

Certification Authority Interoperability

Feature Summary

Certification Authority (CA) interoperability is provided in support of the IP Security (IPSec) standard. CA interoperability permits Cisco IOS devices and CAs to communicate so that your Cisco IOS device can obtain and use digital certificates from the CA. Although IPSec can be implemented in your network without the use of a CA, using a CA provides manageability and scalability for IPSec.

For background and configuration information for IPSec, see the “IPSec Network Security” feature documentation.

Benefits

Without CA interoperability, Cisco IOS devices could not use CAs when deploying IPSec. CAs provide a manageable, scalable solution for IPSec networks. For details, see “Overview of Certificate Authorities.”

Supported Standards

Cisco supports the following standards with this feature:

- **IPSec—IPSec**—IP Security Protocol. IPSec is a framework of open standards that provides data confidentiality, data integrity, and data authentication between participating peers. IPSec provides these security services at the IP layer; it uses IKE to handle negotiation of protocols and algorithms based on local policy, and to generate the encryption and authentication keys to be used by IPSec. IPSec can be used to protect one or more data flows between a pair of hosts, between a pair of security gateways, or between a security gateway and a host.

For more information on IPSec, see the “IPSec Network Security” feature documentation.

- **Internet Key Exchange (IKE)**—A hybrid protocol which implements Oakley and Skeme key exchanges inside the ISAKMP framework. While IKE can be used with other protocols, its initial implementation is with the IPSec protocol. IKE provides authentication of the IPSec peers, negotiates IPSec keys, and negotiates IPSec security associations.

For more information on IKE, see the “Internet Key Exchange Security Protocol” feature documentation.

- **Public-Key Cryptography Standard #7 (PKCS #7)**—A standard from RSA Data Security, Inc. used to encrypt and sign certificate enrollment messages.

- **Public-Key Cryptography Standard #10 (PKCS #10)**—A standard syntax from RSA Data Security, Inc. for certificate requests.
- **RSA Keys**—RSA is the public key cryptographic system developed by Ron Rivest, Adi Shamir, and Leonard Adleman. RSA keys come in pairs: one public key and one private key.
- **X.509v3 certificates**—Certificate support which allows the IPSec-protected network to scale by providing the equivalent of a digital ID card to each device. When two devices wish to communicate, they exchange digital certificates to prove their identity (thus removing the need to manually exchange public keys with each peer or to manually specify a shared key at each peer). These certificates are obtained from a Certification Authority (CA). X.509 is part of the X.500 standard by the ITU.

Restrictions

This feature is useful and should be configured only when you also configure both IPSec and IKE in your network.

Platforms

This feature is supported on these platforms:

- Cisco 1600 series
- Cisco 2500 series
- Cisco 2600 series
- Cisco 3600 series
- Cisco 4000 series (Cisco 4000, 4000-M, 4500, 4500-M, 4700, 4700-M)
- Cisco 7200 series
- Cisco 7500 series
- Cisco AS5300

Prerequisites

You need to have a Certification Authority (CA) available to your network before you configure this interoperability feature. The CA must support Cisco's PKI protocol, the certificate enrollment protocol (CEP).

Supported MIBs and RFCs

No new MIBs are supported by this feature.

This feature will support a number of RFCs which are currently under development.

IPSec is described by a collection of documents developed within the IETF IP Security Working Group (WG). The original set of WG documents are RFCs 1825-1829. These RFCs are being deprecated by the WG in favor of new Internet Drafts which are currently undergoing IETF Last Call.

For more information on IPSec, see the "IPSec Network Security" feature documentation.

Overview of Certificate Authorities

This section provides background information about CAs, including the following:

- Purpose of CAs
- Implementing IPSec Without CAs
- Implementing IPSec With CAs
- How CA Certificates Are Used by IPSec Devices
- About Registration Authorities

Purpose of CAs

CAs are responsible for managing certificate requests and issuing certificates to participating IPSec network devices. These services provide centralized key management for the participating devices.

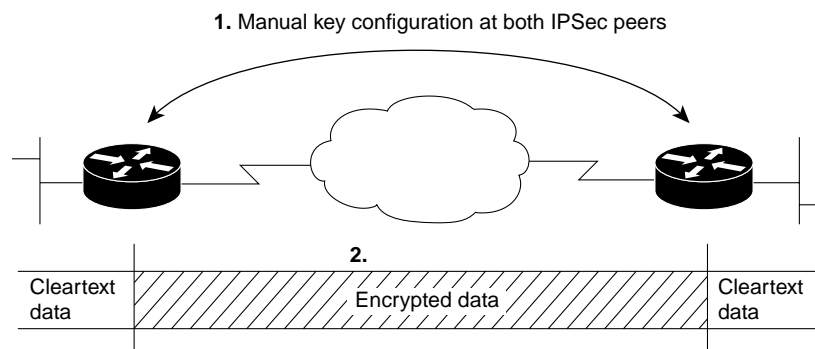
CAs simplify the administration of IPSec network devices. You can use a CA with a network containing multiple IPSec-compliant devices such as routers.

Implementing IPSec Without CAs

Without a CA, if you want to enable IPSec services (such as encryption) between two Cisco routers, you must first ensure that each router has the other router's key (such as an RSA public key or a shared key). This requires that you manually perform one of the following:

- At each router, enter the other router's RSA public key, or
- At each router, specify a shared key to be used between the routers.

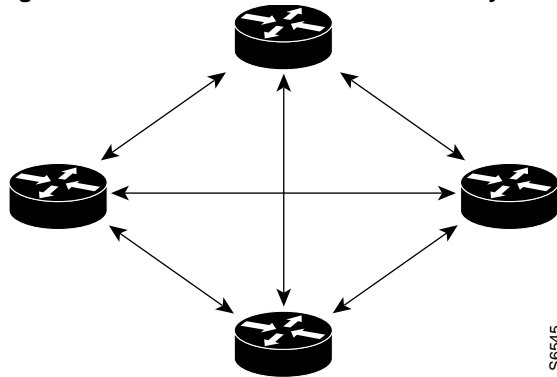
Figure 1 Without a CA: Key Configuration Between Two Routers



In the above scenario, each router uses the other router's key to authenticate the identity of the other router; this authentication always occurs whenever IPSec traffic is exchanged between the two routers.

If you have multiple Cisco routers in a mesh topology, and wish to exchange IPSec traffic passing between all of those routers, you must first configure shared keys or RSP public keys between all of those routers:

Figure 2 Without a CA: Six 2-Part Key Configurations Required for Four IPSec Routers



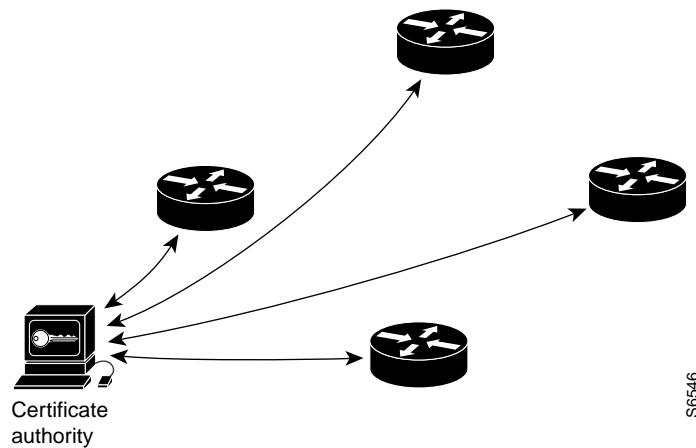
Every time a new router is added to the IPSec network, you must configure keys between the new router and each of the existing routers. (In Figure 2, four additional 2-part key configurations would be required to add a single encrypting router to the network).

