



Internet Key Exchange Security Protocol Commands

This chapter describes Internet Key Exchange Security Protocol (IKE) commands.

The Internet Key Exchange (IKE) protocol is a key management protocol standard that is used in conjunction with the IPsec standard. IPsec is an IP security feature that provides robust authentication and encryption of IP packets.

IPsec can be configured without IKE, but IKE enhances IPsec by providing additional features, flexibility, and ease of configuration for the IPsec standard.

IKE is a hybrid protocol that implements the Oakley key exchange and Skeme key exchange inside the Internet Security Association and Key Management Protocol (ISAKMP) framework. (ISAKMP, Oakley, and Skeme are security protocols implemented by IKE.)

Refer to the *Command Reference Master Index* or search online to find complete descriptions of other commands used when configuring IKE.

For configuration information, refer to the chapter “Configuring Internet Key Exchange Security Protocol” in the *Security Configuration Guide*.

address

To specify the IP address of the remote peer's RSA public key you will manually configure, use the **address** public key configuration command. This command should only be used when the router has a single interface that processes IPsec.

address *ip-address*

Syntax Description

ip-address	Specifies the IP address of the remote peer.
-------------------	--

Defaults

This command has no defaults.

Command Modes

Public key configuration

Command History

Release	Modification
11.3 T	This command was introduced.

Usage Guidelines

Use this command in conjunction with the **named-key** command to specify which IPsec peer's RSA public key you will manually configure next.

Examples

This example manually specifies the RSA public keys of an IPsec peer.

```
myrouter(config)# crypto key pubkey-chain rsa
myrouter(config-pubkey-chain)# named-key otherpeer.domain.com
myrouter(config-pubkey-key)# address 10.5.5.1
myrouter(config-pubkey-key)# key-string
myrouter(config-pubkey)# 005C300D 06092A86 4886F70D 01010105
myrouter(config-pubkey)# 00034B00 30480241 00C5E23B 55D6AB22
myrouter(config-pubkey)# 04AEF1BA A54028A6 9ACC01C5 129D99E4
myrouter(config-pubkey)# 64CAB820 847EDAD9 DF0B4E4C 73A05DD2
myrouter(config-pubkey)# BD62A8A9 FA603DD2 E2A8A6F8 98F76E28
myrouter(config-pubkey)# D58AD221 B583D7A4 71020301 0001
myrouter(config-pubkey)# quit
myrouter(config-pubkey-key)# exit
myrouter(config-pubkey-chain)# exit
myrouter(config)#
```

Related Commands

Command	Description
addressed-key	Specifies the RSA public key of the peer you will manually configure.
crypto key pubkey-chain rsa	Enters public key configuration mode (to allow you to manually specify the RSA public keys of other devices).
key-string (IKE)	Specifies the RSA public key of a remote peer.
show crypto key pubkey-chain rsa	Displays peer RSA public keys stored on your router.

addressed-key

To specify which peer's RSA public key you will manually configure, use the **addressed-key** public key chain configuration command.

```
addressed-key key-address [encryption | signature]
```

Syntax Description	
<i>key-address</i>	Specifies the IP address of the remote peer's RSA keys.
encryption	(Optional) Indicates that the RSA public key to be specified will be an encryption special usage key.
signature	(Optional) Indicates that the RSA public key to be specified will be a signature special usage key.

Defaults If neither the **encryption** nor **signature** keywords are used, general purpose keys will be specified.

Command Modes Public key chain configuration. This command invokes public key configuration mode.

Command History	Release	Modification
	11.3 T	This command was introduced.

Usage Guidelines

Use this command or the **named-key** command to specify which IPSec peer's RSA public key you will manually configure next.

Follow this command with the **key string** (IKE) command to specify the key.

If the IPSec remote peer generated general purpose RSA keys, do not use the **encryption** or **signature** keywords.

If the IPSec remote peer generated special usage keys, you must manually specify both keys: perform this command and the **key-string** command twice and use the **encryption** and **signature** keywords respectively.

Examples

This example manually specifies the RSA public keys of two IPsec peers. The peer at 10.5.5.1 uses general purpose keys, and the other peer uses special usage keys.

```
myrouter(config)# crypto key pubkey-chain rsa
myrouter(config-pubkey-chain)# named-key otherpeer.domain.com
myrouter(config-pubkey-key)# address 10.5.5.1
myrouter(config-pubkey-key)# key-string
myrouter(config-pubkey)# 005C300D 06092A86 4886F70D 01010105
myrouter(config-pubkey)# 00034B00 30480241 00C5E23B 55D6AB22
myrouter(config-pubkey)# 04AEF1BA A54028A6 9ACC01C5 129D99E4
myrouter(config-pubkey)# 64CAB820 847EDAD9 DF0B4E4C 73A05DD2
myrouter(config-pubkey)# BD62A8A9 FA603DD2 E2A8A6F8 98F76E28
myrouter(config-pubkey)# D58AD221 B583D7A4 71020301 0001
myrouter(config-pubkey)# quit
myrouter(config-pubkey-key)# exit
myrouter(config-pubkey-chain)# addressed-key 10.1.1.2 encryption
myrouter(config-pubkey-key)# key-string
myrouter(config-pubkey)# 00302017 4A7D385B 1234EF29 335FC973
myrouter(config-pubkey)# 2DD50A37 C4F4B0FD 9DADE748 429618D5
myrouter(config-pubkey)# 18242BA3 2EDFBDD3 4296142A DDF7D3D8
myrouter(config-pubkey)# 08407685 2F2190A0 0B43F1BD 9A8A26DB
myrouter(config-pubkey)# 07953829 791FCDE9 A98420F0 6A82045B
myrouter(config-pubkey)# 90288A26 DBC64468 7789F76E EE21
myrouter(config-pubkey)# quit
myrouter(config-pubkey-key)# exit
myrouter(config-pubkey-chain)# addressed-key 10.1.1.2 signature
myrouter(config-pubkey-key)# key-string
myrouter(config-pubkey)# 0738BC7A 2BC3E9F0 679B00FE 53987BCC
myrouter(config-pubkey)# 01030201 42DD06AF E228D24C 458AD228
myrouter(config-pubkey)# 58BB5DDD F4836401 2A2D7163 219F882E
myrouter(config-pubkey)# 64CE69D4 B583748A 241BED0F 6E7F2F16
myrouter(config-pubkey)# ODE0986E DF02031F 4B0B0912 F68200C4
myrouter(config-pubkey)# C625C389 0BFF3321 A2598935 C1B1
myrouter(config-pubkey)# quit
myrouter(config-pubkey-key)# exit
myrouter(config-pubkey-chain)# exit
myrouter(config)#
```

