Internet Key Exchange for IPsec VPNs
Configuration Guide Cisco IOS XE
Release 2

Americas Headquarters
Cisco Systems, Inc.
170 West Tasman Drive
San Jose, CA 95134-1706
USA
http://www.cisco.com
Tel: 408 526-4000
800 553-NETS (6387)
Fax: 408 527-0883
# CONTENTS

**Configuring Internet Key Exchange for IPsec**  
- Finding Feature Information  
- Prerequisites for IKE Configuration  
- Restrictions for IKE Configuration  
- Information About Configuring IKE for IPsec VPNs  
  - Supported Standards for Use with IKE  
  - IKE Benefits  
  - IKE Main Mode and Aggressive Mode  
  - IKE Policies Security Parameters for IKE Negotiation  
    - About IKE Policies  
    - IKE Peers Agreeing Upon a Matching IKE Policy  
  - IKE Authentication  
    - RSA Signatures  
    - RSA Encrypted Nonces  
    - Preshared Keys  
      - Preshared Key An Overview  
      - Setting ISAKMP Identity for Preshared Keys  
      - Mask Preshared Keys  
      - Disable Xauth on a Specific IPsec Peer  
  - IKE Mode Configuration  
  - How to Configure IKE for IPsec VPNs  
    - Creating IKE Policies  
      - Examples  
      - Troubleshooting Tips  
      - What to Do Next  
    - Configuring IKE Authentication  
      - Prerequisites  
      - Configuring RSA Keys Manually for RSA Encrypted Nonces  
      - Configuring Preshared Keys  
  - Troubleshooting Tips  
  - What to Do Next  

---

**Internet Key Exchange for IPsec VPNs Configuration Guide Cisco IOS XE Release 2**
Configuring IKE Mode Configuration 21
Configuration Examples for an IKE Configuration 23
  Example: Creating IKE Policies 23
    Creating 3DES IKE Policies Example 23
  Example: Creating an AES IKE Policy 24
  Example: Configuring IKE Authentication 24
Where to Go Next 25
Additional References 25
  Related Documents 25
  Standards 26
  MIBs 26
  RFCs 26
  Technical Assistance 27
Feature Information for Configuring Internet Key Exchange for IPsec VPNs 27
Glossary 29

Call Admission Control for IKE 31
  Finding Feature Information 31
  Prerequisites for Call Admission Control for IKE 31
  Information About Call Admission Control for IKE 31
    IKE Session 32
    Security Association Limit 32
      Limit on Number of In-Negotiation IKE Connections 32
    System Resource Usage 32
  How to Configure Call Admission Control for IKE 33
    Configuring the IKE Security Association Limit 33
    Configuring the System Resource Limit 34
    Verifying the Call Admission Control for IKE Configuration 35
  Configuration Examples for Call Admission Control for IKE 35
    Configuring the IKE Security Association Limit Example 36
    Configuring the System Resource Limit Example 36
  Additional References 36
  Feature Information for Call Admission Control for IKE 37

Certificate to ISAKMP Profile Mapping 39
  Finding Feature Information 39
  Prerequisites for Certificate to ISAKMP Profile Mapping 39
Configuring Internet Key Exchange for IPsec

This module describes how to configure the Internet Key Exchange (IKE) protocol for basic IP Security (IPsec) virtual private networks (VPNs). IKE 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, which implements the Oakley key exchange and Skeme key exchange inside the Internet Security Association Key Management Protocol (ISAKMP) framework. (ISAKMP, Oakley, and Skeme are security protocols implemented by IKE.)

- Finding Feature Information, page 1
- Prerequisites for IKE Configuration, page 1
- Restrictions for IKE Configuration, page 2
- Information About Configuring IKE for IPsec VPNs, page 2
- How to Configure IKE for IPsec VPNs, page 8
- Configuration Examples for an IKE Configuration, page 23
- Where to Go Next, page 25
- Additional References, page 25
- Feature Information for Configuring Internet Key Exchange for IPsec VPNs, page 27
- Glossary, page 29

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest feature information and caveats, see the release notes for your platform and software release. To find information about the features documented in this module, and to see a list of the releases in which each feature is supported, see the Feature Information Table at the end of this document.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to www.cisco.com/go/cfn. An account on Cisco.com is not required.

Prerequisites for IKE Configuration

- You should be familiar with the concepts and tasks explained in the module “Configuring Security for VPNs with IPsec.”
- Ensure that your access control lists (ACLs) are compatible with IKE. Because IKE negotiation uses User Datagram Protocol (UDP) on port 500, your ACLs must be configured so that UDP port 500
traffic is not blocked at interfaces used by IKE and IPsec. In some cases you might need to add a statement to your ACLs to explicitly permit UDP port 500 traffic.

Restrictions for IKE Configuration

The following restrictions are applicable when configuring IKE negotiation:

- The initiating router must not have a certificate associated with the remote peer.
- The preshared key must be a fully qualified domain name (FQDN) on both peers. (To configure the preshared key, enter the `crypto isakmp key` command.)
- The communicating routers must have a FQDN host entry for each other in their configurations.

The communicating routers must be configured to authenticate by hostname, not by IP address; thus, you should use the `crypto isakmp identity hostname` command.

SHA-2 for IPsec is not currently supported, though SHA-2 can be configured on IKE.

Information About Configuring IKE for IPsec VPNs

- Supported Standards for Use with IKE, page 2
- IKE Benefits, page 3
- IKE Main Mode and Aggressive Mode, page 4
- IKE Policies Security Parameters for IKE Negotiation, page 4
- IKE Authentication, page 5
- IKE Mode Configuration, page 8

Supported Standards for Use with IKE

Cisco implements the following standards:

- 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.
- Oakley-- A key exchange protocol that defines how to derive authenticated keying material.
- Skeme-- A key exchange protocol that defines how to derive authenticated keying material, with rapid key refreshment.

The component technologies implemented for use by IKE include the following:

- AES--Advanced Encryption Standard. A cryptographic algorithm that protects sensitive, unclassified information. AES is privacy transform for IPsec and IKE and has been developed to replace the DES. AES is designed to be more secure than DES: AES offers a larger key size, while ensuring that the only known approach to decrypt a message is for an intruder to try every possible key. AES has a
variable key length—the algorithm can specify a 128-bit key (the default), a 192-bit key, or a 256-bit key.

- DES—Data Encryption Standard. An algorithm that is used to encrypt packet data. IKE implements the 56-bit DES-CBC with Explicit IV standard. Cipher Block Chaining (CBC) requires an initialization vector (IV) to start encryption. The IV is explicitly given in the IPsec packet.

Cisco IOS XE software also implements Triple DES (168-bit) encryption, depending on the software versions available for a specific platform. Triple DES (3DES) is a strong form of encryption that allows sensitive information to be transmitted over untrusted networks. It enables customers, particularly in the finance industry, to utilize network-layer encryption.

Note

Cisco IOS XE images that have strong encryption (including, but not limited to, 56-bit data encryption feature sets) are subject to United States government export controls and have a limited distribution. Images that are to be installed outside the United States require an export license. Customer orders might be denied or subject to delay because of United States government regulations. Contact your sales representative or distributor for more information, or send e-mail to export@cisco.com.

- Diffie-Hellman-- A public-key cryptography protocol that allows two parties to establish a shared secret over an unsecure communications channel. Diffie-Hellman is used within IKE to establish session keys. 768-bit (the default), 1024-bit, and 1536-bit Diffie-Hellman groups are supported.
- MD5 (HMAC variant)-- Message Digest 5. A hash algorithm used to authenticate packet data. HMAC is a variant that provides an additional level of hashing.
- SHA (HMAC variant)-- Secure Hash Algorithm. A hash algorithm used to authenticate packet data. HMAC is a variant that provides an additional level of hashing.
- RSA signatures and RSA encrypted nonces-- RSA is the public key cryptographic system developed by Ron Rivest, Adi Shamir, and Leonard Adleman. RSA signatures provide nonrepudiation, and RSA encrypted nonces provide repudiation. (Repudiation and nonrepudiation have to do with traceability.)

IKE interoperates with the following standard:

X.509v3 certificates-- Used with the IKE protocol when authentication requires public keys. This certificate support allows the protected network to scale by providing the equivalent of a digital ID card to each device. When two devices intend 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).

IKE Benefits

IKE automatically negotiates IPsec security associations (SAs) and enables IPsec secure communications without costly manual preconfiguration. Specifically, IKE provides the following benefits:

- Eliminates the need to manually specify all the IPsec security parameters in the crypto maps at both peers.
- Allows you to specify a lifetime for the IPsec SA.
- Allows encryption keys to change during IPsec sessions.
- Allows IPsec to provide anti-replay services.
- Permits certification authority (CA) support for a manageable, scalable IPsec implementation.
- Allows dynamic authentication of peers.
IKE Main Mode and Aggressive Mode

IKE has two phases of key negotiation: phase 1 and phase 2. Phase 1 negotiates a security association (a key) between two IKE peers. The key negotiated in phase 1 enables IKE peers to communicate securely in phase 2. During phase 2 negotiation, IKE establishes keys (security associations) for other applications, such as IPsec.

Phase 1 negotiation can occur using main mode or aggressive mode. Main mode tries to protect all information during the negotiation, meaning that no information is available to a potential attacker. When main mode is used, the identities of the two IKE peers are hidden. Although this mode of operation is very secure, it is relatively costly in terms of the time it takes to complete the negotiation. Aggressive mode takes less time to negotiate keys between peers; however, it gives up some of the security provided by main mode negotiation. For example, the identities of the two parties trying to establish a security association are exposed to an eavesdropper.

The two modes serve different purposes and have different strengths. Main mode is slower than aggressive mode, but main mode is more secure and more flexible because it can offer an IKE peer more security proposals than aggressive mode. Aggressive mode is less flexible and not as secure, but much faster.

In Cisco IOS XE software, the two modes are not configurable. The default action for IKE authentication (rsa-sig, rsa-encr, or preshared) is to initiate main mode; however, in cases where there is no corresponding information to initiate authentication, and there is a preshared key associated with the hostname of the peer, Cisco IOS XE software can initiate aggressive mode. Cisco IOS XE software will respond in aggressive mode to an IKE peer that initiates aggressive mode.

IKE Policies Security Parameters for IKE Negotiation

An IKE policy defines a combination of security parameters to be used during the IKE negotiation. You must create an IKE policy at each peer participating in the IKE exchange.

If you do not configure any IKE policies, your router will use the default policy, which is always set to the lowest priority and which contains the default value of each parameter.

- About IKE Policies, page 4
- IKE Peers Agreeing Upon a Matching IKE Policy, page 5

About IKE Policies

Because IKE negotiations must be protected, each IKE negotiation begins by agreement of both peers on a common (shared) IKE policy. This policy states which security parameters will be used to protect subsequent IKE negotiations and mandates how the peers are authenticated.

After the two peers agree upon a policy, the security parameters of the policy are identified by an SA established at each peer, and these SAs apply to all subsequent IKE traffic during the negotiation.