Consequently, the more devices there are that require IPSec services, the more involved the key administration becomes. Obviously, this approach does not scale well for larger, more complex encrypting networks.

Implementing IPSec With CAs

With a CA, you do not need to configure keys between all of the encrypting routers. Instead, you individually enroll each participating router with the CA, requesting a certificate for the router. When this has been accomplished, each participating router can dynamically authenticate all of the other participating routers.

Figure 3 With a CA: Each Router Individually Makes Requests of the CA



To add a new IPSec router to the network, you only need to configure that new router to request a certificate from the CA, instead of making multiple key configurations with all the other existing IPSec routers.

How CA Certificates Are Used by IPSec Devices

When two IPSec routers want to exchange IPSec-protected traffic passing between them, they must first authenticate each other—otherwise, IPSec protection cannot occur.

Without a CA, a router authenticates itself to the remote router using either RSA encrypted nonces or pre-shared keys. Both methods require that keys must have been previously configured between the two routers.

With a CA, a router authenticates itself to the remote router by sending a certificate to the remote router and performing some public key cryptography. Each router must send its own unique certificate which was issued and validated by the CA. This process works because each router's certificate encapsulates the router's public key, each certificate is authenticated by the CA, and all participating routers recognize the CA as an authenticating authority.

Your router can continue sending its own certificate for multiple IPSec sessions, and to multiple IPSec peers, until the certificate expires. When its certificate expires, the router administrator must obtain a new one from the CA.

CAs can also revoke certificates for devices that will no longer participate in IPSec. Revoked certificates are not recognized as valid by other IPSec devices. Revoked certificates are listed in a Certificate Revocation List (CRL), which each peer may check before accepting another peer's certificate.

About Registration Authorities

Some CAs have a Registration Authority (RA) as part of their implementation. An RA is essentially a server that acts as a proxy for the CA so that CA functions can continue when the CA is offline.

Some of the configuration tasks described in this document differ slightly depending on whether your CA supports an RA or not.

Configuration Tasks

To enable your Cisco device to interoperate with a CA, complete the following configuration tasks:

- Manage NVRAM Memory Usage (Optional)
- Configure the Router's Hostname and IP Domain Name
- Generate an RSA Key Pair
- Declare a Certification Authority
- Authenticate the CA
- Request Your Own Certificate(s)
- Save Your Configuration
- Monitor and Maintain Certification Authority Interoperability

Manage NVRAM Memory Usage (Optional)

Certificates and Certificate Revocation Lists (CRLs) are used by your router when a CA is used. Normally certain of these certificates and all CRLs are stored locally in the router’s NVRAM, and each certificate and CRL uses a moderate amount of memory.

- What certificates are normally stored at your router?
 - Your router’s certificate
 - The CA’s certificate
 - Two Registration Authority (RA) certificates (only if the CA supports an RA)
- What CRLs are normally stored at your router?
 - If your CA does not support an RA, only one CRL gets stored at your router.
 - If your CA supports an RA, multiple CRLs can be stored at your router.

In some cases, storing these certificates and CRLs locally will not present a problem. However, in other cases, memory might become an issue—particularly if your CA supports an RA and a large number of CRLs end up being stored on your router.

To save NVRAM space, you can specify that certificates and CRLs should not be stored locally, but should be retrieved from the CA when needed. This will save NVRAM space but could result in a slight performance impact.

To specify that certificates and CRLs should not be stored locally on your router, but should be retrieved when required, turn on query mode by performing the following task in global configuration mode:

Task	Command
Turn on query mode, which causes certificates and CRLs to not be stored locally.	crypto ca certificate query

If you do not turn on query mode at this time, but later decide that you should, you can turn on query mode at that time even if certificates and CRLs have already been stored on your router. In this case, when you turn on query mode, the stored certificates and CRLs will be deleted from the router after you save your configuration. (If you copy your configuration to a TFTP site prior to turning on query mode, you will save any stored certificates and CRLs at the TFTP site.)

If you turn on query mode now, you can turn off query mode later if you wish. If you turn off query mode later, you could also perform the **copy running-config startup-config** command at that time to save all current certificates and CRLs to NVRAM (otherwise they could be lost during a reboot and would need to be retrieved the next time they were needed by your router).

Configure the Router’s Hostname and IP Domain Name

You must configure the router’s hostname and IP domain name if this has not already been done. This is required because the router assigns a fully qualified domain name (FQDN) to the keys and certificates used by IPSec, and the FQDN is based on the hostname and IP domain name you assign to the router. For example, a certificate is named “router20.companyx.com” based on a router hostname of “router20” and a router IP domain name of “companyx.com.”

To configure the router's hostname and IP domain name, complete the following tasks in global configuration mode:

Task	Command
Configure the router's hostname.	hostname <i>name</i>
Configure the router's IP domain name.	ip domain-name <i>name</i>

Generate an RSA Key Pair

RSA key pairs are used to sign and encrypt IKE key management messages and are required before you can obtain a certificate for your router.

To generate an RSA key pair, perform the following task in global configuration mode:

Task	Command
Generate an RSA key pair.	crypto key generate rsa [usage-keys]
Use the usage-keys keyword to specify special usage keys instead of general purpose keys. See the command description for an explanation of special usage vs. general purpose keys.	

Declare a Certification Authority

You should declare one Certification Authority (CA) to be used by your router.

To declare a CA, perform the following tasks starting in global configuration mode:

Task	Command
Declare a CA. (Create a name for the CA with this command.) This command puts you into the ca-identity configuration mode.	crypto ca identity <i>name</i>
Specify the URL of the CA. (The URL should include any non-standard cgi-bin script location.)	enrollment url <i>url</i>
If your CA system provides a Registration Authority (RA), specify RA mode.	enrollment mode ra
If your CA system provides an RA and supports the LDAP protocol, specify the location of the LDAP server.	query url <i>url</i>
(Optional) Specify a retry period. After requesting a certificate, the router waits to receive a certificate from the CA. If the router doesn't receive a certificate within a period of time (the retry period) the router will send another certificate request. You can change the retry period from the default of 1 minute.	enrollment retry-period <i>minutes</i>
(Optional) Specify how many times the router will continue to send unsuccessful certificate requests before giving up. By default, the router will never give up trying.	enrollment retry-count <i>count</i>

Task	Command
(Optional) Specify that other peers' certificates can still be accepted by your router even if the appropriate CRL is not accessible to your router.	cr1 optional
Exit ca-identity configuration mode.	exit

Authenticate the CA

The router needs to authenticate the CA. It does this by obtaining the CA's self-signed certificate which contains the CA's public key. Because the CA's certificate is self-signed (the CA signs its own certificate) the CA's public key should be manually authenticated by contacting the CA administrator when you perform this step.

To get the CA's public key, perform the following task in global configuration mode:

Task	Command
Get the CA's public key. Use the same <i>name</i> that you used when declaring the CA with the crypto ca identity command.	crypto ca authenticate name

Request Your Own Certificate(s)

You need to obtain a signed certificate from the CA for each of your router's RSA key pairs. If you generated general purpose RSA keys, your router only has one RSA key pair and needs only one certificate. If you previously generated special usage RSA keys, your router has two RSA key pairs and needs two certificates.