Related Commands

Command	Description
crypto key pubkey-chain rsa	Enters public key configuration mode (to allow you to manually specify the RSA public keys of other devices).
key-string (IKE)	Specifies the RSA public key of a remote peer.
named-key	Specifies which peer RSA public key you will manually configure.
show crypto key pubkey-chain rsa	Displays peer RSA public keys stored on your router.

authentication (IKE policy)

To specify the authentication method within an IKE policy, use the **authentication (IKE policy)** ISAKMP policy configuration command. IKE policies define a set of parameters to be used during IKE negotiation. Use the **no** form of this command to reset the authentication method to the default value.

```
authentication { rsa-sig | rsa-encr | pre-share }
```

```
no authentication
```

Syntax Description	Command	Description
	rsa-sig	Specifies RSA signatures as the authentication method.
	rsa-encr	Specifies RSA encrypted nonces as the authentication method.
	pre-share	Specifies preshared keys as the authentication method.

Defaults RSA signatures

Command Modes ISAKMP policy configuration (config-isakmp)

Command History	Release	Modification
	11.3 T	This command was introduced.

Usage Guidelines

Use this command to specify the authentication method to be used in an IKE policy.

If you specify RSA signatures, you must configure your peer routers to obtain certificates from a Certification Authority (CA).

If you specify RSA encrypted nonces, you must ensure that each peer has the other peer's RSA public keys. (See the **crypto key pubkey-chain rsa**, **addressed-key**, **named-key**, **address**, and **key-string** (IKE) commands.)

If you specify preshared keys, you must also separately configure these preshared keys. (See the **crypto isakmp identity** and **crypto isakmp key** commands.)

Examples This example configures an IKE policy with preshared keys as the authentication method (all other parameters are set to the defaults):

```
MyPeerRouter(config)# crypto isakmp policy 15
MyPeerRouter(config-isakmp)# authentication pre-share
MyPeerRouter(config-isakmp)# exit
MyPeerRouter(config)#
```

Related Commands

Command	Description
crypto isakmp key	Configures a preshared authentication key.
crypto isakmp policy	Defines an IKE policy.
crypto key generate rsa (IKE)	Generates RSA key pairs.
encryption (IKE policy)	Specifies the encryption algorithm within an IKE policy.
group (IKE policy)	Specifies the Diffie-Hellman group identifier within an IKE policy.
hash (IKE policy)	Specifies the hash algorithm within an IKE policy.
lifetime (IKE policy)	Specifies the lifetime of an IKE SA.
show crypto isakmp policy	Displays the parameters for each IKE policy.

clear crypto isakmp

To clear active IKE connections, use the **clear crypto isakmp** global configuration command.

clear crypto isakmp [*connection-id*]

Syntax Description	<i>connection-id</i> (Optional) Specifies which connection to clear. If this argument is not used, all existing connections will be cleared.
---------------------------	--

Defaults	If the <i>connection-id</i> argument is not used, all existing IKE connections will be cleared when this command is issued.
-----------------	---

Command Modes	Global configuration
----------------------	----------------------

Command History	Release	Modification
	11.3 T	This command was introduced.

Usage Guidelines	Use this command to clear active IKE connections.
-------------------------	---

Examples This example clears an IKE connection between two peers connected by interfaces 172.21.114.123 and 172.21.114.67.

```
MyPeerRouter# show crypto isakmp sa
      dst          src          state          conn-id  slot
172.21.114.123 172.21.114.67  QM_IDLE        1         0
155.0.0.2      155.0.0.1     QM_IDLE        8         0

MyPeerRouter# configure terminal
Enter configuration commands, one per line. End with CNTL/Z.
MyPeerRouter(config)# clear crypto isakmp 1
MyPeerRouter(config)# exit
MyPeerRouter# show crypto isakmp sa
      dst          src          state          conn-id  slot
155.0.0.2      155.0.0.1     QM_IDLE        8         0

MyPeerRouter#
```

Related Commands	Command	Description
	show crypto isakmp sa	Displays all current IKE SAs at a peer.

crypto isakmp enable

To globally enable IKE at your peer router, use the **crypto isakmp enable** global configuration command. Use the **no** form of this command to disable IKE at the peer.

crypto isakmp enable

no crypto isakmp enable

Syntax Description

This command has no arguments or keywords.

Defaults

IKE is enabled.

Command Modes

Global configuration

Command History

Release	Modification
11.3 T	This command was introduced.

Usage Guidelines

IKE is enabled by default. IKE does not have to be enabled for individual interfaces, but is enabled globally for all interfaces at the router.

If you do not want IKE to be used in your IPsec implementation, you can disable IKE at all your IPsec peers. If you disable IKE at one peer you must disable it at all your IPsec peers.

If you disable IKE, you will have to make these concessions at the peers:

- You must manually specify all the IPsec security associations (SAs) in the crypto maps at the peers. (Crypto map configuration is described in the chapter “Configuring IPsec Network Security” in the *Security Configuration Guide*.)
- The peers’ IPsec SAs will never time out for a given IPsec session.
- During IPsec sessions between the peers, the encryption keys will never change.
- Anti-replay services will not be available between the peers.
- Certification Authority (CA) support cannot be used.