You can configure multiple, prioritized policies on each peer--each with a different combination of parameter values. However, at least one of these policies must contain exactly the same encryption, hash, authentication, and Diffie-Hellman parameter values as one of the policies on the remote peer. For each policy that you create, you assign a unique priority (1 through 10,000, with 1 being the highest priority).
Tip
If you are interoperating with a device that supports only one of the values for a parameter, your choice is limited to the value supported by the other device. Aside from this limitation, there is often a trade-off between security and performance, and many of these parameter values represent such a trade-off. You should evaluate the level of security risks for your network and your tolerance for these risks.

IKE Peers Agreeing Upon a Matching IKE Policy

When the IKE negotiation begins, IKE searches for an IKE policy that is the same on both peers. The peer that initiates the negotiation will send all its policies to the remote peer, and the remote peer will try to find a match. The remote peer looks for a match by comparing its own highest priority policy against the policies received from the other peer. The remote peer checks each of its policies in order of its priority (highest priority first) until a match is found.

A match is made when both policies from the two peers contain the same encryption, hash, authentication, and Diffie-Hellman parameter values.

If a match is found, IKE will complete negotiation, and IPsec security associations will be created. If no acceptable match is found, IKE refuses negotiation and IPsec will not be established.

Note
Depending on which authentication method is specified in a policy, additional configuration might be required (as described in the section “IKE Authentication, page 5”). If a peer’s policy does not have the required companion configuration, the peer will not submit the policy when attempting to find a matching policy with the remote peer.

IKE Authentication

IKE authentication consists of the following options and each authentication method requires additional configuration.

- RSA Signatures, page 5
- RSA Encrypted Nonces, page 6
- Preshared Keys, page 6

RSA Signatures

With RSA signatures, you can configure the peers to obtain certificates from a CA. (The CA must be properly configured to issue the certificates.) Using a CA can dramatically improve the manageability and scalability of your IPsec network. Additionally, RSA signature-based authentication uses only two public key operations, whereas RSA encryption uses four public key operations, making it costlier in terms of overall performance. To properly configure CA support, see the chapter “Implementing and Managing a PKI.”

The certificates are used by each peer to exchange public keys securely. (RSA signatures requires that each peer has the public signature key of the remote peer.) When both peers have valid certificates, they will automatically exchange public keys with each other as part of any IKE negotiation in which RSA signatures are used.

You can also exchange the public keys manually, as described in the section “Configuring RSA Keys Manually for RSA Encrypted Nonces, page 13.”
RSA signatures provide nonrepudiation for the IKE negotiation. And you can prove to a third party after
the fact that you did indeed have an IKE negotiation with the remote peer.

**RSA Encrypted Nonces**

With RSA encrypted nonces, you must ensure that each peer has the public keys of the other peers.

Unlike RSA signatures, the RSA encrypted nonces method cannot use certificates to exchange public keys.
Instead, you ensure that each peer has the others’ public keys by one of the following methods:

- Manually configuring RSA keys as described in the section “Configuring RSA Keys Manually for
  RSA Encrypted Nonces, page 13.”

or

- Ensuring that an IKE exchange using RSA signatures with certificates has already occurred between
  the peers. (The peers’ public keys are exchanged during the RSA-signatures-based IKE negotiations if
  certificates are used.)

To ensure that the IKE exchange happens, specify two policies: a higher-priority policy with RSA
encrypted nonces and a lower-priority policy with RSA signatures. When IKE negotiations occur, RSA
signatures will be used the first time because the peers do not yet have each other’s public keys. Then
future IKE negotiations can use RSA encrypted nonces because the public keys will have been exchanged.

---

**Note**

This alternative requires that you already have CA support configured.

RSA encrypted nonces provide repudiation for the IKE negotiation; however, unlike RSA signatures, you
cannot prove to a third party that you had an IKE negotiation with the remote peer.

**Preshared Keys**

- Preshared Key An Overview, page 6
- Setting ISAKMP Identity for Preshared Keys, page 7
- Mask Preshared Keys, page 7
- Disable Xauth on a Specific IPsec Peer, page 7

**Preshared Key An Overview**

Preshared keys are clumsy to use if your secured network is large, and they do not scale well with a
growing network. However, they do not require use of a CA, as do RSA signatures, and might be easier to
set up in a small network with fewer than ten nodes. RSA signatures also can be considered more secure
when compared with preshared key authentication.

---

**Note**

If RSA encryption is configured and signature mode is negotiated (and certificates are used for signature
mode), the peer will request both signature and encryption keys. Basically, the router will request as many
keys as the configuration will support. If RSA encryption is not configured, it will just request a signature
key.
Setting ISAKMP Identity for Preshared Keys

You should set the ISAKMP identity for each peer that uses preshared keys in an IKE policy. When two peers use IKE to establish IPsec SAs, each peer sends its identity to the remote peer. Each peer sends either its host name or its IP address, depending on how you have set the ISAKMP identity of the router.

By default, a peer’s ISAKMP identity is the IP address of the peer. If appropriate, you could change the identity to be the peer's host name instead. As a general rule, set the identities of all peers the same way--either all peers should use their IP addresses or all peers should use their hostnames. If some peers use their host names and some peers use their IP addresses to identify themselves to each other, IKE negotiations could fail if the identity of a remote peer is not recognized and a DNS lookup is unable to resolve the identity.

Mask Preshared Keys

A mask preshared key allows a group of remote users with the same level of authentication to share an IKE preshared key. The preshared key of the remote peer must match the preshared key of the local peer for IKE authentication to occur.

A mask preshared key is usually distributed through a secure out-of-band channel. In a remote peer-to-local peer scenario, any remote peer with the IKE preshared key configured can establish IKE SAs with the local peer.

If you specify the `mask` keyword with the `crypto isakmp key` command, it is up to you to use a subnet address, which will allow more peers to share the same key. That is, the preshared key is no longer restricted to use between two users.

**Note** Using 0.0.0.0 as a subnet address is not recommended because it encourages group preshared keys, which allow all peers to have the same group key, thereby reducing the security of your user authentication.

Disable Xauth on a Specific IPsec Peer

Disabling Extended Authentication (Xauth) for static IPsec peers prevents the routers from being prompted for Xauth information--username and password.

Without the ability to disable Xauth, a user cannot select which peer on the same crypto map should use Xauth. That is, if a user has router-to-router IPsec on the same crypto map as a VPN-client-to-Cisco-IOS XE IPsec, both peers are prompted for a username and password. In addition, a remote static peer (a Cisco IOS XE router) cannot establish an IKE SA with the local Cisco IOS XE router. (Xauth is not an optional exchange, so if a peer does not respond to an Xauth request, the IKE SA is deleted.) Thus, the same interface cannot be used to terminate IPsec to VPN clients (that need Xauth) as well as other Cisco IOS XE routers (that cannot respond to Xauth) unless this feature is implemented.

**Note** Xauth can be disabled only if preshared keys are used as the authentication mechanism for the given crypto map.
IKE Mode Configuration

IKE mode configuration, as defined by the Internet Engineering Task Force (IETF), allows a gateway to download an IP address (and other network level configuration) to the client as part of an IKE negotiation. Using this exchange, the gateway gives IP addresses to the IKE client to be used as an “inner” IP address encapsulated under IPsec. This method provides a known IP address for the client that can be matched against IPsec policy.

To implement IPsec VPNs between remote access clients that have dynamic IP addresses and a corporate gateway, you have to dynamically administer scalable IPsec policy on the gateway after each client is authenticated. With IKE Mode Configuration, the gateway can set up scalable policy for a very large set of clients irrespective of the IP addresses of those clients.

There are two types of IKE Mode Configuration:

- Gateway initiation--Gateway initiates the configuration mode with the client. After the client responds, the IKE modifies the identity of the sender, the message is processed, and the client receives a response.
- Client initiation--Client initiates the configuration mode with the gateway. The gateway responds with an IP address that it has allocated for the client.

How to Configure IKE for IPsec VPNs

If you do not want IKE to be used with your IPsec implementation, you can disable it at all IPsec peers via the `no crypto isakmp` command, skip the rest of this chapter, and begin your IPsec VPN.

---

**Note**

If you disable IKE, you will have to manually specify all the IPsec SAs in the crypto maps at all peers, the IPsec SAs of the peers will never time out for a given IPsec session, the encryption keys will never change during IPsec sessions between the peers, anti-replay services will not be available between the peers, and public key infrastructure (PKI) support cannot be used.

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

Perform the following tasks to provide authentication of IPsec peers, negotiate IPsec SAs, and establish IPsec keys:

- Creating IKE Policies, page 8
- Configuring IKE Authentication, page 13
- Configuring IKE Mode Configuration, page 21

Creating IKE Policies

Perform this task to create an IKE policy.
If you are configuring an AES IKE policy, note the following restrictions:

- Your router must support IPsec and long keys (the “k9” subsystem).
- AES cannot encrypt IPsec and IKE traffic if an acceleration card is present.

**SUMMARY STEPS**

1. enable
2. configure terminal
3. crypto isakmp policy *priority*
4. encryption {des | 3des | aes| aes 192 aes 256}
5. hash {sha | md5}
6. authentication {rsa-sig | rsa-encr | pre-share}
7. group {1 | 2 | 5 | 14 | 15 | 16}
8. lifetime seconds
9. exit
10. exit
11. show crypto isakmp policy
12. --

**DETAILED STEPS**

<table>
<thead>
<tr>
<th>Command or Action</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Step 1</strong> enable</td>
<td>Enables privileged EXEC mode.</td>
</tr>
<tr>
<td>Example:</td>
<td>• Enter your password if prompted.</td>
</tr>
<tr>
<td>Router&gt; enable</td>
<td></td>
</tr>
<tr>
<td><strong>Step 2</strong> configure terminal</td>
<td>Enters global configuration mode.</td>
</tr>
<tr>
<td>Example:</td>
<td></td>
</tr>
<tr>
<td>Router# configure terminal</td>
<td></td>
</tr>
<tr>
<td><strong>Step 3</strong> crypto isakmp policy <em>priority</em></td>
<td>Defines an IKE policy and enters config-isakmp configuration mode.</td>
</tr>
<tr>
<td>Example:</td>
<td>• <em>priority</em> -- Uniquely identifies the IKE policy and assigns a priority to the policy. Valid values: 1 to 10,000; 1 is the highest priority.</td>
</tr>
<tr>
<td>Router(config)# crypto isakmp policy 10</td>
<td></td>
</tr>
</tbody>
</table>
### Command or Action

| Step 4 | encryption {des | 3des | aes | aes 192 | aes 256} |
|--------|---------------------------------|

**Example:**

```
Router(config-isakmp)# encryption aes 256
```

**Purpose**

Specifies the encryption algorithm.

By default, the `des` keyword is used.

- `des` -- 56-bit DES-CBC
- `3des` -- 168-bit DES
- `aes` -- 128-bit AES
- `aes 192` -- 192-bit AES
- `aes 256` -- 256-bit AES

| Step 5 | hash {sha | md5} |
|--------|-----------------|

**Example:**

```
Router(config-isakmp)# hash sha
```

**Purpose**

Specifies the hash algorithm.