To request signed certificates from the CA, perform the following task in global configuration mode:

Task	Command
Request certificates for all of your RSA key pairs.	crypto ca enroll name
This command causes your router to request as many certificates as there are RSA key pairs, so you only need to perform this command once, even if you have special usage RSA key pairs.	
Note This command requires you to create a challenge password that is not saved with the configuration. This password is required in the event your certificate needs to be revoked, so remember this password.	

Note If your router reboots after you issued the **crypto ca enroll** command but before you received the certificate(s), you must reissue the command.

Save Your Configuration

Always remember to save your work when you make configuration changes.

Perform the **copy running-config startup-config** command to save your configuration—this command includes saving RSA keys to private NVRAM. RSA keys are *not* saved with your configuration when you perform a **copy running-config rcp** or **copy running-config tftp** (previously **write network**) command.

Monitor and Maintain Certification Authority Interoperability

The following tasks are optional, depending on your particular requirements:

- Request a Certificate Revocation List
- Delete Your Router's RSA Keys
- Delete Peer's Public Keys
- Delete Certificates from the Configuration
- View Keys and Certificates

Request a Certificate Revocation List

You can request a Certificate Revocation List (CRL) only if your CA does not support a Registration Authority (RA). The following description and task applies only when the CA does not support an RA.

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

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

If your router has a CRL which has not yet expired, but you suspect that the CRL's contents are out of date, you can request that the latest CRL be immediately downloaded to replace the old CRL.

To request immediate download of the latest CRL, perform the following task in global configuration mode:

Task	Command
Request an updated CRL.	crypto ca crl request <i>name</i>
This command replaces the currently stored CRL at your router with the newest version of the CRL.	

Delete Your Router's RSA Keys

There might be circumstances where you would want to delete your router's RSA keys. For example, if you believe the RSA keys were compromised in some way and should no longer be used, you should delete the keys.

To delete all of your router’s RSA keys, perform the following task in global configuration mode:

Task	Command
Delete all of your router’s RSA keys.	crypto key zeroize rsa

After you delete a router’s RSA keys, you should also do these two additional tasks:

- Ask the CA administrator to revoke your router’s certificates at the CA; you must supply the challenge password you created when you originally obtained the router’s certificates with the **crypto ca enroll** command.
- Manually remove the router’s certificates from the router configuration as described in the section, “Delete Certificates from the Configuration.”

Delete Peer’s Public Keys

There might be circumstances where you would want to delete other peer’s RSA public keys from your router’s configuration. For example, if you no longer trust the integrity of a peer’s public key, you should delete the key.

To delete a peer’s RSA public key, perform the following tasks starting in global configuration mode:

Task	Command
Enter public key configuration mode.	crypto key pubkey-chain rsa
Delete a remote peer’s RSA public key. Specify the peer’s fully qualified domain name or the remote peer’s IP address.	no named-key <i>key-name</i> [encryption signature] or no addressed-key <i>key-address</i> [encryption signature]
Return to global configuration mode.	exit

These commands are documented in the “Internet Key Exchange” feature documentation.

Delete Certificates from the Configuration

If the need arises, you can delete certificates that are saved at your router. Your router saves its own certificate(s), the CA’s certificate, and any RA certificates (unless you put the router into query mode per the “Manage NVRAM Memory Usage (Optional)” section).

To delete your router’s certificate or RA certificates from your router’s configuration, perform the following tasks in global configuration mode:

Task	Command
View the certificates stored on your router; note (or copy) the serial number of the certificate you wish to delete.	show crypto ca certificates
Enter certificate chain configuration mode.	crypto ca certificate chain <i>name</i>
Delete the certificate.	no certificate <i>certificate-serial-number</i>

To delete the CA's certificate, you must remove the entire CA identity, which also removes all certificates associated with the CA—your router's certificate, the CA certificate, and any RA certificates.

To remove a CA identity, perform the following task in global configuration mode:

Task	Command
Delete all identity information and certificates associated with the CA.	no crypto ca identity <i>name</i>

View Keys and Certificates

To view keys and certificates, perform the following tasks in EXEC mode:

Task	Command
View your router's RSA public keys.	show crypto key mypubkey rsa
View a list of all the RSA public keys stored on your router. These include the public keys of peers who have sent your router their certificates during peer authentication for IPSec.	show crypto key pubkey-chain rsa
View details of a particular RSA public key stored on your router.	show crypto key pubkey-chain rsa [name <i>key-name</i> address <i>key-address</i>]
View information about your certificate, the CA's certificate, and any RA certificates.	show crypto ca certificates

Configuration Examples

The following configuration is for a router named "myrouter." In this example IPSec is configured and the IKE protocol and CA interoperability are configured in support of IPSec.

In this example general purpose RSA keys were generated, but you will notice that the keys are not saved or displayed in the configuration.

Comments are included within the configuration to explain various commands.

```

!
version 11.3
no service password-encryption
service udp-small-servers
service tcp-small-servers
!
! CA interoperability requires you to configure your router's hostname:
hostname myrouter
!
enable secret 5 <removed>
enable password <removed>
!
! CA interoperability requires you to configure your router's IP domain name:
ip domain-name companyx.com
ip name-server 172.29.2.132
ip name-server 192.168.30.32
!