Examples

This example disables IKE at one peer. (The same command should be issued at all remote peers.)

```
no crypto isakmp enable
```

crypto isakmp identity

To define the identity the router uses when participating in the IKE protocol, use the **crypto isakmp identity** global configuration command. Set an ISAKMP identity whenever you specify preshared keys. Use the **no** form of this command to reset the ISAKMP identity to the default value (address).

crypto isakmp identity {address | hostname}

no crypto isakmp identity

Syntax Description	address	hostname
	Sets the ISAKMP identity to the IP address of the interface that is used to communicate to the remote peer during IKE negotiations.	Sets the ISAKMP identity to the host name concatenated with the domain name (for example, myhost.domain.com).

Defaults The IP address is used for the ISAKMP identity.

Command Modes Global configuration

Command History	Release	Modification
	11.3 T	This command was introduced.

Usage Guidelines Use this command to specify an ISAKMP identity either by IP address or by host name.

The **address** keyword is typically used when there is only one interface (and therefore only one IP address) that will be used by the peer for IKE negotiations, and the IP address is known.

The **hostname** keyword should be used if there is more than one interface on the peer that might be used for IKE negotiations, or if the interface's IP address is unknown (such as with dynamically assigned IP addresses).

As a general rule, you should set all peers' identities in the same way, either by IP address or by host name.

Examples The following example uses preshared keys at two peers and sets both their ISAKMP identities to IP address. At the local peer (at 10.0.0.1) the ISAKMP identity is set and the preshared key is specified:

```
crypto isakmp identity address
crypto isakmp key sharedkeystring address 192.168.1.33
```

At the remote peer (at 192.168.1.33) the ISAKMP identity is set and the same preshared key is specified:

```
crypto isakmp identity address
crypto isakmp key sharedkeystring address 10.0.0.1
```



Note

In the preceding example if the **crypto isakmp identity** command had not been performed, the ISAKMP identities would have still been set to IP address, the default identity.

The following example uses preshared keys at two peers and sets both their ISAKMP identities to hostname.

At the local peer the ISAKMP identity is set and the preshared key is specified:

```
crypto isakmp identity hostname
crypto isakmp key sharedkeystring hostname RemoteRouter.domain.com
ip host RemoteRouter.domain.com 192.168.0.1
```

At the remote peer the ISAKMP identity is set and the same preshared key is specified:

```
crypto isakmp identity hostname
crypto isakmp key sharedkeystring hostname LocalRouter.domain.com
ip host LocalRouter.domain.com 10.0.0.1 10.0.0.2
```

In the above example, host names are used for the peers' identities because the local peer has two interfaces that might be used during an IKE negotiation.

In the above example the IP addresses are also mapped to the host names; this mapping is not necessary if the routers' host names are already mapped in DNS.

Related Commands

Command	Description
authentication (IKE policy)	Specifies the authentication method within an IKE policy.
crypto isakmp key	Configures a preshared authentication key.

crypto isakmp key

To configure a preshared authentication key, use the **crypto isakmp key** global configuration command. You must configure this key whenever you specify preshared keys in an IKE policy. Use the **no** form of this command to delete a preshared authentication key.

crypto isakmp key *keystring* **address** *peer-address*

crypto isakmp key *keystring* **hostname** *peer-hostname*

no crypto isakmp key *keystring* **address** *peer-address*

no crypto isakmp key *keystring* **hostname** *peer-hostname*

Syntax Description		
<i>keystring</i>	Specify the preshared key. Use any combination of alphanumeric characters up to 128 bytes. This preshared key must be identical at both peers.	
<i>peer-address</i>	Specify the IP address of the remote peer.	
<i>peer-hostname</i>	Specify the host name of the remote peer. This is the peer's host name concatenated with its domain name (for example, myhost.domain.com).	

Defaults There is no default preshared authentication key.

Command Modes Global configuration

Command History	Release	Modification
	11.3 T	This command was introduced.

Usage Guidelines

Use this command to configure preshared authentication keys. You must perform this command at both peers. If an IKE policy includes preshared keys as the authentication method, these preshared keys must be configured at both peers—otherwise the policy cannot be used (the policy will not be submitted for matching by the IKE process). The **crypto isakmp key** command is the second task required to configure the preshared keys at the peers. (The first task is accomplished with the **crypto isakmp identity** command.)

Use the **address** keyword if the remote peer ISAKMP identity was set with its IP address.

Use the **hostname** keyword if the remote ISAKMP identity was set with its host name.

With the **hostname** keyword, you might also need to map the remote peer's host name to all IP addresses of the remote peer interfaces that could be used during the IKE negotiation. (This is done with the **ip host** command.) You need to map the host name to IP address unless this mapping is already done in a DNS server.

Examples The remote peer "RemoteRouter" specifies an ISAKMP identity by address:

```
crypto isakmp identity address
```

The local peer "LocalRouter" also specifies an ISAKMP identity, but by host name:

```
crypto isakmp identity hostname
```

Now, the preshared key must be specified at each peer.

The local peer specifies the preshared key and designates the remote peer by its IP address:

```
crypto isakmp key sharedkeystring address 192.168.1.33
```

The remote peer specifies the same preshared key and designates the local peer by its host name:

```
crypto isakmp key sharedkeystring hostname LocalRouter.domain.com
```

The remote peer also maps multiple IP addresses to the same host name for the local peer because the local peer has two interfaces which both might be used during an IKE negotiation with the local peer. These two interfaces' IP addresses (10.0.0.1 and 10.0.0.2) are both mapped to the remote peer's host name:

```
ip host LocalRouter.domain.com 10.0.0.1 10.0.0.2
```

(This mapping would not have been necessary if LocalRouter.domain.com was already mapped in DNS.)