By default, SHA-1 (`sha`) is the used.

**Note** MD5 has a smaller digest and is considered to be slightly faster than SHA-1.

| Step 6 | authentication {rsa-sig | rsa-encr | pre-share} |
|--------|---------------------------------|

**Example:**

```
Router(config-isakmp)# authentication pre-share
```

**Purpose**

Specifies the authentication method.

By default, RSA signatures are used.

- `rsa-sig` -- RSA signatures require that you configure your peer routers to obtain certificates from a CA.
- `rsa-encr` -- RSA encrypted nonces require that you ensure each peer has the other peer's RSA public keys.
- `pre-share` -- Preshared keys require that you separately configure these preshared keys.
### Command or Action

<table>
<thead>
<tr>
<th>Step</th>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
</table>
| Step 7 | `group {1 | 2 | 5 | 14 | 15 | 16}` | Specifies the Diffie-Hellman group identifier.  
By default, D-H group 1 is used.  
- 1 -- 768-bit Diffie-Hellman  
- 2 -- 1024-bit Diffie-Hellman  
- 5 -- 1536-bit Diffie-Hellman  
- 14 -- Specifies the 2048-bit Diffie-Hellman group.  
- 15 -- Specifies the 3072-bit Diffie-Hellman group.  
- 16 -- Specifies the 4096-bit Diffie-Hellman group.  

**Example:**  
Router(config-isakmp)# group 1  

**Note** The 1024-bit and 1536-bit Diffie-Hellman options are harder to “crack,” but require more CPU time to execute.  
**Note** Group 5 may be used for 128-bit keys, but group 14 is better.  
**Note** The group chosen must be strong enough (have enough bits) to protect the IPsec keys during negotiation. While there is some disagreement regarding how many bits are necessary in the Diffie-Hellman group to protect a specific key size, it is generally agreed that **group 14** is good protection for 128-bit keys, **group 15** is good protection for 192-bit keys, and **group 16** is good protection for 256-bit keys. |
| Step 8 | `lifetime seconds` | Specifies the lifetime of the IKE SA.  
- `seconds` -- Time, in seconds, before each SA expires. Valid values: 60 to 86,400 seconds; default value: 86,400.  

**Example:**  
Router(config-isakmp)# lifetime 180  

**Note** The shorter the lifetime (up to a point), the more secure your IKE negotiations will be. However, with longer lifetimes, future IPsec SAs can be set up more quickly. |
| Step 9 | `exit` | Exits config-isakmp configuration mode.  
**Example:**  
Router(config-isakmp)# exit |
| Step 10 | `exit` | Exits the global configuration mode.  
**Example:**  
Router(config)# exit |
| Step 11 | `show crypto isakmp policy` | (Optional) Displays all existing IKE policies.  
**Example:**  
Router# show crypto isakmp policy |
### Command or Action

**Step 12**

Repeat these steps for each policy you want to create.

### Note

These parameters apply to the IKE negotiations after the IKE SA is established.

- Examples, page 12
- Troubleshooting Tips, page 12
- What to Do Next, page 12

### Examples

The following sample output from the `show crypto isakmp policy` command displays a warning message after a user tries to configure an IKE encryption method that the hardware does not support:

```
Router# show crypto isakmp policy
Protection suite of priority 1
  WARNING:encryption hardware does not support the configured
  encryption method for ISAKMP policy 1
  hash algorithm: Secure Hash Standard
  authentication method: Pre-Shared Key
  Diffie-Hellman group: #1 (768 bit)
  lifetime: 3600 seconds, no volume limit
```

### Troubleshooting Tips

- Clear (and reinitialize) IPsec SAs by using the `clear crypto sa` in privileged EXEC command.

Using the `clear crypto sa` command without parameters will clear out the full SA database, which will clear out active security sessions. You may also specify the `peer`, `map`, or `entry` keywords to clear out only a subset of the SA database. For more information, see the `clear crypto sa` command in the Cisco IOS Security Command Reference.

- The default policy and default values for configured policies do not show up in the configuration when you issue the `show running-config` command. To see the default policy and any default values within configured policies, use the `show crypto isakmp policy` command.

- Any IPsec transforms or IKE encryption methods that the current hardware does not support should be disabled; they are ignored whenever an attempt to negotiate with the peer is made.

If a user enters an IPsec transform or an IKE encryption method that the hardware does not support, a warning message will be generated. These warning messages are also generated at boot time. When an encrypted card is inserted, the current configuration is scanned. If any IPsec transforms or IKE encryption methods are found that are not supported by the hardware, a warning message will be generated.

### What to Do Next

Depending on which authentication method you specified in your IKE policies (RSA signatures, RSA encrypted nonces, or preshared keys), you must do certain additional configuration tasks before IKE and IPsec can successfully use the IKE policies. For information on completing these additional tasks, refer to the following section “Configuring IKE Authentication, page 13.”
To configure an AES-based transform set, see the module “Configuring Security for VPNs with IPsec.”

**Configuring IKE Authentication**

After you have created at least one IKE policy in which you specified an authentication method (or accepted the default method), you need to configure an authentication method. IKE policies cannot be used by IPsec until the authentication method is successfully configured.

To configure IKE authentication, you should perform one of the following tasks, as appropriate:

- Configuring RSA Keys Manually for RSA Encrypted Nonces, page 13
- Configuring Preshared Keys, page 18
- Configuring RSA Keys to Obtain Certificates from a CA. For information on completing this task, see the module “Deploying RSA Keys Within a PKI.”
  - Prerequisites, page 13
  - Configuring RSA Keys Manually for RSA Encrypted Nonces, page 13
  - Configuring Preshared Keys, page 18

**Prerequisites**

You must have configured at least one IKE policy, which is where the authentication method was specified (or RSA signatures were accepted by default).

**Configuring RSA Keys Manually for RSA Encrypted Nonces**

**Note**

This task can be performed only if a CA is not in use.

To manually configure RSA keys, perform this task for each IPsec peer that uses RSA encrypted nonces in an IKE policy.
SUMMARY STEPS

1. enable
2. configure terminal
3. crypto key generate rsa [general-keys | usage-keys] [label key-label] [exportable] [modulus modulus-size]
4. exit
5. show crypto key mypubkey rsa
6. configure terminal
7. crypto key pubkey-chain rsa
8. Do one of the following:
   - named-key key-name [encryption | signature]
9. address ip-address
10. key-string key-string
11. quit
12. --
13. exit
14. exit
15. show crypto key pubkey-chain rsa [name key-name | address key-address]

DETAILED STEPS

<table>
<thead>
<tr>
<th>Command or Action</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>Step 1 enable</td>
<td>Enables privileged EXEC mode.</td>
</tr>
<tr>
<td></td>
<td>• Enter your password if prompted.</td>
</tr>
<tr>
<td>Example:</td>
<td>Router&gt; enable</td>
</tr>
<tr>
<td>Step 2 configure terminal</td>
<td>Enters global configuration mode.</td>
</tr>
<tr>
<td>Example:</td>
<td>Router# configure terminal</td>
</tr>
<tr>
<td>Step 3 crypto key generate rsa [general-keys</td>
<td>usage-keys] [label key-label] [exportable] [modulus modulus-size]</td>
</tr>
<tr>
<td></td>
<td>• If a key-label argument is not specified, the default value, which is the fully qualified domain name (FQDN) of the router, is used.</td>
</tr>
<tr>
<td>Example:</td>
<td>Router(config)# crypto key generate rsa general-keys modulus 360</td>
</tr>
<tr>
<td>Step 4</td>
<td>exit</td>
</tr>
<tr>
<td>---</td>
<td>---</td>
</tr>
<tr>
<td><strong>Example:</strong></td>
<td></td>
</tr>
<tr>
<td>Router(config) # exit</td>
<td></td>
</tr>
<tr>
<td><strong>Purpose:</strong></td>
<td>(Optional) Exits global configuration mode.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Step 5</th>
<th>show crypto key mypubkey rsa</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Example:</strong></td>
<td></td>
</tr>
<tr>
<td>Router# show crypto key mypubkey rsa</td>
<td></td>
</tr>
<tr>
<td><strong>Purpose:</strong></td>
<td>(Optional) Displays the generated RSA public keys.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Step 6</th>
<th>configure terminal</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Example:</strong></td>
<td></td>
</tr>
<tr>
<td>Router# configure terminal</td>
<td></td>
</tr>
<tr>
<td><strong>Purpose:</strong></td>
<td>Returns to global configuration mode.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Step 7</th>
<th>crypto key pubkey-chain rsa</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Example:</strong></td>
<td></td>
</tr>
<tr>
<td>Router(config) # crypto key pubkey-chain rsa</td>
<td></td>
</tr>
<tr>
<td><strong>Purpose:</strong></td>
<td>Enters public key configuration mode (so you can manually specify the RSA public keys of other devices).</td>
</tr>
</tbody>
</table>
### Step 8
Do one of the following:

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

**Example:**

```
Router(config-pubkey-chain)# named-key otherpeer.example.com
```

**Example:**

**Example:**

**Example:**

```
Router(config-pubkey-chain)# addressed-key 10.1.1.2 encryption
```

**Purpose:**
Indicates which remote peer’s RSA public key you are going to specify and enters public key configuration mode.

If the remote peer uses its host name as its ISAKMP identity, use the **named-key** command and specify the remote peer’s FQDN, such as somerouter.example.com, as the **key-name**.

If the remote peer uses its IP address as its ISAKMP identity, use the **addressed-key** command and specify the remote peer’s IP address as the **key-address**.