```

Configuration Examples

```
! The following configures a transform set (part of IPSec configuration):
crypto ipsec transform-set my-transformset esp-des esp-sha-hmac
!
! The following declares the CA. (In this example, the CA does not support an RA.)
crypto ca identity myca
  enrollment url http://ca_server
!
! The following shows the certificates and CRLs stored at the router, including
!   the CA certificate (shown first), the router's certificate (shown next)
!   and a CRL (shown last).
crypto ca certificate chain myca
! The following is the CA certificate
!   received via the 'crypto ca authenticate' command:
certificate ca 3051DF7169BEE31B821DFE4B3A338E5F
  30820182 3082012C A0030201 02021030 51DF7169 BEE31B82 1DFE4B3A 338E5F30
  0D06092A 864886F7 0D010104 05003042 31163014 06035504 0A130D43 6973636F
  20537973 74656D73 3110300E 06035504 0B130744 65767465 73743116 30140603
  55040313 0D434953 434F4341 2D554C54 5241301E 170D3937 31323032 30313036
  32385A17 0D393831 32303230 31303632 385A3042 31163014 06035504 0A130D43
  6973636F 20537973 74656D73 3110300E 06035504 0B130744 65767465 73743116
  30140603 55040313 0D434953 434F4341 2D554C54 5241305C 300D0609 2A864886
  F70D0101 01050003 4B003048 024100C1 B69D7BF6 34E4EE28 A84E0DC6 FCA4DEA8
  04D89E50 C5EBE862 39D51890 D0D4B732 678BDBF2 80801430 E5E56E7C C126E2DD
  DBE9695A DF8E5BA7 E67BAE87 29375302 03010001 300D0609 2A864886 F70D0101
  04050003 410035AA 82B5A406 32489413 A7FF9A9A E349E5B4 74615E05 058BA3CE
  7C5F00B4 019552A5 E892D2A3 86763A1F 2852297F C68EECE1 F41E9A7B 2F38D02A
  B1D2F817 3F7B
quit
! The following is the router's certificate
!   received via the 'crypto ca enroll' command:
certificate 7D28D4659D22C49134B3D1A0C2C9C8FC
  308201A6 30820150 A0030201 0202107D 28D4659D 22C49134 B3D1A0C2 C9C8FC30
  0D06092A 864886F7 0D010104 05003042 31163014 06035504 0A130D43 6973636F
  20537973 74656D73 3110300E 06035504 0B130744 65767465 73743116 30140603
  55040313 0D434953 434F4341 2D554C54 5241301E 170D3938 30343234 30303030
  30305A17 0D393930 34323432 33353935 395A302F 311D301B 06092A86 4886F70D
  01090216 0E73636F 742E6369 73636F2E 636F6D31 0E300C06 03550405 13053137
  41464230 5C300D06 092A8648 86F70D01 01010500 034B0030 48024100 A207ED75
  DE8A9BC4 980958B7 28ADF562 1371D043 1FC93C24 8E9F8384 4D1A2407 60CBD7EC
  B15BD782 A687CA49 883369BE B35A4219 8FE742B0 91CF76EE 07EC9E69 02030100
  01A33530 33300B06 03551D0F 04040302 05A03019 0603551D 11041230 10820E73
  636F742E 63697363 6F2E636F 6D300906 03551D13 04023000 300D0609 2A864886
  F70D0101 04050003 410085F8 A5AFA907 B38731A5 0195D921 D8C45EFD B6082C28
  04A88CEC E9EC6927 F24874E4 30C4D7E2 2686E0B5 77F197E4 F82A8BA2 1E03944D
  286B661F 0305DF5F 3CE7
quit
! The following is a CRL received by the router (via the router's own action):
crl
  3081C530 71300D06 092A8648 86F70D01 01020500 30423116 30140603 55040A13
  0D436973 636F2053 79737465 6D733110 300E0603 55040B13 07446576 74657374
  31163014 06035504 03130D43 4953434F 43412D55 4C545241 170D3938 30333233
  32333232 31305A17 0D393930 34323230 30303030 305A300D 06092A86 4886F70D
  01010205 00034100 7AA83057 AC5E5C65 B9812549 37F11B7B 5CA4CAED 830B3955
  A4DDD268 F567E29A E4B34691 C2162BD1 0540D7E6 5D6650D1 81DDBF1D 788F1DAC
  BBF761B2 81FCC0F1
quit
!
! The following is an IPSec crypto map (part of IPSec configuration):
crypto map map-to-remotesite 10 ipsec-isakmp
  set peer 172.21.114.196
  set transform-set my-transformset
  match address 124
!
!
```

```
interface Loopback0
  ip address 10.0.0.1 255.0.0.0
!
interface Tunnel0
  ip address 10.0.0.2 255.0.0.0
  ip mtu 1490
  no ip route-cache
  no ip mroute-cache
  tunnel source 10.10.0.1
  tunnel destination 172.21.115.119
!
interface FastEthernet0/0
  ip address 172.21.115.118 255.255.255.240
  no ip mroute-cache
  loopback
  no keepalive
  shutdown
  media-type MII
  full-duplex
!
! The IPsec crypto map is applied to interface Ethernet1/0:
interface Ethernet1/0
  ip address 172.21.114.197 255.255.255.0
  bandwidth 128
  no keepalive
  no fair-queue
  no cdp enable
  crypto map map-to-remotesite
!
interface Ethernet1/1
  no ip address
  no ip mroute-cache
  shutdown
!
interface Ethernet1/2
  no ip address
  shutdown
!
interface Ethernet1/3
  no ip address
  shutdown
!
interface Serial3/0
  no ip address
  shutdown
!
interface Serial3/1
  no ip address
  shutdown
!
interface Serial3/2
  no ip address
  shutdown
!
interface Serial3/3
  no ip address
  shutdown
!
interface Serial4/0
  no ip address
  shutdown
!
interface Serial4/1
  ip address 172.21.115.66 255.255.255.252
!
```

Configuration Examples

```
interface Serial4/2
  no ip address
  shutdown
!
interface Serial4/3
  no ip address
  shutdown
  clockrate 2015232
!
router eigrp 10
  passive-interface Ethernet1/0
  network 172.21.0.0
  no auto-summary
!
no ip classless
ip route 10.0.0.1 255.255.255.255 Serial4/1
ip route 10.60.0.0 255.0.0.0 10.10.0.2
ip route 172.69.0.0 255.255.0.0 172.21.114.193
ip route 172.21.114.3 255.255.255.255 10.1.1.1
ip route 172.21.114.14 255.255.255.255 10.1.1.1
ip route 172.21.114.69 255.255.255.255 Tunnel0
ip route 172.21.114.89 255.255.255.255 10.10.0.2
ip route 172.21.114.92 255.255.255.255 Tunnel0
ip route 172.21.114.165 255.255.255.255 10.10.0.2
ip route 172.21.114.170 255.255.255.255 10.10.0.2
ip route 172.21.115.119 255.255.255.255 10.10.0.2
access-list 100 permit ip host 172.21.114.197 host 172.21.114.204
access-list 101 permit ip host 11.0.0.1 host 10.0.0.1
access-list 102 permit ip 172.21.114.0 0.0.0.255 host 172.21.114.3
access-list 102 permit ip 70.1.1.0 0.0.0.255 host 172.21.114.3
access-list 102 permit ip 15.0.0.0 0.0.0.255 host 172.21.114.3
access-list 104 permit ip host 172.21.114.197 host 172.21.114.14
access-list 110 permit gre host 172.21.114.197 host 172.21.114.199
access-list 114 permit ip host 172.21.114.199 host 172.21.114.163
access-list 114 permit ip host 172.21.115.66 host 172.21.115.65
access-list 122 permit ip host 172.21.115.66 host 172.21.115.65
!
! Access list 124 is the IPsec access list included in the IPsec crypto map.
! This access list causes all traffic between hosts 172.21.114.197 and 172.21.114.196
! to be protected by IPsec.
access-list 124 permit ip host 172.21.114.197 host 172.21.114.196
access-list 132 permit ip host 172.21.114.197 host 172.21.114.37
access-list 141 permit ip host 172.21.114.197 host 172.21.114.196
access-list 150 permit gre host 15.0.0.1 host 172.21.115.119
access-list 180 permit ip host 70.1.1.1 host 60.1.1.1
access-list 186 permit ip host 172.21.115.66 host 172.21.114.9
!
line con 0
  exec-timeout 0 0
line aux 0
line vty 0 4
  password mypassword
  login
!
end
```

Command Reference

This section documents new or modified commands. All other commands used with this feature are documented in the Cisco IOS Release 11.3 command references.

- **certificate**
- **crl optional**
- **crypto ca authenticate**
- **crypto ca certificate chain**
- **crypto ca certificate query**
- **crypto ca crl request**
- **crypto ca enroll**
- **crypto ca identity**
- **crypto key generate rsa**
- **crypto key zeroize rsa**
- **enrollment mode ra**
- **enrollment retry-count**
- **enrollment retry-period**
- **enrollment url**
- **query url**
- **show crypto ca certificates**
- **show crypto key mypubkey rsa**
- **show crypto key pubkey-chain rsa**

certificate

To manually add certificates, use the **certificate** certificate chain configuration command. Use the **no** form of this command to delete your router's certificate or any RA certificates stored on your router.

```
certificate certificate-serial-number  
no certificate certificate-serial-number
```

Syntax Description

certificate-serial-number Specify the serial number of the CA to add or delete.

