command.

lifetime {**ca-certificate** | **certificate**} *time*

no lifetime {**ca-certificate** | **certificate**} *time*

Syntax Description

ca-certificate	Lifetime is for the CA certificate of the certificate server.
certificate	Lifetime is for the certificate of the certificate server. The maximum certificate lifetime is one month less than the expiration date of the CA certificate's lifetime.
time	Lifetime value in days. Valid values range from 1 day to 1825 days. All certificates are valid on the date that they are issued.

Defaults

The default CA certificate lifetime is 3 years.

The default certificate lifetime is 1 year.

Command Modes

Certificate server configuration

Command History

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

Usage Guidelines

After you enable a certificate server via the **crypto pki server** command, use the **lifetime** command if you wish to specify lifetime values other than the default values for the CA certificate and the certificate of the certificate server.

After the certificate generates its signed certificate, the lifetime cannot be changed.

Examples

The following example shows how to set the lifetime value for the CA to 30 days:

```
Router(config)# ip http server
Router(config)# crypto pki server mycertserver
Router(cs-server)# lifetime ca certificate 30
```

Related Commands

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

lifetime crl

To define the lifetime of the certificate revocation list (CRL) that is used by the certificate server, use the **lifetime crl** command in certificate server configuration mode. To return to the default value of 1 week, use the **no** form of this command.

lifetime crl *time*

no lifetime crl *time*

Syntax Description

time	Lifetime value, in hours, of the CRL. Maximum lifetime value is 336 hours (2 weeks). The default value is 168 hours (1 week).

Defaults

168 hours (1 week)

Command Modes

Certificate server configuration

Command History

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

Usage Guidelines

After you create a certificate server via the **crypto pki server** command, use the **lifetime crl** command if you want to specify a value other than the default value for the CRL. The lifetime value is added to the CRL when the CRL is created.

The CRL is written to the specified database location as *ca-label*.crl.

Examples

The following example shows how to set the lifetime value for the CRL to 24 hours:

```
Router(config)# ip http server
Router(config)# crypto pki server mycertserver
Router(cs-server)# lifetime crl 24
```

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 PKI configuration mode.

lifetime enrollment-request

To specify how long an enrollment request should stay in the enrollment database, use the **lifetime enrollment-request** command in certificate server configuration mode. To return to the default value of 1 week, use the **no** form of this command.

lifetime enrollment-request *time*

no lifetime enrollment-request

Syntax Description

time	Lifetime value, in hours, of an enrollment request. The maximum lifetime value is 1000 hours. The default value is 168 hours (1 week).

Defaults

Lifetime value default is 168 hours.

Command Modes

Certificate server configuration

Command History

Release	Modification
12.3(7)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. The request is left in the Enrollment Request Database for the lifetime of the enrollment request until the client polls the certificate server for the result of the request.

Examples

The following example shows how to set the lifetime value for the enrollment request to 24 hours:

```
Router (config)# crypto pki server mycs
Router (cs-server)# lifetime enrollment-request 24
```

Related Commands

Command	Description
crypto pki server	Enables a Cisco IOS certificate server.
crypto pki server grant	Grants all or certain SCEP requests.
crypto pki server remove	Removes enrollment requests that are in the certificate server Enrollment Request Database.

show crypto pki server

To display the current state and configuration of the certificate server, use the **show crypto pki server** command in privileged EXEC mode.

show crypto pki server

Syntax Description

This command has no arguments or keywords.

Command Modes

Privileged EXEC

Command History

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

Usage Guidelines

At startup, the certificate server must check the current configuration before issuing any certificates. As it starts up, the certificate server transitions through the states defined in Table 1. Use the **show crypto pki server** command to display the state of the certificate server.

Table 5 Certificate Server Startup State Descriptions

Certificate Server State	Description
configured	The server is available and has generated the certificate server certificates.
storage configuration incomplete	The server is verifying that the configured storage location is available.
waiting for HTTP server	The server is verifying that the HTTP server is running.
waiting for time setting	The server is verifying that the time has been set.

Examples

The following example is sample output for the **show crypto pki server** command:

```
Router# show crypto pki server

Certificate Server status: disabled, storage configuration incomplete
    Granting mode is: manual
    Last certificate issued serial number: 0
    CA certificate expiration timer: 21:29:38 GMT Jun 5 2006
    CRL NextUpdate timer: 21:31:39 GMT Jun 6 2003
    Current storage dir: ftp://myftpserver
    Database Level: Minimum - no cert data written to storage
```

Table 2 describes the significant fields shown in the display.

Table 6 show crypto pki server Field Descriptions

Field	Description
Granting mode is	Specifies whether certificate enrollment requests should be granted manually (which is the default) or automatic (via the **grant automatic** command). **Note** The **grant automatic** command should be used *only* when testing and building simple networks. This command *must* be disabled before the network is accessible by the Internet.
Last certificate issued serial number	The serial number of the latest certificate. (To specify the distinguished name (DN) as the certification authority (CA) issuer name, use the **issuer-name** command.)
CA certificate expiration timer	The expiration date for the CA certificate. (To specify the expiration date, use the **lifetime** command.)
CRL NextUpdate timer	The next time the certificate revocation list (CRL) will be updated. (To specify the CRL lifetime, in hours, use the **lifetime crl** command.
Current storage dir	The location where all database entries for the certificate server will be written out. (To specify a location, use the **database url** command.)
Database Level	The type of data that is stored in the certificate enrollment database—minimal, names, or complete. (To specify the data type to be stored, use **database level** command.)

Related Commands

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

shutdown

To allow a certificate server to be disabled without removing the configuration, use the **shutdown** command in certificate server configuration mode. To reenable the certificate server, use the **no** form of this command.

shutdown

no shutdown

Syntax Description

This command has no arguments or keywords.

Defaults

no shutdown

Command Modes

Certificate server configuration

Command History

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

Usage Guidelines

You should issue the **no shutdown** command only after you have completely configured your certificate server.

The **shutdown** command disables the certificate server. If you prefer to disable simple certificate enrollment protocol (SCEP) but still want the certificate server for manual certificate enrollment, use the **no ip http server** command.

Examples

To ensure that the specified URL is working correctly, configure the **database url** command before you issue the **no shutdown** command on the certificate server for the first time. If the URL is broken, you will see output as follows:

```
Router(config)# crypto pki server mycs
Router(cs-server)# database url ftp://myftpserver
Router(cs-server)# no shutdown
% Once you start the server, you can no longer change some of
% the configuration.
Are you sure you want to do this? [yes/no]: yes
Translating "myftpserver"

% Failed to generate CA certificate - 0xFFFFFFFF
% The Certificate Server has been disabled.
```

Related Commands

Command	Description
crypto pki server	Enables a Cisco IOS certificate server and enters PKI configuration mode.
database url	Specifies the location where all database entries for the certificate server will be written out.
ip http server	Enables an HTTP server on your network.