### Step 9
**address ip-address**

**Example:**

```
Router(config-pubkey-key)# address 10.5.5.1
```

**Purpose:**
Specifies the IP address of the remote peer.

If you use the **named-key** command, you need to use this command to specify the IP address of the peer.
<table>
<thead>
<tr>
<th>Command or Action</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Step 10</strong> key-string key-string</td>
<td>Specifies the RSA public key of the remote peer. (This key was previously viewed by the administrator of the remote peer when the RSA keys of the remote router were generated.)</td>
</tr>
<tr>
<td></td>
<td><strong>Example:</strong></td>
</tr>
<tr>
<td></td>
<td><code>Router(config-pubkey-key)# key-string</code></td>
</tr>
<tr>
<td></td>
<td><strong>Example:</strong></td>
</tr>
<tr>
<td></td>
<td><code>Router(config-pubkey)# 00302017 4A7D385B 1234EF29 335FC973</code></td>
</tr>
<tr>
<td></td>
<td><strong>Example:</strong></td>
</tr>
<tr>
<td></td>
<td><code>Router(config-pubkey)# 00302017 4A7D385B 1234EF29 335FC973</code></td>
</tr>
<tr>
<td></td>
<td><strong>Example:</strong></td>
</tr>
<tr>
<td></td>
<td><code>Router(config-pubkey)# 00302017 4A7D385B 1234EF29 335FC973</code></td>
</tr>
<tr>
<td></td>
<td><strong>Example:</strong></td>
</tr>
<tr>
<td></td>
<td><code>Router(config-pubkey)# 00302017 4A7D385B 1234EF29 335FC973</code></td>
</tr>
<tr>
<td></td>
<td><strong>Example:</strong></td>
</tr>
<tr>
<td></td>
<td><code>Router(config-pubkey)# 00302017 4A7D385B 1234EF29 335FC973</code></td>
</tr>
<tr>
<td></td>
<td><strong>Example:</strong></td>
</tr>
<tr>
<td></td>
<td><code>Router(config-pubkey)# 00302017 4A7D385B 1234EF29 335FC973</code></td>
</tr>
<tr>
<td></td>
<td><strong>Example:</strong></td>
</tr>
<tr>
<td></td>
<td><code>Router(config-pubkey)# 00302017 4A7D385B 1234EF29 335FC973</code></td>
</tr>
<tr>
<td></td>
<td><strong>Example:</strong></td>
</tr>
<tr>
<td></td>
<td><code>Router(config-pubkey)# 00302017 4A7D385B 1234EF29 335FC973</code></td>
</tr>
<tr>
<td></td>
<td><strong>Example:</strong></td>
</tr>
<tr>
<td></td>
<td><code>Router(config-pubkey)# 00302017 4A7D385B 1234EF29 335FC973</code></td>
</tr>
<tr>
<td></td>
<td><strong>Example:</strong></td>
</tr>
<tr>
<td></td>
<td><code>Router(config-pubkey)# 00302017 4A7D385B 1234EF29 335FC973</code></td>
</tr>
<tr>
<td></td>
<td><strong>Example:</strong></td>
</tr>
<tr>
<td></td>
<td><code>Router(config-pubkey)# 00302017 4A7D385B 1234EF29 335FC973</code></td>
</tr>
<tr>
<td></td>
<td><strong>Example:</strong></td>
</tr>
<tr>
<td></td>
<td><code>Router(config-pubkey)# 00302017 4A7D385B 1234EF29 335FC973</code></td>
</tr>
<tr>
<td></td>
<td><strong>Example:</strong></td>
</tr>
<tr>
<td></td>
<td><code>Router(config-pubkey)# 00302017 4A7D385B 1234EF29 335FC973</code></td>
</tr>
<tr>
<td></td>
<td><strong>Example:</strong></td>
</tr>
<tr>
<td></td>
<td><code>Router(config-pubkey)# 00302017 4A7D385B 1234EF29 335FC973</code></td>
</tr>
<tr>
<td></td>
<td><strong>Example:</strong></td>
</tr>
<tr>
<td></td>
<td><code>Router(config-pubkey)# 00302017 4A7D385B 1234EF29 335FC973</code></td>
</tr>
<tr>
<td></td>
<td><strong>Example:</strong></td>
</tr>
<tr>
<td></td>
<td><code>Router(config-pubkey)# 00302017 4A7D385B 1234EF29 335FC973</code></td>
</tr>
<tr>
<td><strong>Step 11</strong> Quit</td>
<td>Returns to public key chain configuration mode.</td>
</tr>
<tr>
<td></td>
<td><strong>Example:</strong></td>
</tr>
<tr>
<td></td>
<td><code>Router(config-pubkey-k)# quit</code></td>
</tr>
<tr>
<td><strong>Step 12</strong> --</td>
<td>Repeat these steps at each peer that uses RSA encrypted nonces in an IKE policy.</td>
</tr>
<tr>
<td></td>
<td><strong>Example:</strong></td>
</tr>
<tr>
<td></td>
<td><code>Router(config-pubkey-k)# quit</code></td>
</tr>
<tr>
<td>Command or Action</td>
<td>Purpose</td>
</tr>
<tr>
<td>------------------</td>
<td>---------</td>
</tr>
<tr>
<td><strong>Step 13</strong> exit</td>
<td>Returns to global configuration mode.</td>
</tr>
</tbody>
</table>

**Example:**

```
Router(config-pubkey-c)
# exit
```

<table>
<thead>
<tr>
<th><strong>Step 14</strong> exit</th>
<th>Returns to EXEC mode.</th>
</tr>
</thead>
</table>

**Example:**

```
Router(config)# exit
```

<table>
<thead>
<tr>
<th><strong>Step 15</strong> <code>show crypto key pubkey-chain rsa [name key-name] address key-address</code></th>
<th>(Optional) Displays either a list of all RSA public keys that are stored on your router or details of a particular RSA key that is stored on your router.</th>
</tr>
</thead>
</table>

**Example:**

```
Router# show crypto key pubkey-chain rsa
```

## Configuring Preshared Keys

To configure preshared keys, perform these steps for each peer that uses preshared keys in an IKE policy.

### Note

- Preshared keys do not scale well with a growing network.
- Mask preshared keys have the following restrictions:
  - The SA cannot be established between the IPsec peers until all IPsec peers are configured for the same preshared key.
  - The mask preshared key must be distinctly different for remote users requiring varying levels of authorization. You must configure a new preshared key for each level of trust and assign the correct keys to the correct parties. Otherwise, an untrusted party may obtain access to protected data.
### SUMMARY STEPS

1. `enable`
2. `configure terminal`
3. `crypto isakmp identity {address | hostname}
4. `ip host hostname address1 [address2...address8]
5. Do one of the following:
   - `crypto isakmp key keystring address peer-address [mask] [no-xauth]
6. Do one of the following:
   - `crypto isakmp key keystring address peer-address [mask] [no-xauth]
7. Repeat these steps at each peer that uses preshared keys in an IKE policy.

### DETAILED STEPS

<table>
<thead>
<tr>
<th>Command or Action</th>
<th>Purpose</th>
</tr>
</thead>
</table>
| **Step 1** enable | Enables privileged EXEC mode.  
| **Example:** | - Enter your password if prompted. |
| **Step 2** configure terminal | Enters global configuration mode. |
| **Example:** | |
| **Step 3** crypto isakmp identity {address | hostname} | Specifies the peer’s ISAKMP identity by IP address or by hostname at the local peer.  
| **Example:** | - *address* --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.  
| | - *hostname* --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). |
| **Step 4** ip host hostname address1 [address2...address8] | If the local peer’s ISAKMP identity was specified using a hostname, maps the peer’s host name to its IP address(es) at all the remote peers.  
| **Example:** | (This step might be unnecessary if the hostname or address is already mapped in a DNS server.) |
### Command or Action

**Step 5** Do one of the following:

- `crypto isakmp key keystring address peer-address [mask] [no-xauth]`

**Example:**

```
Router(config)# crypto isakmp key sharedkeystring address 192.168.1.33 no-xauth
```

- `crypto isakmp key keystring hostname [no-xauth]`

**Example:**

```
Router(config)# crypto isakmp key sharedkeystring hostname RemoteRouter.example.com
```

### Purpose

Specifies at the local peer the shared key to be used with a particular remote peer.

If the remote peer specified its ISAKMP identity with an address, use the `address` keyword in this step; otherwise use the `hostname` keyword in this step.

- `no-xauth`-- Prevents the router from prompting the peer for Xauth information. Use this keyword if router-to-router IPsec is on the same crypto map as VPN-client-to-Cisco IOS XE IPsec.

**Note** According to the design of preshared key authentication in IKE main mode, preshared keys must be based on the IP address of the peers. Although you can send the hostname as the identity of preshared key authentication, the key is searched on the IP address of the peer; if the key is not found (based on the IP address) the negotiation will fail.
<table>
<thead>
<tr>
<th>Command or Action</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Step 6</strong> Do one of the following:</td>
<td>Specifies at the remote peer the shared key to be used with the local peer.</td>
</tr>
<tr>
<td>· <code>crypto isakmp key keystring address peer-address [mask] [no-xauth]</code></td>
<td>This is the same key you just specified at the local peer.</td>
</tr>
<tr>
<td>Example:</td>
<td>If the local peer specified its ISAKMP identity with an address, use the <code>address</code> keyword in this step; otherwise use the <code>hostname</code> keyword in this step.</td>
</tr>
<tr>
<td><code>Router(config) crypto isakmp key sharedkeystring address 10.0.0.1</code></td>
<td></td>
</tr>
<tr>
<td><strong>Step 7</strong> Repeat these steps at each peer that uses preshared keys in an IKE policy.</td>
<td>--</td>
</tr>
</tbody>
</table>

**Configuring IKE Mode Configuration**

Perform this task to configure IKE mode configuration.
IKE Mode Configuration has the following restrictions:

- Interfaces with crypto maps that are configured for IKE Mode Configuration may experience a slightly longer connection setup time, which is true even for IKE peers that refuse to be configured or do not respond to the configuration mode request. In both cases, the gateway initiates the configuration of the client.
- This feature was not designed to enable the configuration mode for every IKE connection by default. Configure this feature at the global crypto map level.

**SUMMARY STEPS**

1. `enable`
2. `configure terminal`
3. `ip local pool pool-name start-addr end-addr`
4. `crypto isakmp client configuration address-pool local pool-name`
5. `crypto map tag client configuration address [initiate | respond]`

**DETAILED STEPS**

<table>
<thead>
<tr>
<th>Command or Action</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Step 1</strong> enable</td>
<td>Enables privileged EXEC mode.</td>
</tr>
<tr>
<td>Example:</td>
<td>- Enter your password if prompted.</td>
</tr>
<tr>
<td>Router&gt; enable</td>
<td></td>
</tr>
<tr>
<td><strong>Step 2</strong> configure terminal</td>
<td>Enters global configuration mode.</td>
</tr>
<tr>
<td>Example:</td>
<td></td>
</tr>
<tr>
<td>Router# configure terminal</td>
<td></td>
</tr>
<tr>
<td><strong>Step 3</strong> <code>ip local pool pool-name start-addr end-addr</code></td>
<td>Defines an existing local address pool that defines a set of addresses.</td>
</tr>
<tr>
<td>Example:</td>
<td></td>
</tr>
<tr>
<td>Router(config)# ip local pool ire 172.16.23.0 172.16.23.255</td>
<td></td>
</tr>
<tr>
<td><strong>Step 4</strong> crypto isakmp client configuration address-pool local pool-name</td>
<td>References the local address pool in the IKE configuration.</td>
</tr>
<tr>
<td>Example:</td>
<td></td>
</tr>
<tr>
<td>Router(config)# crypto isakmp client configuration address-pool local ire</td>
<td></td>
</tr>
</tbody>
</table>
Step 5:  
**crypto map tag client configuration address [initiate | respond]**

**Purpose:** Configures IKE Mode Configuration in global crypto map configuration mode.

**Example:**

```bash
Router(config)# crypto map dyn client configuration address initiate
```

---

**Configuration Examples for an IKE Configuration**

- Example: Creating IKE Policies, page 23
- Example: Configuring IKE Authentication, page 24

**Example: Creating IKE Policies**

This section contains the following examples, which show how to configure a 3DES IKE policy and an AES IKE policy:

- Creating 3DES IKE Policies Example, page 23
- Example: Creating an AES IKE Policy, page 24

**Creating 3DES IKE Policies Example**

This example creates two IKE policies, with policy 15 as the highest priority, policy 20 as the next priority, and the existing default priority as the lowest priority. It also creates a preshared key to be used with policy 20 with the remote peer whose IP address is 192.168.224.33.

```bash
crypto isakmp policy 15
    encryption 3des
    hash md5
    authentication rsa-sig
    group 2
    lifetime 5000
! crypto isakmp policy 20
    authentication pre-share
    lifetime 10000
! crypto isakmp key 1234567890 address 192.168.224.33
```

In the example, the encryption des of policy 15 would not appear in the written configuration because this is the default value for the encryption algorithm parameter.