Default

There are no defaults for this command.

Command Mode

Certificate chain configuration (config-cert-chain)

Usage Guidelines

This command first appeared in Cisco IOS Release 11.3 T.

You could use this command to manually specify a certificate. However, this command is rarely used in this manner. Instead, this command is usually only used to delete certificates.

Example

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

```
myrouter# show crypto ca certificates  
  
Certificate  
  Subject Name  
    Name: myrouter.companyx.com  
    IP Address: 10.0.0.1  
  Status: Available  
  Certificate Serial Number: 0123456789ABCDEF0123456789ABCDEF  
  Key Usage: General Purpose  
  
CA Certificate  
  Status: Available  
  Certificate Serial Number: 3051DF7123BEE31B8341DFE4B3A338E5F  
  Key Usage: Not Set  
  
myrouter# configure terminal  
myrouter(config)# crypto ca certificate chain myca  
myrouter(config-cert-chain)# no certificate 0123456789ABCDEF0123456789ABCDEF  
% Are you sure you want to remove the certificate [yes/no]? yes  
% Be sure to ask the CA administrator to revoke this certificate.  
myrouter(config-cert-chain)# exit  
myrouter(config)#
```

Related Commands

crypto ca certificate chain

crl optional

To allow other peers' certificates to still be accepted by your router even if the appropriate Certificate Revocation List (CRL) is not accessible to your router, use the **crl optional** ca-identity configuration command. Use the **no** form of the command to return to the default behavior in which CRL checking is mandatory before your router can accept a certificate.

crl optional
no crl optional

Syntax Description

There are no arguments or keywords with this command.

Default

The router must have and check the appropriate CRL before accepting another IPSec peer's certificate.

Command Mode

Ca-identity configuration

Usage Guidelines

This command first appeared in Cisco IOS Release 11.3 T.

When your router receives a certificate from a peer, your router will download a Certificate Revocation List (CRL) from either the CA or a CRL distribution point as designated in the peer's certificate. Your router then checks the CRL to make sure the certificate the peer sent has not been revoked. (If the certificate appears on the CRL, your router will not accept the certificate and will not authenticate the peer.)

With CA systems that support Registration Authorities (RAs), multiple CRLs exist and the peer's certificate will indicate which CRL applies and should be downloaded by your router.

When an IPSec peer sends your router a certificate, if your router does not have the applicable CRL and is unable to obtain the applicable CRL, your router will reject the peer's certificate—unless you include the **crl optional** command in your configuration. If you use the **crl optional** command, your router will still try to obtain a CRL, but if it cannot obtain a CRL it can accept the peer's certificate anyway.

When your router receives additional certificates from peers, your router will continue to attempt to download the appropriate CRL, even if it was previously unsuccessful, and even if the **crl optional** command is enabled. The **crl optional** command only specifies that when the router cannot obtain the CRL, the router is not forced to reject a peer's certificate outright.

Example

The following example declares a CA and permits your router to accept certificates when CRLs are not obtainable. This example also specifies a non-standard retry period and retry count.

```
crypto ca identity myca
  enrollment url http://ca_server
  enrollment retry-period 20
  enrollment retry-count 100
  crl optional
```

Related Commands

crypto ca identity

crypto ca authenticate

To authenticate the CA (by getting the CA's certificate), use the **crypto ca authenticate** global configuration command.

crypto ca authenticate *name*

Syntax Description

name Specify the name of the CA. Use the same name as when you declared the CA using the **crypto ca identity** command.

Default

There are no defaults for this command.

Command Mode

Global configuration

Usage Guidelines

This command first appeared in Cisco IOS Release 11.3 T.

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

This command authenticates the CA to your router by obtaining the CA's self-signed certificate which contains the CA's public key. Because the CA's certificate is self-signed (the CA signs its own certificate) you should manually authenticate the CA's public key by contacting the CA administrator when you perform this command.

If you are using RA mode (using the **enrollment mode ra** command) when you issue the **crypto ca authenticate** command, then RA signing and encryption certificates will be returned from the CA as well as the CA certificate.

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

If the CA does not respond by a timeout period after this command is issued, the terminal control will simply be returned so it will not be tied up. If this happens you must re-enter the command.

Example

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

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

Related Commands

crypto ca identity

show crypto ca certificates

crypto ca certificate chain

To enter the certificate chain configuration mode, use the **crypto ca certificate chain** global configuration command. (You need to be in certificate chain configuration mode to delete certificates.)

```
crypto ca certificate chain name
```

Syntax Description

name Specify the name of the CA. Use the same name as when you declared the CA using the **crypto ca identity** command.

Default

There are no defaults for this command.

Command Mode

Global configuration.

Usage Guidelines

This command first appeared in Cisco IOS Release 11.3 T.

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

Example

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

```
myrouter# show crypto ca certificates

Certificate
  Subject Name
    Name: myrouter.companyx.com
    IP Address: 10.0.0.1
  Status: Available
  Certificate Serial Number: 0123456789ABCDEF0123456789ABCDEF
  Key Usage: General Purpose

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

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

Related Commands
certificate

crypto ca certificate query

To specify that certificates and Certificate Revocation Lists (CRLs) should not be stored locally but retrieved from the CA when needed, use the **crypto ca certificate query** global configuration command. This command puts the router into query mode. Use the **no** form of this command to cause certificates and CRLs to be stored locally (the default).

crypto ca certificate query
no crypto ca certificate query

Syntax Description

This command has no arguments or keywords.

Default

Certificates and CRLs are stored locally in the router's NVRAM.

Command Mode

Global configuration

Usage Guidelines

This command first appeared in Cisco IOS Release 11.3 T.

Normally, certain certificates and Certificate Revocation Lists (CRLs) are stored locally in the router's NVRAM, and each certificate and CRL uses a moderate amount of memory.

To save NVRAM space, you can use this command to put the router into query mode, which prevents certificates and CRLs from being stored locally; instead, they are retrieved from the CA when needed. This will save NVRAM space but could result in a slight performance impact.

Examples

This example prevents certificates and CRLs from being stored locally on the router; instead, they are retrieved from the CA when needed.

```
crypto ca certificate query
```

crypto ca crl request

To request that a new Certificate Revocation List (CRL) be obtained immediately from the CA, use the **crypto ca crl request** global configuration command. Use this command only when your CA does not support a Registration Authority (RA).

crypto ca crl request *name*

Syntax Description.

name Specify the name of the CA. Use the same name as when you declared the CA using the **crypto ca identity** command.

Default

Normally, the router requests a new CRL only after the existing one expires.

Command Mode

Global configuration

Usage Guidelines

This command first appeared in Cisco IOS Release 11.3 T.

Use this command only if your CA does not support a Registration Authority (RA).

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

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

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

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

This command is never saved to the configuration.

Example

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

```
crypto ca crl request
```

crypto ca enroll

To obtain your router's certificate(s) from the CA, use the **crypto ca enroll** global configuration command. Use the **no** form of this command to delete a current enrollment request.

crypto ca enroll *name*
no crypto ca enroll *name*

Syntax Description

name Specify the name of the CA. Use the same name as when you declared the CA using the **crypto ca identity** command.

Default

There are no defaults for this command.

Command Mode

Global configuration

Usage Guidelines

This command first appeared in Cisco IOS Release 11.3 T.

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

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

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

The **crypto ca enroll** command is never saved in the router configuration.

Note If your router reboots after you issued the **crypto ca enroll** command but before you received the certificate(s), you must reissue the command.

Responding to Prompts

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

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

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

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

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

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

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

Example

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

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

```
myrouter(config)# crypto ca enroll myca
%
% Start certificate enrollment ..
% Create a challenge password. You will need to verbally provide this
password to the CA Administrator in order to revoke your certificate.
For security reasons your password will not be saved in the configuration.
Please make a note of it.