In this example, a remote peer specifies its ISAKMP identity by address, and the local peer specifies its ISAKMP identity by host name. Depending on the circumstances in your network, both peers could specify their ISAKMP identity by address, or both by host name.

Related Commands

Command	Description
authentication (IKE policy)	Specifies the authentication method within an IKE policy.
crypto isakmp identity	Defines the identity the router uses when participating in the IKE protocol.
ip host	Defines a static host name-to-address mapping in the host cache.

crypto isakmp policy

To define an IKE policy, use the **crypto isakmp policy** global configuration command. IKE policies define a set of parameters to be used during the IKE negotiation. Use the **no** form of this command to delete an IKE policy.

crypto isakmp policy *priority*

no crypto isakmp policy

Syntax Description	<i>priority</i>	Uniquely identifies the IKE policy and assigns a priority to the policy. Use an integer from 1 to 10,000, with 1 being the highest priority and 10,000 the lowest.
---------------------------	-----------------	--

Defaults

There is a default policy which is always the lowest priority. This default policy contains default values for the encryption, hash, authentication, Diffie-Hellman group, and lifetime parameters. (The parameter defaults are listed below in the Usage Guidelines section.)

When you create an IKE policy, if you do not specify a value for a particular parameter, the default for that parameter will be used.

Command Modes

Global configuration

Command History	Release	Modification
	11.3 T	This command was introduced.

Usage Guidelines

Use this command to specify the parameters to be used during an IKE negotiation. (These parameters are used to create the IKE security association [SA].)

This command invokes the ISAKMP policy configuration (config-isakmp) command mode. While in the ISAKMP policy configuration command mode, the following commands are available to specify the parameters in the policy:

- **encryption (IKE policy)**; default = 56-bit DES-CBC
- **hash (IKE policy)**; default = SHA-1
- **authentication (IKE policy)**; default = RSA signatures
- **group (IKE policy)**; default = 768-bit Diffie-Hellman
- **lifetime (IKE policy)**; default = 86,400 seconds (one day)

If you do not specify one of these commands for a policy, the default value will be used for that parameter.

To exit the config-isakmp command mode, type **exit**.

You can configure multiple IKE policies on each peer participating in IPSec. When the IKE negotiation begins, it tries to find a common policy configured on both peers, starting with the highest priority policies as specified on the remote peer.

Examples

The following example configures two policies for the peer:

```
crypto isakmp policy 15
  hash md5
  authentication rsa-sig
  group 2
  lifetime 5000
crypto isakmp policy 20
  authentication pre-share
  lifetime 10000
```

The above configuration results in the following policies:

```
MyPeerRouter# show crypto isakmp policy
```

```
Protection suite priority 15
encryption algorithm:DES - Data Encryption Standard (56 bit keys)
hash algorithm:Message Digest 5
authentication method:Rivest-Shamir-Adleman Signature
Diffie-Hellman Group:#2 (1024 bit)
lifetime:5000 seconds, no volume limit
Protection suite priority 20
encryption algorithm:DES - Data Encryption Standard (56 bit keys)
hash algorithm:Secure Hash Standard
authentication method:preshared Key
Diffie-Hellman Group:#1 (768 bit)
lifetime:10000 seconds, no volume limit
Default protection suite
encryption algorithm:DES - Data Encryption Standard (56 bit keys)
hash algorithm:Secure Hash Standard
authentication method:Rivest-Shamir-Adleman Signature
Diffie-Hellman Group:#1 (768 bit)
lifetime:86400 seconds, no volume limit
```

Related Commands

Command	Description
authentication (IKE policy)	Specifies the authentication method within an IKE policy.
encryption (IKE policy)	Specifies the encryption algorithm within an IKE policy.
group (IKE policy)	Specifies the Diffie-Hellman group identifier within an IKE policy.
hash (IKE policy)	Specifies the hash algorithm within an IKE policy.
lifetime (IKE policy)	Specifies the lifetime of an IKE SA.
show crypto isakmp policy	Displays the parameters for each IKE policy.

crypto key generate rsa (IKE)

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 RSA special usage key pairs should be generated (i.e. one encryption pair and one signature pair), instead of one general purpose key pair.

Defaults

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

Command Modes

Global configuration

Command History

Release	Modification
11.3 T	This command was introduced.

Usage Guidelines

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 host name 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 host name and IP domain name.

This command is not 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 types of RSA key pairs: special usage keys and general purpose keys. When you generate RSA key pairs, you can indicate 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.

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 authentication methods, 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 414 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 414 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

The following example generates special usage RSA keys:

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

The following 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.domain.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

Command	Description
show crypto key mypubkey rsa	Displays the RSA public key(s) of your router.

crypto key pubkey-chain rsa

To enter public key configuration mode (so you can manually specify other devices' RSA public keys), use the **crypto key pubkey-chain rsa** global configuration command.

crypto key pubkey-chain rsa

Syntax Description

This command has no arguments or keywords.

Defaults

This command has no defaults.

Command Modes

Global configuration. This command invokes public key chain configuration mode.

Command History

Release	Modification
11.3 T	This command was introduced.

Usage Guidelines

Use this command to enter public key chain configuration mode. Use this command when you need to manually specify other IPSec peers' RSA public keys. You need to specify other peers' keys when you configure RSA encrypted nonces as the authentication method in an IKE policy at your peer router.