If the `show crypto isakmp policy` command is issued with this configuration, the output is as follows:

```
Protection suite priority 15
    encryption algorithm: 3DES - Triple Data Encryption Standard (168 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 that although the output shows “no volume limit” for the lifetimes, you can configure only a time lifetime (such as 86,400 seconds); volume-limit lifetimes are not configurable.

Example: Creating an AES IKE Policy

The following example is sample output from the `show running-config` command. In this example, the AES 256-bit key is enabled.

Current configuration : 1665 bytes
!
version 12.2
service timestamps debug datetime msec
service timestamps log datetime msec
no service password-encryption
!
hostname "Router1"
!
ip subnet-zero
!
no ip domain lookup
!
ip audit notify log
ip audit po max-events 100
!
crypto isakmp policy 10
  encryption aes 256
  authentication pre-share
  lifetime 180
crypto isakmp key cisco123 address 10.0.110.1
!
crypto ipsec transform-set aesset esp-aes 256 esp-sha-hmac
  mode transport
!
crypto map aesmap 10 ipsec-isakmp
  set peer 10.0.110.1
  set transform-set aesset
  match address 120
!
.
.
.

Example: Configuring IKE Authentication

The following example shows how to manually specify the RSA public keys of two IPsec peer-- the peer at 10.5.5.1 uses general-purpose keys, and the other peer uses special-usage keys:

crypto key pubkey-chain rsa
  named-key otherpeer.example.com
  address 10.5.5.1
  key-string
  005C300D 06092A86 4886F70D 01010105 00034B00 30480241 00C5E23B 55D6AB22
Where to Go Next

After you have successfully configured IKE negotiation, you can begin configuring IPsec. For information on completing these tasks, see the module “Configuring Security for VPNs With IPsec.”

Additional References

The following sections provide references related to configuring IKE for IPsec VPNs.

- Related Documents, page 25
- Standards, page 26
- MIBs, page 26
- RFCs, page 26
- Technical Assistance, page 27

Related Documents

<table>
<thead>
<tr>
<th>Related Topic</th>
<th>Document Title</th>
</tr>
</thead>
<tbody>
<tr>
<td>Configuring RSA keys to obtain certificates from a CA</td>
<td>“Deploying RSA Keys Within a PKI” module in the Cisco IOS XE Security Configuration Guide: Secure Connectivity</td>
</tr>
</tbody>
</table>
Standards

<table>
<thead>
<tr>
<th>Standards</th>
<th>Title</th>
</tr>
</thead>
<tbody>
<tr>
<td>No new or modified standards are supported by this feature, and support for existing standards has not been modified by this feature.</td>
<td>--</td>
</tr>
</tbody>
</table>

MIBs

<table>
<thead>
<tr>
<th>MIBs</th>
<th>MIBs Link</th>
</tr>
</thead>
<tbody>
<tr>
<td>No new or modified MIBs are supported by this feature, and support for existing MIBs has not been modified by this feature.</td>
<td>To locate and download MIBs for selected platforms, Cisco IOS XE releases, and feature sets, use Cisco MIB Locator found at the following URL: <a href="http://www.cisco.com/go/mibs">http://www.cisco.com/go/mibs</a></td>
</tr>
</tbody>
</table>

RFCs

<table>
<thead>
<tr>
<th>RFCs</th>
<th>Title</th>
</tr>
</thead>
<tbody>
<tr>
<td>RFC 2408</td>
<td>Internet Security Association and Key Management Protocol (ISAKMP)</td>
</tr>
<tr>
<td>RFC 2409</td>
<td>The Internet Key Exchange (IKE)</td>
</tr>
<tr>
<td>RFC 2412</td>
<td>The OAKLEY Key Determination Protocol</td>
</tr>
</tbody>
</table>
Technical Assistance

<table>
<thead>
<tr>
<th>Description</th>
<th>Link</th>
</tr>
</thead>
<tbody>
<tr>
<td>The Cisco Support website provides extensive online resources, including documentation and tools for troubleshooting and resolving technical issues with Cisco products and technologies. To receive security and technical information about your products, you can subscribe to various services, such as the Product Alert Tool (accessed from Field Notices), the Cisco Technical Services Newsletter, and Really Simple Syndication (RSS) Feeds. Access to most tools on the Cisco Support website requires a Cisco.com user ID and password.</td>
<td><a href="http://www.cisco.com/cisco/web/support/index.html">http://www.cisco.com/cisco/web/support/index.html</a></td>
</tr>
</tbody>
</table>

Feature Information for Configuring Internet Key Exchange for IPsec VPNs

The following table provides release information about the feature or features described in this module. This table lists only the software release that introduced support for a given feature in a given software release train. Unless noted otherwise, subsequent releases of that software release train also support that feature.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to [www.cisco.com/go/cfn](http://www.cisco.com/go/cfn). An account on Cisco.com is not required.
### Table 1  Feature Information for Configuring Internet Key Exchange for IPsec VPNs

<table>
<thead>
<tr>
<th>Feature Name</th>
<th>Releases</th>
<th>Feature Information</th>
</tr>
</thead>
<tbody>
<tr>
<td>Ability to Disable Extended Authentication for Static IPsec Peers</td>
<td>Cisco IOS XE Release 2.1</td>
<td>This feature allows a user to disable Xauth while configuring the preshared key for router-to-router IPsec. Thus, the router will not prompt the peer for a username and password, which are transmitted when Xauth occurs for VPN-client-to-Cisco-IOS XE IPsec. The following section provides information about this feature: Configuring Preshared Keys task_1057066 The following command was modified by this feature: <code>crypto isakmp key</code></td>
</tr>
</tbody>
</table>
| Advanced Encryption Standard (AES)                | Cisco IOS XE Release 2.1 | This feature adds support for the new encryption standard AES, which is a privacy transform for IPsec and IKE and has been developed to replace DES. The following sections provide information about this feature:  
  - Supported Standards for Use with IKE  
  - Creating IKE Policies task  
  The following commands were modified by this feature: `crypto ipsec transform-set, encryption (IKE policy), show crypto isakmp policy, show crypto ipsec transform-set` |
<p>| IKE Extended Authentication (Xauth)               | Cisco IOS XE Release 2.1 | Xauth allows all Cisco IOS XE software AAA authentication methods to perform user authentication in a separate phase after the IKE authentication Phase 1 exchange.                                                                                                                                                                                                                  |</p>
<table>
<thead>
<tr>
<th>Feature Name</th>
<th>Releases</th>
<th>Feature Information</th>
</tr>
</thead>
<tbody>
<tr>
<td>Wildcard Pre-Shared Key</td>
<td>Cisco IOS XE Release 2.1</td>
<td>A wildcard preshared key allows a group of remote users with the same level of authentication to share an IKE preshared key. The remote peer’s preshared key must match the local peer’s preshared key for IKE authentication to occur. The term wildcard means that any remote peer with the preshared key can access the local peer, regardless of the remote peer’s IP address assignment. The term preshared key is a shared secret key exchanged during IKE negotiation.</td>
</tr>
<tr>
<td>IKE - Diffie-Hellman (768 Bit or 1024 Bit) PKCS #3 Support</td>
<td>Cisco IOS XE Release 2.1</td>
<td>This feature provides support for Diffie-Hellman PKCS #3 (768 or 1024 bit).</td>
</tr>
<tr>
<td>IKE Phase 1 Main Mode and Phase 1 Aggressive Mode</td>
<td>Cisco IOS XE Release 2.1</td>
<td>This feature provides support for IKE aggressive mode negotiation using tunnel attributes.</td>
</tr>
<tr>
<td>IKE - RSA Signature</td>
<td>Cisco IOS XE Release 2.1</td>
<td>This feature provides RSA signature support.</td>
</tr>
</tbody>
</table>

**Glossary**

- **anti-replay** -- Security service in which the receiver can reject old or duplicate packets to protect itself against replay attacks. IPsec provides optional anti-replay services by use of a sequence number and the use of authentication.

- **data authentication** -- Verification of the integrity and origin of the data.
  Data authentication can refer either to integrity alone or to both of these concepts (although data origin authentication is dependent upon data integrity).

- **peer** -- In the context of this chapter, a “peer” is a router or other device that participates in IPsec and IKE.

- **PFS** -- Perfect forward secrecy. Cryptographic characteristic associated with a derived shared secret value. With PFS, if one key is compromised, previous and subsequent keys are not also compromised, because subsequent keys are not derived from previous keys.

- **repudiation** -- Quality that prevents a third party from being able to prove that a communication between two other parties ever took place. Repudiation is a desirable quality if you do not want your communications to be traceable.

- **nonrepudiation** -- Quality that allows a third party to prove that a communication between two other parties took place. Nonrepudiation is desirable if you want to be able to trace your communications and prove that they occurred.
SA --security association. How two or more entities utilize security services to communicate securely.

For example, an IPsec SA defines the encryption algorithm (if used), the authentication algorithm, and the shared session key to be used during the IPsec connection. Both IPsec and IKE require and use SAs to identify the parameters of their connections. IKE can negotiate and establish its own SA. The IPsec SA is established either by IKE or by manual user configuration.

Refer to *Internetworking Terms and Acronyms* for terms not included in this glossary.