Password: <mypassword>
Re-enter password: <mypassword>
```

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

```
myrouter(config)#
```

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

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

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

myrouter(config)#
```

If necessary, the router administrator could verify the displayed Fingerprint with the CA administrator.

If there had been a problem with the certificate request and the certificate was not granted, the following message would have been displayed on the console instead:

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

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

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

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

Related Commands

show crypto ca certificates

crypto ca identity

To declare the CA your router should use, use the **crypto ca identity** global configuration command. Use the **no** form of this command to delete all identity information and certificates associated with the CA.

```
crypto ca identity name
no crypto ca identity name
```

Syntax Description

name Create a name for the CA. (If you previously declared the CA and just want to update its characteristics, specify the name you previously created.)

Default

Your router does not know about any CA until you declare one with this command.

Command Mode

Global configuration

Usage Guidelines

This command first appeared in Cisco IOS Release 11.3 T.

Use this command to declare a CA. Performing this command puts you into the ca-identity configuration mode, where you can specify characteristics for the CA with the following commands:

- **enrollment url** (Specify the URL of the CA—always required.)
- **enrollment mode ra** (Specify RA mode, required only if your CA system provides a Registration Authority [RA]).
- **query url** (Specify the URL of the LDAP server, required only if your CA supports an RA and the LDAP protocol.)
- **enrollment retry-period** (Specify a period of time the router should wait between sending certificate request retries—optional.)
- **enrollment retry-count** (Specify how many certificate request retries your router will send before giving up—optional.)
- **crl optional** (Specify that your router can still accept other peers' certificates if the CRL is not accessible—optional.)

Examples

The following example declares a CA and identifies characteristics of the CA. In this example, the name “myca” is created for the CA, which is located at `http://ca_server`.

The CA does not use an RA or LDAP, and the CA's scripts are stored in the default location. This is the minimum possible configuration required to declare a CA.

```
crypto ca identity myca
  enrollment url http://ca_server
```

The following example declares a CA when the CA uses an RA. The CA's scripts are stored in the default location, and the CA uses the certificate enrollment protocol (CEP) instead of LDAP. This is the minimum possible configuration required to declare a CA that uses an RA.

```
crypto ca identity myca_with_ra
  enrollment url http://companyx_ca
  enrollment mode ra
  query url ldap://serverx
```

The following example declares a CA that uses an RA, and that uses a non-standard cgi-bin script location. This example also specifies a non-standard retry period and retry count, and permits your router to accept certificates when CRLs are not obtainable.

```
crypto ca identity myca_with_ra
  enrollment url http://companyx_ca/cgi-bin/somewhere/scripts.exe
  enrollment mode ra
  query url ldap://serverx
  enrollment retry-period 20
  enrollment retry-count 100
  crl optional
```

In the previous example, if the router does not receive a certificate back from the CA within 20 minutes of sending a certificate request, the router will resend the certificate request. The router will keep sending a certificate request every 20 minutes until a certificate is received or until 100 requests have been sent.

If the CA cgi-bin script location is not /cgi-bin/pkiclient.exe at the CA (the default CA cgi-bin script location) you need to also include the non-standard script location in the URL, in the form of http://CA_name/script_location where script_location is the full path to the CA scripts.

Related Commands

- enrollment url**
- enrollment mode ra**
- query url**
- enrollment retry-period**
- enrollment retry-count**
- crl optional**

crypto key generate rsa

To generate RSA key pairs, use the **crypto key generate rsa** global configuration command.

```
crypto key generate rsa [usage-keys]
```

Syntax Description

usage-keys (Optional) Specifies that two special-usage key pairs should be generated, instead of one general-purpose key pair.

Default

RSA key pairs do not exist. If the **usage-keys** keyword is not used, general-purpose keys will be generated.

Command Mode

Global configuration

Usage Guidelines

This command first appeared in Cisco IOS Release 11.3 T.

Use this command to generate RSA key pairs for your Cisco device (such as a router).

RSA keys are generated in pairs—one public RSA key and one private RSA key.

If your router already has RSA keys when you issue this command, you will be warned and prompted to replace the existing keys with new keys.

Note Before issuing this command, make sure your router has a hostname and IP domain name configured (with the **hostname** and **ip domain-name** commands). You will be unable to complete the **crypto key generate rsa** command without a hostname and IP domain name.

This command is never saved in the router configuration; however, the keys generated by this command are saved in the private configuration in NVRAM (which is never displayed to the user or backed up to another device).

There are two mutually-exclusive styles of RSA key pairs: special usage keys and general purpose keys. When you generate RSA key pairs, you will be prompted to select whether to generate special usage keys or general purpose keys.

Special Usage Keys

If you generate special usage keys, two pairs of RSA keys will be generated. One pair will be used with any IKE policy that specifies RSA signatures as the authentication method, and the other pair used with any IKE policy that specifies RSA encrypted nonces as the authentication method. (You configure RSA signatures or RSA encrypted nonces in your IKE policies as described in the “Internet Key Exchange Security Policy” feature documentation.)

A CA is used only with IKE policies specifying RSA signatures, not with IKE policies specifying RSA encrypted nonces. (However, you could specify more than one IKE policy, and have RSA signatures specified in one policy and RSA encrypted nonces in another policy.)

If you plan to have both types of RSA authentication methods in your IKE policies, you might prefer to generate special usage keys. With special usage keys, each key is not unnecessarily exposed. (Without special usage keys, one key is used for both purposes, increasing that key's exposure.)

General Purpose Keys

If you generate general purpose keys, only one pair of RSA keys will be generated. This pair will be used with IKE policies specifying either RSA signatures or RSA encrypted nonces. Therefore, a general purpose key pair might get used more frequently than a special usage key pair.

Modulus Length

When you generate RSA keys, you will be prompted to enter a modulus length. A longer modulus could offer stronger security, but takes longer to generate (see Table 1 for sample times) and takes longer to use. Below 512 is normally not recommended. (In certain situations, the shorter modulus may not function properly with IKE, so Cisco recommends using a minimum modulus of 1024.)

Table 1 Sample Times Required to Generate RSA Keys

Router	Modulus Length			
	360 bits	512 bits	1024 bits	2048 bits
Cisco 2500	11 seconds	20 seconds	4 minutes, 38 seconds	longer than 1 hour
Cisco 4700	less than 1 second	1 second	4 seconds	50 seconds

Examples

This example generates special usage RSA keys.

```
myrouter(config)# crypto key generate rsa usage-keys
The name for the keys will be: myrouter.companyx.com
```

```
Choose the size of the key modulus in the range of 360 to 2048 for your Signature Keys.
Choosing a key modulus greater than 512 may take a few minutes.
How many bits in the modulus[512]? <return>
Generating RSA keys.... [OK].
```

```
Choose the size of the key modulus in the range of 360 to 2048 for your Encryption
Keys. Choosing a key modulus greater than 512 may take a few minutes.
How many bits in the modulus[512]? <return>
Generating RSA keys.... [OK].
```

```
myrouter(config)#
```

This example generates general purpose RSA keys. (Note, you cannot generate both special usage and general purpose keys; you can only generate one or the other.)

```
myrouter(config)# crypto key generate rsa
The name for the keys will be: myrouter.companyx.com

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]? <return>
Generating RSA keys.... [OK].

myrouter(config)#
```

Related Commands

show crypto key mypubkey rsa

crypto key zeroize rsa

To delete all of your router's RSA keys, use the **crypto key zeroize rsa** command.

crypto key zeroize rsa

Syntax Description

There are no arguments or keywords for this command.