Examples

This example manually specifies the RSA public keys of two other IPSec peers. The remote peers use their IP address as their identity.

```
myrouter(config)# crypto key pubkey-chain rsa
myrouter(config-pubkey-chain)# addressed-key 10.5.5.1
myrouter(config-pubkey-key)# key-string
myrouter(config-pubkey)# 00302017 4A7D385B 1234EF29 335FC973
myrouter(config-pubkey)# 2DD50A37 C4F4B0FD 9DADE748 429618D5
myrouter(config-pubkey)# 18242BA3 2EDFBDD3 4296142A DDF7D3D8
myrouter(config-pubkey)# 08407685 2F2190A0 0B43F1BD 9A8A26DB
myrouter(config-pubkey)# 07953829 791FCDE9 A98420F0 6A82045B
myrouter(config-pubkey)# 90288A26 DBC64468 7789F76E EE21
myrouter(config-pubkey)# quit
myrouter(config-pubkey-key)# exit
myrouter(config-pubkey-chain)# addressed-key 10.1.1.2
myrouter(config-pubkey-key)# key-string
myrouter(config-pubkey)# 0738BC7A 2BC3E9F0 679B00FE 53987BCC
myrouter(config-pubkey)# 01030201 42DD06AF E228D24C 458AD228
myrouter(config-pubkey)# 58BB5DDD F4836401 2A2D7163 219F882E
myrouter(config-pubkey)# 64CE69D4 B583748A 241BED0F 6E7F2F16
myrouter(config-pubkey)# 0DE0986E DF02031F 4B0B0912 F68200C4
myrouter(config-pubkey)# C625C389 0BFF3321 A2598935 C1B1
myrouter(config-pubkey)# quit
myrouter(config-pubkey-key)# exit
myrouter(config-pubkey-chain)# exit
myrouter(config)#
```

Related Commands	Command	Description
	address	Specifies the IP address of the remote RSA public key of the remote peer you will manually configure.
	addressed-key	Specifies the RSA public key of the peer you will manually configure.
	key-string (IKE)	Specifies the RSA public key of a remote peer.
	named-key	Specifies which peer RSA public key you will manually configure.
	show crypto key pubkey-chain rsa	Displays peer RSA public keys stored on your router.

encryption (IKE policy)

To specify the encryption algorithm within an IKE policy, use the **encryption** (IKE policy) ISAKMP policy configuration command. IKE policies define a set of parameters to be used during IKE negotiation. Use the **no** form of this command to reset the encryption algorithm to the default value.

encryption des

no encryption

Syntax Description	des Specifies 56-bit DES-CBC as the encryption algorithm.														
Defaults	The 56-bit DES-CBC encryption algorithm.														
Command Modes	ISAKMP policy configuration (config-isakmp)														
Command History	<table border="1"> <thead> <tr> <th>Release</th> <th>Modification</th> </tr> </thead> <tbody> <tr> <td>11.3 T</td> <td>This command was introduced.</td> </tr> </tbody> </table>	Release	Modification	11.3 T	This command was introduced.										
Release	Modification														
11.3 T	This command was introduced.														
Usage Guidelines	Use this command to specify the encryption algorithm to be used in an IKE policy.														
Examples	<p>This example configures an IKE policy with the 56-bit DES encryption algorithm (all other parameters are set to the defaults):</p> <pre>MyPeerRouter(config)# crypto isakmp policy 15 MyPeerRouter(config-isakmp)# encryption des MyPeerRouter(config-isakmp)# exit MyPeerRouter(config)#</pre>														
Related Commands	<table border="1"> <thead> <tr> <th>Command</th> <th>Description</th> </tr> </thead> <tbody> <tr> <td>authentication (IKE policy)</td> <td>Specifies the authentication method within an IKE policy.</td> </tr> <tr> <td>crypto isakmp policy</td> <td>Defines an IKE policy.</td> </tr> <tr> <td>group (IKE policy)</td> <td>Specifies the Diffie-Hellman group identifier within an IKE policy.</td> </tr> <tr> <td>hash (IKE policy)</td> <td>Specifies the hash algorithm within an IKE policy.</td> </tr> <tr> <td>lifetime (IKE policy)</td> <td>Specifies the lifetime of an IKE SA.</td> </tr> <tr> <td>show crypto isakmp policy</td> <td>Displays the parameters for each IKE policy.</td> </tr> </tbody> </table>	Command	Description	authentication (IKE policy)	Specifies the authentication method within an IKE policy.	crypto isakmp policy	Defines an IKE policy.	group (IKE policy)	Specifies the Diffie-Hellman group identifier within an IKE policy.	hash (IKE policy)	Specifies the hash algorithm within an IKE policy.	lifetime (IKE policy)	Specifies the lifetime of an IKE SA.	show crypto isakmp policy	Displays the parameters for each IKE policy.
Command	Description														
authentication (IKE policy)	Specifies the authentication method within an IKE policy.														
crypto isakmp policy	Defines an IKE policy.														
group (IKE policy)	Specifies the Diffie-Hellman group identifier within an IKE policy.														
hash (IKE policy)	Specifies the hash algorithm within an IKE policy.														
lifetime (IKE policy)	Specifies the lifetime of an IKE SA.														
show crypto isakmp policy	Displays the parameters for each IKE policy.														

group (IKE policy)

To specify the Diffie-Hellman group identifier within an IKE policy, use the **group (IKE policy)** ISAKMP policy configuration command. IKE policies define a set of parameters to be used during IKE negotiation. Use the **no** form of this command to reset the Diffie-Hellman group identifier to the default value.

group {1 | 2}

no group

Syntax Description	1	Specifies the 768-bit Diffie-Hellman group.
	2	Specifies the 1024-bit Diffie-Hellman group.

Defaults 768-bit Diffie-Hellman (group 1)

Command Modes ISAKMP policy configuration (config-isakmp)

Command History	Release	Modification
	11.3 T	This command was introduced.

Usage Guidelines Use this command to specify the Diffie-Hellman group to be used in an IKE policy.

Examples This example configures an IKE policy with the 1024-bit Diffie-Hellman group (all other parameters are set to the defaults):

```
MyPeerRouter (config)# crypto isakmp policy 15
MyPeerRouter (config-isakmp)# group 2
MyPeerRouter (config-isakmp)# exit
MyPeerRouter (config)#
```

Related Commands	Command	Description
	authentication (IKE policy)	Specifies the authentication method within an IKE policy.
	crypto isakmp policy	Defines an IKE policy.
	encryption (IKE policy)	Specifies the encryption algorithm within an IKE policy.
	hash (IKE policy)	Specifies the hash algorithm within an IKE policy.
	lifetime (IKE policy)	Specifies the lifetime of an IKE SA.
	show crypto isakmp policy	Displays the parameters for each IKE policy.

hash (IKE policy)

To specify the hash algorithm within an IKE policy, use the **hash (IKE policy)** ISAKMP policy configuration command. IKE policies define a set of parameters to be used during IKE negotiation. Use the **no** form of this command to reset the hash algorithm to the default SHA-1 hash algorithm.

```
hash {sha | md5}
no hash
```

Syntax Description

sha	Specifies SHA-1 (HMAC variant) as the hash algorithm.
md5	Specifies MD5 (HMAC variant) as the hash algorithm.

Defaults

The SHA-1 hash algorithm.

Command Modes

ISAKMP policy configuration (config-isakmp)

Command History

Release	Modification
11.3 T	This command was introduced.

Usage Guidelines

Use this command to specify the hash algorithm to be used in an IKE policy.

Examples

This example configures an IKE policy with the MD5 hash algorithm (all other parameters are set to the defaults):

```
MyPeerRouter (config)# crypto isakmp policy 15
MyPeerRouter (config-isakmp)# hash md5
MyPeerRouter (config-isakmp)# exit
MyPeerRouter (config)#
```

Related Commands

Command	Description
authentication (IKE policy)	Specifies the authentication method within an IKE policy.
crypto isakmp policy	Defines an IKE policy.
encryption (IKE policy)	Specifies the encryption algorithm within an IKE policy.
group (IKE policy)	Specifies the Diffie-Hellman group identifier within an IKE policy.
lifetime (IKE policy)	Specifies the lifetime of an IKE SA.
show crypto isakmp policy	Displays the parameters for each IKE policy.

key-string (IKE)

To manually specify a remote peer's RSA public key, use the **key-string** public key configuration command.

key-string *key-string*

Syntax Description	<i>key-string</i>	Enter the key in hexadecimal format. While entering the key data you can press the return key to continue entering data.
---------------------------	-------------------	--

Defaults This command has no defaults.

Command Modes Public key configuration

Command History	Release	Modification
	11.3 T	This command was introduced.

Usage Guidelines Use this command to manually specify the RSA public key of an IPSec peer. Before using this command you must identify the remote peer using either the **addressed-key** or **named-key** command.

If possible, to avoid mistakes, you should cut and paste the key data (instead of attempting to type in the data).

To complete the command, you must return to the global configuration mode by typing **quit** at the config-pubkey prompt.

Examples This example manually specifies the RSA public keys of an IPSec peer.

```
myrouter(config)# crypto key pubkey-chain rsa
myrouter(config-pubkey-chain)# named-key otherpeer.domain.com
myrouter(config-pubkey-key)# address 10.5.5.1
myrouter(config-pubkey-key)# key-string
myrouter(config-pubkey)# 005C300D 06092A86 4886F70D 01010105
myrouter(config-pubkey)# 00034B00 30480241 00C5E23B 55D6AB22
myrouter(config-pubkey)# 04AEF1BA A54028A6 9ACC01C5 129D99E4
myrouter(config-pubkey)# 64CAB820 847EDAD9 DF0B4E4C 73A05DD2
myrouter(config-pubkey)# BD62A8A9 FA603DD2 E2A8A6F8 98F76E28
myrouter(config-pubkey)# D58AD221 B583D7A4 71020301 0001
myrouter(config-pubkey)# quit
myrouter(config-pubkey-key)# exit
myrouter(config-pubkey-chain)# exit
myrouter(config)#
```

Related Commands	Command	Description
	addressed-key	Specifies the RSA public key of the peer you will manually configure.
	crypto key pubkey-chain rsa	Enters public key configuration mode (to allow you to manually specify the RSA public keys of other devices).
	named-key	Specifies which peer RSA public key you will manually configure.

Command	Description
show crypto key pubkey-chain rsa	Displays peer RSA public keys stored on your router.

lifetime (IKE policy)

To specify the lifetime of an IKE security association (SA), use the **lifetime** (IKE policy) ISAKMP policy configuration command. Use the **no** form of this command to reset the SA lifetime to the default value.

lifetime *seconds*

no lifetime

Syntax Description	<i>seconds</i>	Specifies how many seconds each SA should exist before expiring. Use an integer from 60 to 86,400 seconds.
---------------------------	----------------	--

Defaults	86,400 seconds (one day)
-----------------	--------------------------

Command Modes	ISAKMP policy configuration (config-isakmp)
----------------------	---

Command History	Release	Modification
	11.3 T	This command was introduced.

Usage Guidelines Use this command to specify how long an IKE SA exists before expiring.

When IKE begins negotiations, the first thing it does is agree upon the security parameters for its own session. The agreed-upon parameters are then referenced by an SA at each peer. The SA is retained by each peer until the SA's lifetime expires. Before an SA expires, it can be reused by subsequent IKE negotiations, which can save time when setting up new IPSec SAs. New SAs are negotiated before current SAs expire.

So, to save setup time for IPSec, configure a longer IKE SA lifetime. However, the shorter the lifetime (up to a point), the more secure the IKE negotiation is likely to be.

Note that when your local peer initiates an IKE negotiation between itself and a remote peer, an IKE policy can be selected only if the lifetime of the remote peer's policy is longer than or equal to the lifetime of the local peer's policy. Then, if the lifetimes are not equal, the shorter lifetime will be selected. To restate this behavior: If the two peer's policies' lifetimes are not the same, the initiating peer's lifetime must be shorter and the responding peer's lifetime must be longer, and the shorter lifetime will be used.