Default

There are no defaults for this command.

Command Mode

Global configuration.

Usage Guidelines

This command first appeared in Cisco IOS Release 11.3 T.

This command deletes all RSA keys that were previously generated by your router. If you issue this command, you must also do these two additional tasks:

- Ask the CA administrator to revoke your router's certificates at the CA; you must supply the challenge password you created when you originally obtained the router's certificates with the **crypto ca enroll** command.
- Manually remove the router's certificates from the configuration using the **certificate** command.

Note This command cannot be undone (after you save your configuration), and after RSA keys have been deleted you cannot use certificates or the CA or participate in certificate exchanges with other IPSec peers unless you reconfigure CA interoperability by regenerating RSA keys, getting the CA's certificate, and requesting your own certificate again.

This command is not saved to the configuration.

Example

This example deletes the general purpose RSA key pair that was previously generated for the router. After deleting the RSA key pair, the administrator contacts the CA administrator and requests that the router's certificate be revoked. The administrator then deletes the router's certificate from the configuration.

```
crypto key zeroize rsa
crypto ca certificate chain
no certificate
```

Related Commands

crypto ca certificate chain
certificate

enrollment mode ra

To turn on RA mode, use the **enrollment mode ra** ca-identity configuration command. Use the **no** form of the command to turn off RA mode.

enrollment mode ra
no enrollment mode ra

Syntax Description

This command has no arguments or keywords.

Default

RA mode is turned off.

Command Mode

Ca-identity configuration

Usage Guidelines

This command first appeared in Cisco IOS Release 11.3 T.

This command is required if your CA system provides a Registration Authority (RA). This command provides compatibility with RA systems.

Example

The following is an example of the minimum configuration required to declare a CA when the CA provides an RA.

```
crypto ca identity myca
  enrollment url http://ca_server
  enrollment mode ra
  query url ldap://serverx
```

Related Commands

crypto ca identity

enrollment retry-count

To specify how many times a router will resend a certificate request, use the **enrollment retry-count** ca-identity configuration command. Use the **no** form of the command to reset the retry count to the default of 0 which indicates an infinite number of retries.

enrollment retry-count *number*
no enrollment retry-count

Syntax Description.

number Specify how many times the router will resend a certificate request when the router does not receive a certificate from the CA from the previous request.
Specify from 1 to 100 retries.

Default

The router will send the CA another certificate request until a valid certificate is received (no limit to the number of retries).

Command Mode

Ca-identity configuration

Usage Guidelines

This command first appeared in Cisco IOS Release 11.3 T.

After requesting a certificate, the router waits to receive a certificate from the CA. If the router does not receive a certificate within a period of time (the retry period) the router will send another certificate request. The router will continue to send requests until it receives a valid certificate, until the CA returns an enrollment error, or until the configured number of retries (the retry count) is exceeded. By default, the router will keep sending requests forever, but you can change this to a finite number of permitted retries with this command.

A retry count of 0 indicates that there is no limit to the number of times the router should resend the certificate request. By default, the retry count is 0.

Examples

This example declares a CA, changes the retry period to 10 minutes, and changes the retry count to 60 retries. The router will resend the certificate request every 10 minutes until the router receives the certificate or until approximately 10 hours pass since the original request was sent, whichever occurs first. (10 minutes x 60 tries = 600 minutes = 10 hours.)

```
crypto ca identity myca
enrollment url http://ca_server
enrollment retry-period 10
enrollment retry-count 60
```

Related Commands
crypto ca identity
enrollment retry-period

enrollment retry-period

To specify the wait period between certificate request retries, use the **enrollment retry-period** *ca-identity* configuration command. Use the **no** form of the command to reset the retry period to the default of 1 minute.

enrollment retry-period *minutes*
no enrollment retry-period

Syntax Description

minutes Specify the number of minutes the router waits before resending a certificate request to the CA, when the router does not receive a certificate from the CA by the previous request.

Specify from 1 to 60 minutes. By default, the router retries every 1 minute.

Default

The router will send the CA another certificate request every 1 minute until a valid certificate is received.

Command Mode

Ca-identity configuration

Usage Guidelines

This command first appeared in Cisco IOS Release 11.3 T.

After requesting a certificate, the router waits to receive a certificate from the CA. If the router does not receive a certificate within a period of time (the retry period) the router will send another certificate request. The router will continue to send requests until it receives a valid certificate, until the CA returns an enrollment error, or until the configured number of retries is exceeded. (By default, the router will keep sending requests forever, but you can change this to a finite number of permitted retries with the **enrollment retry-count** command.)

Use the **enrollment retry-period** command to change the retry period from the default of 1 minute between retries.

Examples

This example declares a CA and changes the retry period to 5 minutes.

```
crypto ca identity myca
enrollment url http://ca_server
enrollment retry-period 5
```

Related Commands

crypto ca identity
enrollment retry-count

enrollment url

To specify the CA location by naming the CA's URL, use the **enrollment url** ca-identity configuration command. The **no** form of this command removes the CA's URL from the configuration.

```
enrollment url url  
no enrollment url url
```

Syntax Description

url Specify the URL of the CA where your router should send certificate requests, for example, `http://ca_server`.

This URL must be in the form of `http://CA_name` where `CA_name` is the CA's host DNS name or IP address.

If the CA cgi-bin script location is not `/cgi-bin/pkiclient.exe` at the CA (the default CA cgi-bin script location) you need to also include the non-standard script location in the URL, in the form of `http://CA_name/script_location` where `script_location` is the full path to the CA scripts.

Default

Your router does not know the CA URL until you specify it with this command.

Command Mode

Ca-identity configuration

Usage Guidelines

This command first appeared in Cisco IOS Release 11.3 T.

Use this command to specify the CA's URL. This command is required when you declare a CA with the **crypto ca identity** command.

The URL must include the CA script location if the CA scripts are not loaded into the default cgi-script location. The CA administrator should be able to tell you where the CA scripts are located.

To change a CA's URL, repeat the **enrollment url** command to overwrite the older URL.

Example

The following is an example of the absolute minimum configuration required to declare a CA.

```
crypto ca identity myca  
  enrollment url http://ca_server
```

Related Commands

crypto ca identity

query url

To specify LDAP protocol support, use the **query url** ca-identity configuration command. The **no** form of this command removes the query URL from the configuration and specifies the default query protocol, certificate enrollment protocol (CEP).

```
query url url  
no query url url
```

Syntax Description

url Specify the URL of the LDAP server; for example, ldap://another_server.
This URL must be in the form of ldap://server_name where server_name is the host DNS name or IP address of the LDAP server.

Default

The router uses CEP.

Command Mode

Ca-identity configuration

Usage Guidelines

This command first appeared in Cisco IOS Release 11.3 T.