Examples This example configures an IKE policy with a security association lifetime of 600 seconds (10 minutes), and all other parameters are set to the defaults:

```
MyPeerRouter(config)# crypto isakmp policy 15
MyPeerRouter(config-isakmp)# lifetime 600
MyPeerRouter(config-isakmp)# exit
MyPeerRouter(config)#
```

Related Commands

Command	Description
authentication (IKE policy)	Specifies the authentication method within an IKE policy.
crypto isakmp policy	Defines an IKE policy.
encryption (IKE policy)	Specifies the encryption algorithm within an IKE policy.
group (IKE policy)	Specifies the Diffie-Hellman group identifier within an IKE policy.
hash (IKE policy)	Specifies the hash algorithm within an IKE policy.
show crypto isakmp policy	Displays the parameters for each IKE policy.

named-key

To specify which peer's RSA public key you will manually configure, use the **named-key** public key chain configuration command. This command should only be used when the router has a single interface that processes IPsec.

named-key *key-name* [**encryption** | **signature**]

Syntax Description

<i>key-name</i>	Specifies the name of the remote peer's RSA keys. This is always the fully qualified domain name of the remote peer; for example, router.domain.com.
encryption	(Optional) Indicates that the RSA public key to be specified will be an encryption special usage key.
signature	(Optional) Indicates that the RSA public key to be specified will be a signature special usage key.

Defaults

If neither the **encryption** nor **signature** keywords are used, general purpose keys will be specified.

Command Modes

Public key chain configuration. This command invokes public key configuration mode.

Command History

Release	Modification
11.3 T	This command was introduced.

Usage Guidelines

Use this command or the **addressed-key** command to specify which IPsec peer's RSA public key you will manually configure next.

Follow this command with the **key-string** (IKE) command to specify the key.

If you use the **named-key** command you also need to use the **address** public key configuration command to specify the IP address of the peer.

If the IPsec remote peer generated general purpose RSA keys, do not use the **encryption** or **signature** keywords.

If the IPsec remote peer generated special usage keys, you must manually specify both keys: perform this command and the **key-string** command twice and use the **encryption** and **signature** keywords, respectively.

Examples

This example manually specifies the RSA public keys of two IPsec peers. The peer at 10.5.5.1 uses general purpose keys, and the other peer uses special purpose keys.

```
myrouter(config)# crypto key pubkey-chain rsa
myrouter(config-pubkey-chain)# named-key otherpeer.domain.com
myrouter(config-pubkey-key)# address 10.5.5.1
myrouter(config-pubkey-key)# key-string
myrouter(config-pubkey)# 005C300D 06092A86 4886F70D 01010105
myrouter(config-pubkey)# 00034B00 30480241 00C5E23B 55D6AB22
myrouter(config-pubkey)# 04AEF1BA A54028A6 9ACC01C5 129D99E4
myrouter(config-pubkey)# 64CAB820 847EDAD9 DF0B4E4C 73A05DD2
myrouter(config-pubkey)# BD62A8A9 FA603DD2 E2A8A6F8 98F76E28
myrouter(config-pubkey)# D58AD221 B583D7A4 71020301 0001
myrouter(config-pubkey)# quit
myrouter(config-pubkey-key)# exit
myrouter(config-pubkey-chain)# addressed-key 10.1.1.2 encryption
myrouter(config-pubkey-key)# key-string
myrouter(config-pubkey)# 00302017 4A7D385B 1234EF29 335FC973
myrouter(config-pubkey)# 2DD50A37 C4F4B0FD 9DADE748 429618D5
myrouter(config-pubkey)# 18242BA3 2EDFBDD3 4296142A DDF7D3D8
myrouter(config-pubkey)# 08407685 2F2190A0 0B43F1BD 9A8A26DB
myrouter(config-pubkey)# 07953829 791FCDE9 A98420F0 6A82045B
myrouter(config-pubkey)# 90288A26 DBC64468 7789F76E EE21
myrouter(config-pubkey)# quit
myrouter(config-pubkey-key)# exit
myrouter(config-pubkey-chain)# addressed-key 10.1.1.2 signature
myrouter(config-pubkey-key)# key-string
myrouter(config-pubkey)# 0738BC7A 2BC3E9F0 679B00FE 098533AB
myrouter(config-pubkey)# 01030201 42DD06AF E228D24C 458AD228
myrouter(config-pubkey)# 58BB5DDD F4836401 2A2D7163 219F882E
myrouter(config-pubkey)# 64CE69D4 B583748A 241BED0F 6E7F2F16
myrouter(config-pubkey)# ODE0986E DF02031F 4B0B0912 F68200C4
myrouter(config-pubkey)# C625C389 0BFF3321 A2598935 C1B1
myrouter(config-pubkey)# quit
myrouter(config-pubkey-key)# exit
myrouter(config-pubkey-chain)# exit
myrouter(config)#
```

Related Commands

Command	Description
address	Specifies the IP address of the remote RSA public key of the remote peer you will manually configure.
addressed-key	Specifies the RSA public key of the peer you will manually configure.
crypto key pubkey-chain rsa	Enters public key configuration mode (to allow you to manually specify the RSA public keys of other devices).
key-string (IKE)	Specifies the RSA public key of a remote peer.
show crypto key pubkey-chain rsa	Displays peer RSA public keys stored on your router.

show crypto isakmp policy

To view the parameters for each IKE policy, use the **show crypto isakmp policy** EXEC command.

show crypto isakmp policy

Syntax Description

This command has no arguments or keywords.

Command Modes

EXEC

Command History

Release	Modification
11.3 T	This command was introduced.

Examples

The following is sample output from the **show crypto isakmp policy** command, after two IKE policies have been configured (with priorities 15 and 20 respectively):

```
MyPeerRouter# show crypto isakmp policy

Protection suite priority 15
encryption algorithm:DES - Data Encryption Standard (56 bit keys)
hash algorithm:Message Digest 5
authentication method:Rivest-Shamir-Adleman Signature
Diffie-Hellman Group:#2 (1024 bit)
lifetime:5000 seconds, no volume limit
Protection suite priority 20
encryption algorithm:DES - Data Encryption Standard (56 bit keys)
hash algorithm:Secure Hash Standard
authentication method:preshared Key
Diffie-Hellman Group:#1 (768 bit)
lifetime:10000 seconds, no volume limit
Default protection suite
encryption algorithm:DES - Data Encryption Standard (56 bit keys)
hash algorithm:Secure Hash Standard
authentication method:Rivest-Shamir-Adleman Signature
Diffie-Hellman Group:#1 (768 bit)
lifetime:86400 seconds, no volume limit
```



Note

Although the output shows “no volume limit” for the lifetimes, you can currently only configure a time lifetime (such as 86,400 seconds); volume limit lifetimes are not configurable.

Related Commands

Command	Description
authentication (IKE policy)	Specifies the authentication method within an IKE policy.
crypto isakmp policy	Defines an IKE policy.
encryption (IKE policy)	Specifies the encryption algorithm within an IKE policy.
group (IKE policy)	Specifies the Diffie-Hellman group identifier within an IKE policy.
hash (IKE policy)	Specifies the hash algorithm within an IKE policy.
lifetime (IKE policy)	Specifies the lifetime of an IKE SA.

show crypto isakmp sa

To view all current IKE security associations (SAs) at a peer, use the **show crypto isakmp sa** EXEC command.

show crypto isakmp sa

Syntax Description This command has no arguments or keywords.

Command Modes EXEC

Command History	Release	Modification
	11.3 T	This command was introduced.

Examples

The following is sample output from the **show crypto isakmp sa** command, after IKE negotiations have successfully completed between two peers:

```
MyPeerRouter# show crypto isakmp sa
      dst          src          state          conn-id  slot
172.21.114.123 172.21.114.67 QM_IDLE        1        0
155.0.0.2      155.0.0.1    QM_IDLE        8        0
```

Table 415 through Table 417 show the various states that may be displayed in the output of the **show crypto isakmp sa** command. When an ISAKMP SA exists, it will most likely be in its quiescent state (OAK_QM_IDLE). For long exchanges, some of the OAK_MM_XXX states may be observed.

Table 415 States in Main Mode Exchange

State	Explanation
OAK_MM_NO_STATE	The ISAKMP SA has been created but nothing else has happened yet. It is “larval” at this stage—there is no state.
OAK_MM_SA_SETUP	The peers have agreed on parameters for the ISAKMP SA.
OAK_MM_KEY_EXCH	The peers have exchanged Diffie-Hellman public keys and have generated a shared secret. The ISAKMP SA remains unauthenticated.
OAK_MM_KEY_AUTH	The ISAKMP SA has been authenticated. If the router initiated this exchange, this state transitions immediately to OAK_QM_IDLE and a Quick Mode exchange begins.

```
show crypto isakmp sa
```

Table 416 States in Aggressive Mode Exchange

State	Explanation
OAK_AG_NO_STATE	The ISAKMP SA has been created but nothing else has happened yet. It is “larval” at this stage—there is no state.
OAK_AG_INIT_EXCH	The peers have done the first exchange in Aggressive Mode but the SA is not authenticated.
OAK_AG_AUTH	The ISAKMP SA has been authenticated. If the router initiated this exchange, this state transitions immediately to OAK_QM_IDLE and a Quick Mode exchange begins.

Table 417 States in Quick Mode Exchange

State	Explanation
OAK_QM_IDLE	The ISAKMP SA is idle. It remains authenticated with its peer and may be used for subsequent Quick Mode exchanges. It is in a quiescent state.

Related Commands

Command	Description
crypto isakmp policy	Defines an IKE policy.
lifetime (IKE policy)	Specifies the lifetime of an IKE SA.

show crypto key mypubkey rsa

To view your router's RSA public key(s), 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 Modes

EXEC

Command History

Release	Modification
11.3 T	This command was introduced.

Usage Guidelines

This command displays your router's RSA public key(s).

Examples

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 (IKE)** command:

```
% Key pair was generated at: 06:07:49 UTC Jan 13 1996
Key name: myrouter.domain.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.domain.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

Command	Description
crypto key generate rsa (IKE)	Generates RSA key pairs.

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 <i>key-name</i>	(Optional) Specify the name of a particular public key to view.
address <i>key-address</i>	(Optional) Specify the address of a particular public key to view.

Defaults

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

Command Modes

EXEC

Command History

Release	Modification
11.3 T	This command was introduced.

Usage Guidelines

This command shows RSA public keys stored on your router. This includes peers' RSA public keys manually configured at your router and keys received by your router via other means (such as by a certificate, if CA support is configured).

If a router reboots, any public key derived by certificates will be lost. This is because the router will ask for certificates again, at which time the public key will be derived again.

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

Examples

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        myrouter.domain.com
M     Encryption  10.0.0.1        myrouter.domain.com
C     Signature   172.16.0.1      routerA.domain.com
C     Encryption  172.16.0.1      routerA.domain.com
C     General     192.168.10.3    routerB.domain1.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."

Certificate support is used in the above example; if certificate support was not in use, none of the peers' keys would show "C" in the code column, but would all have to be manually configured.

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

```
Key name: somerouter.domain.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.domain.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

The Source field in the above example indicates “Manual,” meaning that the keys were manually configured on the 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.domain.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
```

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

■ show crypto key pubkey-chain rsa