This command is required if the CA supports a Registration Authority (RA) and the LDAP protocol; LDAP is a query protocol used when the router retrieves certificates and CRLs. The CA administrator should be able to tell you whether the CA supports LDAP or CEP; if the CA supports the LDAP protocol, the CA administrator can tell you the LDAP location from where certificates and CRLs should be retrieved.

To change the query URL, repeat the **query url** command to overwrite the older URL.

This command is only valid if you also use the **enrollment mode ra** command.

Example

The following is an example of a configuration required to declare a CA when the CA supports LDAP.

```
crypto ca identity myca  
  enrollment url http://ca_server  
  enrollment mode ra  
  query url ldap://bobs_server
```

Related Commands

crypto ca identity

show crypto ca certificates

To view information about your certificate, the CA's certificate, and any RA certificates, use the **show crypto ca certificates EXEC** command.

show crypto ca certificates

Syntax Description

This command has no arguments or keywords.

Command Mode

EXEC

Usage Guidelines

This command first appeared in Cisco IOS Release 11.3T.

This command shows information about the following certificates:

- Your certificate, if you have requested one from the CA (see the **crypto ca enroll** command)
- The CA's certificate, if you have received the CA's certificate (see the **crypto ca authenticate** command)
- RA certificates, if you have received RA certificates (see the **crypto ca authenticate** command)

Sample Display

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

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

The CA certificate might show Key Usage as "Not Set."

The following is sample output from the **show crypto ca certificates** command, and shows the router's certificate and the CA's certificate. In this example, a single, general purpose RSA key pair had been previously generated, and a certificate was requested but not yet received for that key pair:

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

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

Note that in the previous sample, the router's certificate Status shows "Pending." After the router receives its certificate from the CA, the Status field changes to "Available" in the **show** output.

The following is sample output from the **show crypto ca certificates** command, and shows two router's certificates and the CA's certificate. In this example, special usage RSA key pairs had been previously generated, and a certificate was requested and received for each key pair:

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

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

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

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

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

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

RA KeyEncipher Certificate
  Status: Available
  Certificate Serial Number: 34BCF89F
  Key Usage: Encryption
```

Related Commands

crypto ca enroll
crypto ca authenticate

show crypto key mypubkey rsa

To view your router's RSA public keys, use the **show crypto key mypubkey rsa** EXEC command.

```
show crypto key mypubkey rsa
```

Syntax Description

There are no arguments or keywords with this command.

Command Mode

EXEC

Usage Guidelines

This command first appeared in Cisco IOS Release 11.3T.

This command displays your router's RSA public keys.

Sample Display

The following is sample output from the **show crypto key mypubkey rsa** command. Special usage RSA keys were previously generated for this router using the **crypto key generate rsa** command:

```
% Key pair was generated at: 06:07:49 UTC Jan 13 1996
Key name: myrouter.companyx.com
Usage: Signature Key
Key Data:
005C300D 06092A86 4886F70D 01010105 00034B00 30480241 00C5E23B 55D6AB22
04AEF1BA A54028A6 9ACC01C5 129D99E4 64CAB820 847EDAD9 DF0B4E4C 73A05DD2
BD62A8A9 FA603DD2 E2A8A6F8 98F76E28 D58AD221 B583D7A4 71020301 0001

% Key pair was generated at: 06:07:50 UTC Jan 13 1996
Key name: myrouter.companyx.com
Usage: Encryption Key
Key Data:
00302017 4A7D385B 1234EF29 335FC973 2DD50A37 C4F4B0FD 9DADE748 429618D5
18242BA3 2EDFBDD3 4296142A DDF7D3D8 08407685 2F2190A0 0B43F1BD 9A8A26DB
07953829 791FCDE9 A98420F0 6A82045B 90288A26 DBC64468 7789F76E EE21
```

Related Commands

crypto key generate rsa

show crypto key pubkey-chain rsa

To view peers' RSA public keys stored on your router, use the **show crypto key pubkey-chain rsa** EXEC command.

```
show crypto key pubkey-chain rsa [name key-name | address key-address]
```

Syntax Description

name *key-name* (Optional) Specify the name of a particular public key to view.

address *key-address* (Optional) Specify the address of a particular public key to view.

Default

If no keywords are used, this command displays a list of all RSA public keys stored on your router.

Command Mode

EXEC

Usage Guidelines

This command first appeared in Cisco IOS Release 11.3T.

This command shows RSA public keys stored on your router. This includes the RSA public keys of peers who previously sent your router their certificates during peer authentication for IPSec. This also includes any peers' RSA public keys manually configured at your router.

Use the **name** or **address** keywords to display details about a particular RSA public key stored on your router.

Sample Display

The following is sample output from the **show crypto key pubkey-chain rsa** command:

```
Codes: M - Manually Configured, C - Extracted from certificate

Code  Usage      IP-address      Name
M     Signature   10.0.0.1        somerouter.companyx.com
M     Encryption  10.0.0.1        somerouter.companyx.com
C     Signature   172.16.0.1      routerA.companyx.com
C     Encryption  172.16.0.1      routerA.companyx.com
C     General     192.168.10.3    routerB.comanyx.com
```

This sample shows manually configured special usage RSA public keys for the peer "somerouter." This sample also shows three keys obtained from peers' certificates: special usage keys for peer "routerA" and a general purpose key for peer "routerB."

The following is sample output when you issue the command **show crypto key pubkey rsa name somerouter.companyx.com**:

```
Key name: somerouter.companyx.com
Key address: 10.0.0.1
Usage: Signature Key
Source: Manual
Data:
 305C300D 06092A86 4886F70D 01010105 00034B00 30480241 00C5E23B 55D6AB22
 04AEF1BA A54028A6 9ACC01C5 129D99E4 64CAB820 847EDAD9 DF0B4E4C 73A05DD2
 BD62A8A9 FA603DD2 E2A8A6F8 98F76E28 D58AD221 B583D7A4 71020301 0001

Key name: somerouter.companyx.com
Key address: 10.0.0.1
Usage: Encryption Key
Source: Manual
Data:
 00302017 4A7D385B 1234EF29 335FC973 2DD50A37 C4F4B0FD 9DADE748 429618D5
 18242BA3 2EDFBDD3 4296142A DDF7D3D8 08407685 2F2190A0 0B43F1BD 9A8A26DB
 07953829 791FCDE9 A98420F0 6A82045B 90288A26 DBC64468 7789F76E EE21
```

Note that the Source field in the above example indicates “Manual,” meaning that the keys were manually configured on your router, not received in the peer’s certificate.

The following is sample output when you issue the command **show crypto key pubkey rsa address 192.168.10.3**:

```
Key name: routerB.companyx.com
Key address: 192.168.10.3
Usage: General Purpose Key
Source: Certificate
Data:
 0738BC7A 2BC3E9F0 679B00FE 53987BCC 01030201 42DD06AF E228D24C 458AD228
 58BB5DDD F4836401 2A2D7163 219F882E 64CE69D4 B583748A 241BED0F 6E7F2F16
 0DE0986E DF02031F 4B0B0912 F68200C4 C625C389 0BFF3321 A2598935 C1B1
```

Note that the Source field in the above example indicates “Certificate,” meaning that the keys were received by your router by way of the other router’s certificate.

What to Do Next

After you are done configuring this feature, you should configure IKE and IPSec. IKE configuration is described in the “Internet Key Exchange Security Protocol” feature documentation. IPSec configuration is described in the “IPSec Network Security” feature documentation.