Cisco and the Cisco logo are trademarks or registered trademarks of Cisco and/or its affiliates in the U.S. and other countries. To view a list of Cisco trademarks, go to this URL: [www.cisco.com/go/trademarks](http://www.cisco.com/go/trademarks). Third-party trademarks mentioned are the property of their respective owners. The use of the word partner does not imply a partnership relationship between Cisco and any other company. (1110R)

Any Internet Protocol (IP) addresses and phone numbers used in this document are not intended to be actual addresses and phone numbers. Any examples, command display output, network topology diagrams, and other figures included in the document are shown for illustrative purposes only. Any use of actual IP addresses or phone numbers in illustrative content is unintentional and coincidental.
Call Admission Control for IKE

The Call Admission Control for IKE feature describes the application of Call Admission Control (CAC) to the Internet Key Exchange (IKE) protocol in Cisco IOS XE. CAC limits the number of simultaneous IKE security associations (SAs) (that is, calls to CAC) that a router can establish.

- Finding Feature Information, page 31
- Prerequisites for Call Admission Control for IKE, page 31
- Information About Call Admission Control for IKE, page 31
- How to Configure Call Admission Control for IKE, page 33
- Configuration Examples for Call Admission Control for IKE, page 35
- Additional References, page 36
- Feature Information for Call Admission Control for IKE, page 37

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest feature information and caveats, see the release notes for your platform and software release. To find information about the features documented in this module, and to see a list of the releases in which each feature is supported, see the Feature Information Table at the end of this document.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to www.cisco.com/go/cfn. An account on Cisco.com is not required.

Prerequisites for Call Admission Control for IKE

- Configure IKE on the router.

Information About Call Admission Control for IKE

- IKE Session, page 32
- Security Association Limit, page 32
- System Resource Usage, page 32
IKE Session

There are two ways to limit the number of IKE SAs that a router can establish to or from another router:

- Configure the absolute IKE SA limit by entering the crypto call admission limit command. The router drops new IKE SA requests when the value has been reached.
- Configure the system resource limit by entering the call admission limit command. The router drops new IKE SA requests when the level of system resources that are configured in the unit of charge is being used.

CAC is applied only to new SAs (that is, when an SA does not already exist between the peers). Every effort is made to preserve existing SAs. Only new SA requests will ever be denied due to a lack of system resources or because the configured IKE SA limit has been reached.

Security Association Limit

An SA is a description of how two or more entities will utilize security services to communicate securely on behalf of a particular data flow. IKE requires and uses SAs to identify the parameters of its connections. IKE can negotiate and establish its own SA. An IKE SA is used by IKE only, and it is bidirectional. An IKE SA cannot limit IPsec.

IKE drops SA requests based on a user-configured SA limit. To configure an IKE SA limit, enter the crypto call admission limit command. When there is a new SA request from a peer router, IKE determines if the number of active IKE SAs plus the number of SAs being negotiated meets or exceeds the configured SA limit. If the number is greater than or equal to the limit, the new SA request is rejected and a syslog is generated. This log contains the source destination IP address of the SA request.

- Limit on Number of In-Negotiation IKE Connections, page 32

Limit on Number of In-Negotiation IKE Connections

A limit on the number of in-negotiation IKE connections can be configured. This type of IKE connection represents either an aggressive mode IKE SA or a main mode IKE SA prior to its authentication and actual establishment.

Using the crypto call admission limit ike in-negotiation-sa {number} command allows the configured number of in-negotiation IKE SAs to start negotiation without contributing to the maximum number of IKE SAs allowed.

System Resource Usage

CAC polls a global resource monitor so that IKE knows when the router is running short of CPU cycles or memory buffers. You can configure a limit, in the range to 100000, that represents the level of system resource usage in system resource usage units. When that level of resources is being used, IKE drops (will not accept new) SA requests. To configure the system resource usage limit, enter the call admission limit command.

For each incoming new SA request, the current load on the router is converted into a numerical value, representing the system resource usage level, and is compared to the resource limit set by the call admission limit command. If the current load is more than the configured resource limit, IKE drops the new SA request. Load on the router includes active SAs, CPU usage, and SA requests being considered.

The call admission load command configures a multiplier value from 0 to 1000 that represents a scaling factor for current system resource usage and a load metric poll rate of 1 to 32 seconds. The numerical value
for the system resource usage level is calculated by the formula (scaling factor * current system resource usage) / 100. It is recommended that the call admission load command not be used unless advised by a Cisco Technical Assistance Center (TAC) engineer.

**How to Configure Call Admission Control for IKE**

- Configuring the IKE Security Association Limit, page 33
- Configuring the System Resource Limit, page 34
- Verifying the Call Admission Control for IKE Configuration, page 35

**Configuring the IKE Security Association Limit**

Perform this task to configure the absolute IKE SA limit. The router drops new IKE SA requests when the limit has been reached.

**SUMMARY STEPS**

1. enable
2. configure terminal
3. crypto call admission limit {ike{in-negotiation-sa|sa number}}
4. exit

**DETAILED STEPS**

<table>
<thead>
<tr>
<th>Command or Action</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>Step 1 enable</td>
<td>Enables privileged EXEC mode.</td>
</tr>
<tr>
<td></td>
<td>• Enter your password if prompted.</td>
</tr>
<tr>
<td>Example:</td>
<td>Router&gt; enable</td>
</tr>
<tr>
<td>Step 2 configure terminal</td>
<td>Enters global configuration mode.</td>
</tr>
<tr>
<td>Example:</td>
<td>Router# configure terminal</td>
</tr>
<tr>
<td>Step 3 crypto call admission limit {ike{in-negotiation-sa</td>
<td>sa number}}</td>
</tr>
<tr>
<td>Example:</td>
<td>Router(config)# crypto call admission limit ike sa 25</td>
</tr>
</tbody>
</table>
### Configuring the System Resource Limit

Perform this task to configure the system resource limit. The router drops new IKE SA requests when the level of system resources that are configured in the unit of charge is being used.

**SUMMARY STEPS**

1. `enable`
2. `configure terminal`
3. `call admission limit charge`
4. `exit`

**DETAILED STEPS**

<table>
<thead>
<tr>
<th>Command or Action</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Step 1 enable</strong></td>
<td>Enables privileged EXEC mode.</td>
</tr>
<tr>
<td></td>
<td>· Enter your password if prompted.</td>
</tr>
<tr>
<td><strong>Example:</strong></td>
<td></td>
</tr>
<tr>
<td><code>Router&gt; enable</code></td>
<td></td>
</tr>
<tr>
<td><strong>Step 2 configure terminal</strong></td>
<td>Enters global configuration mode.</td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>Example:</strong></td>
<td></td>
</tr>
<tr>
<td><code>Router# configure terminal</code></td>
<td></td>
</tr>
<tr>
<td><strong>Step 3 call admission limit charge</strong></td>
<td>Sets the level of the system resources that, when used, causes IKE to stop accepting new SA requests.</td>
</tr>
<tr>
<td></td>
<td>· charge --Valid values are 1 to 100000.</td>
</tr>
<tr>
<td><strong>Example:</strong></td>
<td></td>
</tr>
<tr>
<td><code>Router(config)# call admission limit 1000</code></td>
<td></td>
</tr>
<tr>
<td><strong>Step 4 exit</strong></td>
<td>Returns to privileged EXEC mode.</td>
</tr>
<tr>
<td><strong>Example:</strong></td>
<td></td>
</tr>
<tr>
<td><code>Router(config)# exit</code></td>
<td></td>
</tr>
</tbody>
</table>
Verifying the Call Admission Control for IKE Configuration

To verify the CAC for IKE configuration, perform the following steps.

SUMMARY STEPS

1. show call admission statistics
2. show crypto call admission statistics

DETAILED STEPS

**Step 1**

**show call admission statistics**

Use this command to monitor the global CAC configuration parameters and the behavior of CAC.

**Example:**

Router# show call admission statistics
Total Call admission charges: 82, limit 1000
Total calls rejected 1430, accepted 0
Load metric: charge 82, unscaled 82%

**Step 2**

**show crypto call admission statistics**

Use this command to monitor crypto CAC statistics.

**Example:**

Router# show crypto call admission statistics

---

Configuration Examples for Call Admission Control for IKE

- Configuring the IKE Security Association Limit Example, page 36
- Configuring the System Resource Limit Example, page 36
Configuring the IKE Security Association Limit Example

The following example shows how to specify that there can be a maximum of 25 SAs before IKE starts rejecting new SA requests:

```
Router(config)# crypto call admission limit ike sa 25
```

Configuring the System Resource Limit Example

The following example shows how to specify that IKE should drop SA requests when the level of system resources that are configured in the unit of charge reaches 9000:

```
Router(config)# call admission limit 9000
```

Additional References

The following sections provide references related to Call Admission Control for IKE.

### Related Documents

<table>
<thead>
<tr>
<th>Related Topic</th>
<th>Document Title</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cisco IOS commands</td>
<td>Cisco IOS Master Commands List, All Releases</td>
</tr>
<tr>
<td>Configuring IKE</td>
<td>Configuring Internet Key Exchange for IPsec VPNs</td>
</tr>
<tr>
<td>IKE commands</td>
<td>Cisco IOS Security Command Reference</td>
</tr>
</tbody>
</table>

### Standards

<table>
<thead>
<tr>
<th>Standards</th>
<th>Title</th>
</tr>
</thead>
<tbody>
<tr>
<td>None</td>
<td>--</td>
</tr>
</tbody>
</table>

### MIBs

<table>
<thead>
<tr>
<th>MIBs</th>
<th>MIBs Link</th>
</tr>
</thead>
<tbody>
<tr>
<td>No new or modified MIBs are supported by this feature, and support for existing MIBs has not been modified by this feature.</td>
<td>To locate and download MIBs for selected platforms, Cisco IOS XE software releases, and feature sets, use Cisco MIB Locator found at the following URL: <a href="http://www.cisco.com/go/mibs">http://www.cisco.com/go/mibs</a></td>
</tr>
</tbody>
</table>
RFCs

<table>
<thead>
<tr>
<th>RFCs</th>
<th>Title</th>
</tr>
</thead>
<tbody>
<tr>
<td>RFC 2409</td>
<td><em>The Internet Key Exchange</em></td>
</tr>
</tbody>
</table>

Technical Assistance

<table>
<thead>
<tr>
<th>Description</th>
<th>Link</th>
</tr>
</thead>
<tbody>
<tr>
<td>The Cisco Support and Documentation website provides online resources to download documentation, software, and tools. Use these resources to install and configure the software and to troubleshoot and resolve technical issues with Cisco products and technologies. Access to most tools on the Cisco Support and Documentation website requires a Cisco.com user ID and password.</td>
<td><a href="http://www.cisco.com/cisco/web/support/index.html">http://www.cisco.com/cisco/web/support/index.html</a></td>
</tr>
</tbody>
</table>

**Feature Information for Call Admission Control for IKE**

The following table provides release information about the feature or features described in this module. This table lists only the software release that introduced support for a given feature in a given software release train. Unless noted otherwise, subsequent releases of that software release train also support that feature.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to [www.cisco.com/go/cfn](http://www.cisco.com/go/cfn). An account on Cisco.com is not required.
Table 2  Feature Information for Call Admission Control for IKE

<table>
<thead>
<tr>
<th>Feature Name</th>
<th>Releases</th>
<th>Feature Information</th>
</tr>
</thead>
<tbody>
<tr>
<td>Call Admission Control for IKE</td>
<td>Cisco IOS XE Release 2.1</td>
<td>The Call Admission Control for IKE feature describes the application of Call Admission Control (CAC) to the Internet Key Exchange (IKE) protocol in Cisco IOS XE. CAC limits the number of simultaneous IKE security associations (SAs) (that is, calls to CAC) that a router can establish. The following commands were introduced or modified: <code>call admission limit</code>, <code>clear crypto call admission statistics</code>, <code>crypto call admission limit</code>, <code>show call admission statistics</code>, <code>show crypto call admission statistics</code>.</td>
</tr>
<tr>
<td>Ability to Configure a Limit on the Number of In-negotiation IKE Connections</td>
<td>Cisco IOS XE Release 2.1</td>
<td>This feature was implemented on the Cisco ASR 1000 Series Routers. The following command was introduced or modified: <code>crypto call admission limit</code>.</td>
</tr>
</tbody>
</table>

Cisco and the Cisco logo are trademarks or registered trademarks of Cisco and/or its affiliates in the U.S. and other countries. To view a list of Cisco trademarks, go to this URL: [www.cisco.com/go/trademarks](http://www.cisco.com/go/trademarks). Third-party trademarks mentioned are the property of their respective owners. The use of the word partner does not imply a partnership relationship between Cisco and any other company. (1110R)

Any Internet Protocol (IP) addresses and phone numbers used in this document are not intended to be actual addresses and phone numbers. Any examples, command display output, network topology diagrams, and other figures included in the document are shown for illustrative purposes only. Any use of actual IP addresses or phone numbers in illustrative content is unintentional and coincidental.
The Certificate to ISAKMP Profile Mapping feature enables you to assign an Internet Security Association and Key Management Protocol (ISAKMP) profile to a peer on the basis of the contents of arbitrary fields in the certificate. In addition, this feature allows you to assign a group name to those peers that are assigned an ISAKMP profile.

- Finding Feature Information, page 39
- Prerequisites for Certificate to ISAKMP Profile Mapping, page 39
- Restrictions for Certificate to ISAKMP Profile Mapping, page 39
- Information About Certificate to ISAKMP Profile Mapping, page 40
- How to Configure Certificate to ISAKMP Profile Mapping, page 41
- Configuration Examples for Certificate to ISAKMP Profile Mapping, page 44
- Additional References, page 47
- Feature Information for Certificate to ISAKMP Profile Mapping, page 48

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest feature information and caveats, see the release notes for your platform and software release. To find information about the features documented in this module, and to see a list of the releases in which each feature is supported, see the Feature Information Table at the end of this document.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to www.cisco.com/go/cfn. An account on Cisco.com is not required.

Prerequisites for Certificate to ISAKMP Profile Mapping

- You should be familiar with configuring certificate maps.
- You should be familiar with configuring ISAKMP profiles.

Restrictions for Certificate to ISAKMP Profile Mapping

- This feature will not be applicable if you use Rivest, Shamir, and Adelman- (RSA-) signature or RSA-encryption authentication without certificate exchange. ISAKMP peers must be configured to do RSA-signature or RSA-encryption authentication using certificates.
- When there are two or more ISAKMP profiles, each having a different trustpoint enrolled in the same certificate authority server, the responder selects the last global trustpoint. This is because trustpoints
are selected in the reverse order in which they are defined globally. For the IPsec tunnel establishment to be successful for peers, the trustpoint selected by the initiator should match the trustpoint selected by the responder. All other IPsec tunnels will fail to establish connection if the trustpoints do not match.

Information About Certificate to ISAKMP Profile Mapping

- Certificate to ISAKMP Profile Mapping Overview, page 40
- How Certificate to ISAKMP Profile Mapping Works, page 40
- Assigning an ISAKMP Profile and Group Name to a Peer, page 41

Certificate to ISAKMP Profile Mapping Overview

Prior to Cisco IOS XE Release 2.1, the only way to map a peer to an ISAKMP profile was as follows. The ISAKMP identity field in the ISAKMP exchange was used for mapping a peer to an ISAKMP profile. When certificates were used for authentication, the ISAKMP identity payload contained the subject name from the certificate. If a certificate authority (CA) did not provide the required group value in the first Organizational Unit (OU) field of a certificate, an ISAKMP profile could not be assigned to a peer.

Effective with Cisco IOS XE Release 2.1, a peer can still be mapped as explained above. However, the Certificate to ISAKMP Profile Mapping feature enables you to assign an ISAKMP profile to a peer on the basis of the contents of arbitrary fields in the certificate. You are no longer limited to assigning an ISAKMP profile on the basis of the subject name of the certificate. In addition, this feature allows you to assign a group to a peer to which an ISAKMP profile has been assigned.

How Certificate to ISAKMP Profile Mapping Works

The figure below illustrates how certificate maps may be attached to ISAKMP profiles and assigned group names.

**Figure 1**  Certificate Maps Mapped for Profile Group Assignment

![Diagram showing certificate maps mapped for profile group assignment](image-url)
A certificate map can be attached to only one ISAKMP profile although an ISAKMP profile can have several certificate maps attached to it.

Certificate maps provide the ability for a certificate to be matched with a given set of criteria. ISAKMP profiles can bind themselves to certificate maps, and if the presented certificate matches the certificate map present in an ISAKMP profile, the peer will be assigned the ISAKMP profile. If the ISAKMP profile contains a client configuration group name, the same group name will be assigned to the peer. This ISAKMP profile information will override the information in the ID_KEY_ID identity or in the first OU field of the certificate.

### Assigning an ISAKMP Profile and Group Name to a Peer

To assign an ISAKMP profile to a peer on the basis of arbitrary fields in the certificate, use the `match certificate` command after the ISAKMP profile has been defined.

To associate a group name with an ISAKMP profile that will be assigned to a peer, use the `client configuration group` command, also after the ISAKMP profile has been defined.

### How to Configure Certificate to ISAKMP Profile Mapping

- Mapping the Certificate to the ISAKMP Profile, page 41
- Verifying That the Certificate Has Been Mapped, page 42
- Assigning the Group Name to the Peer, page 43
- Monitoring and Maintaining Your Certificate to ISAKMP Profile Mapping, page 43

### Mapping the Certificate to the ISAKMP Profile

To map the certificate to the ISAKMP profile, perform the following steps. This configuration will enable you to assign the ISAKMP profile to a peer on the basis of the contents of arbitrary fields in the certificate.

**SUMMARY STEPS**

1. `enable`
2. `configure terminal`
3. `crypto isakmp profile profile-name`
4. `match certificate certificate-map`

**DETAILED STEPS**

<table>
<thead>
<tr>
<th>Command or Action</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Step 1</strong> enable</td>
<td>Enables privileged EXEC mode.</td>
</tr>
<tr>
<td></td>
<td>• Enter your password if prompted.</td>
</tr>
</tbody>
</table>

**Example:**

```
Router# enable
```
### Command or Action | Purpose
--- | ---
**Step 2** configure terminal | Enters global configuration mode.

**Example:**
```
Router# configure terminal
```

**Step 3** crypto isakmp profile *profile-name* | Defines an ISAKMP profile and enters into crypto ISAKMP profile configuration mode.

**Example:**
```
Router (config)# crypto isakmp profile vpnprofile
```

**Step 4** match certificate *certificate-map* | Accepts the name of a certificate map.

**Example:**
```
Router (conf-isa-prof)# match certificate map1
```

### Verifying That the Certificate Has Been Mapped

The following `show crypto ca certificates` command may be used to verify that the subject name of the certificate map has been properly configured.

**SUMMARY STEPS**
1. enable
2. show crypto ca certificates

**DETAILED STEPS**

| Command or Action | Purpose |
--- | --- |
**Step 1** enable | Enables privileged EXEC mode. |

**Example:**
```
Router# enable
```

**Step 2** show crypto ca certificates | Displays information about your certificate.

**Example:**
```
Router# show crypto ca certificates
```
**Assigning the Group Name to the Peer**

To associate a group name with a peer when the peer is mapped to an ISAKMP profile, perform the following steps.

**SUMMARY STEPS**

1. `enable`
2. `configure terminal`
3. `crypto isakmp profile profile-name`
4. `client configuration group group-name`

**DETAILED STEPS**

<table>
<thead>
<tr>
<th>Command or Action</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Step 1</strong> <code>enable</code></td>
<td>Enables privileged EXEC mode.</td>
</tr>
<tr>
<td>Example:</td>
<td></td>
</tr>
<tr>
<td>Router# enable</td>
<td></td>
</tr>
<tr>
<td><strong>Step 2</strong> <code>configure terminal</code></td>
<td>Enters global configuration mode.</td>
</tr>
<tr>
<td>Example:</td>
<td></td>
</tr>
<tr>
<td>Router# configure terminal</td>
<td></td>
</tr>
<tr>
<td><strong>Step 3</strong> <code>crypto isakmp profile profile-name</code></td>
<td>Defines an ISAKMP profile and enters into isakmp profile configuration mode.</td>
</tr>
<tr>
<td>Example:</td>
<td></td>
</tr>
<tr>
<td>Router (config)# crypto isakmp profile vpnprofile</td>
<td></td>
</tr>
<tr>
<td><strong>Step 4</strong> <code>client configuration group group-name</code></td>
<td>Accepts the name of a group that will be assigned to a peer when the peer is assigned this crypto ISAKMP profile.</td>
</tr>
<tr>
<td>Example:</td>
<td></td>
</tr>
<tr>
<td>Router (conf-isa-prof)# client configuration group group1</td>
<td></td>
</tr>
</tbody>
</table>

**Monitoring and Maintaining Your Certificate to ISAKMP Profile Mapping**

To monitor and maintain your certificate to ISAKMP profile mapping, you may use the following `debug crypto isakmp` command.
SUMMARY STEPS

1. enable
2. debug crypto isakmp

DETAILED STEPS

<table>
<thead>
<tr>
<th>Command or Action</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Step 1</strong> enable</td>
<td>Enables privileged EXEC mode.</td>
</tr>
<tr>
<td>Example:</td>
<td></td>
</tr>
<tr>
<td>Router# enable</td>
<td></td>
</tr>
<tr>
<td><strong>Step 2</strong> debug crypto isakmp</td>
<td>Displays output showing that the certificate has gone through certificate map matching and that the certificate matches the ISAKMP profile. The command may also be used to verify that the peer has been assigned a group.</td>
</tr>
<tr>
<td>Example:</td>
<td></td>
</tr>
<tr>
<td>Router# debug crypto isakmp</td>
<td></td>
</tr>
</tbody>
</table>

Certificate Examples for Certificate to ISAKMP Profile Mapping

- Certificates Mapped to the ISAKMP Profile on the Basis of Arbitrary Fields Example, page 44
- Group Name Assigned to a Peer That Is Associated with an ISAKMP Profile Example, page 45
- Mapping a Certificate to an ISAKMP Profile Verification Example, page 45
- Group Name Assigned to a Peer Verification Example, page 46

Certificates Mapped to the ISAKMP Profile on the Basis of Arbitrary Fields Example

The following configuration example shows that whenever a certificate contains “ou = green,” the ISAKMP profile “cert_pro” will be assigned to the peer:

```
crypto pki certificate map cert_map 10
  subject-name co ou = green
  !
crypto isakmp identity dn
crypto isakmp profile cert_pro
crypto ca trust-point 2315
  ca trust-point LaBcA
  initiate mode aggressive
  match certificate cert_map
```
Group Name Assigned to a Peer That Is Associated with an ISAKMP Profile Example

The following example shows that the group “some_group” is to be associated with a peer that has been assigned an ISAKMP profile:

```
crypto isakmp profile id_profile
  ca trust-point 2315
  match identity host domain cisco.com
  client configuration group some_group
```

Mapping a Certificate to an ISAKMP Profile Verification Example

The following examples show that a certificate has been mapped to an ISAKMP profile. The examples include the configurations for the responder and initiator, the show command output verifying that the subject name of the certificate map has been configured, and debug command output showing that the certificate has gone through certificate map matching and been matched to the ISAKMP profile.

Responder Configuration

```
crypto pki certificate map cert_map 10
  ! The above line is the certificate map definition.
  subject-name co ou = green
  ! The above line shows that the subject name must have “ou = green.”
  !
  crypto isakmp profile certpro
  ! The above line shows that this is the ISAKMP profile that will match if the certificate
  ! of the peer matches cert_map (shown on third line below).
  ca trust-point 2315
  ca trust-point LaBcA
  match certificate cert_map
  initiate mode aggressive
```

Initiator Configuration

```
crypto ca trustpoint LaBcA
  enrollment url http://10.76.82.20:80/cgi-bin/openscep
  subject-name ou=green,c=IN
  ! The above line ensures that the subject name “ou = green” is set.
  revocation-check none

show crypto ca certificates Command Output for the Initiator

Router# show crypto ca certificates
Certificate
  Status: Available
  Certificate Serial Number: 21
  Certificate Usage: General Purpose
  Issuer:
    cn=blue-lab CA
    o=CISCO
    c=IN
  Subject:
    Name: Router1.cisco.com
    c=IN
    ou=green
  ! The above line is a double check that “ou = green” has been set as the subject name.
  hostname=Router1.cisco.com
  Validity Date:
    start date: 14:34:30 UTC Mar 31 2004
debug crypto isakmp Command Output for the Responder

Router# debug crypto isakmp
6d23h: ISAKMP (0:268435460): received packet from 192.0.0.2 dport 500 sport 500 Global
(R) MM_KEY_EXCH
6d23h: ISAKMP: Main Mode packet contents (flags 1, len 892):
6d23h:       ID payload
6d23h:       FQDN <Router1.cisco.com> port 500 protocol 17
6d23h:       CERT payload
6d23h:       SIG payload
6d23h:       KEEPALIVE payload
6d23h:       NOTIFY payload
6d23h: ISAKMP: (0:4:HW:2): processing ID payload. message ID = 0
6d23h: ISAKMP (0:268435460): ID payload
next-payload : 6
type         : 2
FQDN name    : Router1.cisco.com
protocol     : 17
port         : 500
length       : 28
6d23h: ISAKMP: (0:4:HW:2): peer matches *none* of the profiles
6d23h: ISAKMP: (0:4:HW:2): processing CERT payload. message ID = 0
6d23h: ISAKMP: (0:4:HW:2): processing a CT_X509_SIGNATURE cert
6d23h: ISAKMP: (0:4:HW:2): peer's pubkey isn't cached
6d23h: ISAKMP: (0:4:HW:2): OU = green
6d23h: ISAKMP: (0:4:HW:2): certificate map matches certpro profile
! The above line shows that the certificate has gone through certificate map matching and
that it matches the “certpro” profile.
6d23h: ISAKMP: (0:4:HW:2): Trying to re-validate CERT using new profile
6d23h: ISAKMP: (0:4:HW:2): Creating CERT validation list: 2315, LaBcA,
6d23h: ISAKMP: (0:4:HW:2): CERT validity confirmed.

Group Name Assigned to a Peer Verification Example

The following configuration and debug output show that a group has been assigned to a peer.

Initiator Configuration

```
crypto isakmp profile certpro
  ca trust-point 2315
  ca trust-point LaBcA
  match certificate cert_map
  client configuration group new_group
! The statement on the above line will assign the group “new_group” to any peer that
matches the ISAKMP profile “certpro.”
  initiate mode aggressive
!```

debug crypto isakmp profile Command Output for the Responder

The following debug output example shows that the peer has been matched to the ISAKMP profile named “certpro” and that it has been assigned a group named “new_group.”

Router# debug crypto isakmp profile
6d23h: ISAKMP (0:268435461): received packet from 192.0.0.2 dport 500 sport 500 Global
(R) MM_KEY_EXCH
6d23h: ISAKMP: Main Mode packet contents (flags 1, len 892):
6d23h:       ID payload
6d23h:       FQDN <Router1.cisco.com> port 500 protocol 17
6d23h: CERT payload
6d23h: SIG payload
6d23h: KEEPALIVE payload
6d23h: NOTIFY payload
6d23h: ISAKMP:(0:5:HW:2):Input = IKE_MESG_FROM_PEER, IKE_MM_EXCH
6d23h: ISAKMP:(0:5:HW:2): processing ID payload. message ID = 0
6d23h: ISAKMP (0:268435461): ID payload
   next-payload : 6
   type         : 2
   FQDN name    : Router1.cisco.com
   protocol     : 17
   port         : 500
   length       : 28
6d23h: ISAKMP:(0:5:HW:2): peer matches *none* of the profiles
6d23h: ISAKMP:(0:5:HW:2): processing CERT payload. message ID = 0
6d23h: ISAKMP:(0:5:HW:2): processing a CT_X509_SIGNATURE cert
6d23h: ISAKMP:(0:5:HW:2): peer's pubkey isn't cached
6d23h: ISAKMP:(0:5:HW:2): OU = green
6d23h: ISAKMP:(0:5:HW:2): certificate map matches certpro profile
6d23h: ISAKMP:(0:5:HW:2): Trying to re-validate CERT using new profile
6d23h: ISAKMP:(0:5:HW:2): Creating CERT validation list: 2315, LaBch,
6d23h: ISAKMP:(0:5:HW:2): CERT validity confirmed.
6d23h: ISAKMP:(0:5:HW:2):Profile has no keyring, aborting key search
6d23h: ISAKMP:(0:5:HW:2): Profile certpro assigned peer the group named new_group

Additional References

- Related Documents, page 47
- Standards, page 47
- MIBs, page 48
- RFCs, page 48
- Technical Assistance, page 48

Related Documents

<table>
<thead>
<tr>
<th>Related Topic</th>
<th>Document Title</th>
</tr>
</thead>
<tbody>
<tr>
<td>Configuring ISAKMP profiles</td>
<td>VRF-Aware IPSec</td>
</tr>
<tr>
<td>Security commands</td>
<td>Cisco IOS Security Command Reference</td>
</tr>
</tbody>
</table>

Standards

<table>
<thead>
<tr>
<th>Standards</th>
<th>Title</th>
</tr>
</thead>
<tbody>
<tr>
<td>None</td>
<td>--</td>
</tr>
</tbody>
</table>
MIBs

<table>
<thead>
<tr>
<th>MIBs</th>
<th>MIBs Link</th>
</tr>
</thead>
</table>
| None | To locate and download MIBs for selected platforms, Cisco IOS XE releases, and feature sets, use Cisco MIB Locator found at the following URL:  

http://www.cisco.com/go/mibs |

RFCs

<table>
<thead>
<tr>
<th>RFCs</th>
<th>Title</th>
</tr>
</thead>
<tbody>
<tr>
<td>None</td>
<td>--</td>
</tr>
</tbody>
</table>

Technical Assistance

<table>
<thead>
<tr>
<th>Description</th>
<th>Link</th>
</tr>
</thead>
<tbody>
<tr>
<td>The Cisco Support website provides extensive online resources, including documentation and tools for troubleshooting and resolving technical issues with Cisco products and technologies. To receive security and technical information about your products, you can subscribe to various services, such as the Product Alert Tool (accessed from Field Notices), the Cisco Technical Services Newsletter, and Really Simple Syndication (RSS) Feeds. Access to most tools on the Cisco Support website requires a Cisco.com user ID and password.</td>
<td><a href="http://www.cisco.com/cisco/web/support/index.html">http://www.cisco.com/cisco/web/support/index.html</a></td>
</tr>
</tbody>
</table>

Feature Information for Certificate to ISAKMP Profile Mapping

The following table provides release information about the feature or features described in this module. This table lists only the software release that introduced support for a given feature in a given software release train. Unless noted otherwise, subsequent releases of that software release train also support that feature.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to www.cisco.com/go/cfn. An account on Cisco.com is not required.
Table 3  Feature Information for Certificate to ISAKMP Profile Mapping

<table>
<thead>
<tr>
<th>Feature Name</th>
<th>Releases</th>
<th>Feature Information</th>
</tr>
</thead>
<tbody>
<tr>
<td>Certificate to ISAKMP Profile Mapping</td>
<td>Cisco IOS XE Release 2.1</td>
<td>This feature enables you to assign an ISAKMP profile to a peer on the basis of the contents of arbitrary fields in the certificate. In addition, this feature allows you to assign a group name to those peers that are assigned an ISAKMP profile. The following commands were introduced or modified: client configuration group, match certificate (ISAKMP).</td>
</tr>
</tbody>
</table>
Encrypted Preshared Key

The Encrypted Preshared Key feature allows you to securely store plain text passwords in type 6 (encrypted) format in NVRAM.

- Finding Feature Information, page 51
- Restrictions for Encrypted Preshared Key, page 51
- Information About Encrypted Preshared Key, page 51
- How to Configure an Encrypted Preshared Key, page 53
- Configuration Examples for Encrypted Preshared Key, page 63
- Where to Go Next, page 64
- Additional References, page 64

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest feature information and caveats, see the release notes for your platform and software release. To find information about the features documented in this module, and to see a list of the releases in which each feature is supported, see the Feature Information Table at the end of this document.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to www.cisco.com/go/cfn. An account on Cisco.com is not required.

Restrictions for Encrypted Preshared Key

- Old ROM monitors (ROMMONs) and boot images cannot recognize the new type 6 passwords. Therefore, errors are expected if you boot from an old ROMMON.

Information About Encrypted Preshared Key

- Using the Encrypted Preshared Key Feature to Securely Store Passwords, page 51
- Enabling the Encrypted Preshared Key, page 53

Using the Encrypted Preshared Key Feature to Securely Store Passwords

Using the Encrypted Preshared Key feature, you can securely store plain text passwords in type 6 format in NVRAM using a command-line interface (CLI). Type 6 passwords are encrypted. Although the encrypted
passwords can be seen or retrieved, it is difficult to decrypt them to find out the actual password. Use the  
key config-key password-encryption command with the password encryption aes command to configure  
and enable the password (symmetric cipher AES is used to encrypt the keys). The password (key)  
configured using the config-key password-encryption command is the master encryption key that is used  
to encrypt all other keys in the router.

If you configure the password encryption aes command without configuring the key config-key  
password-encryption command, the following message is printed at startup or during any nonvolatile  
generation (NVGEN) process, such as when the show running-config or copy running-config startup-  
cfg commands have been configured:

“Can not encrypt password. Please configure a configuration-key with 'key config-key’”

- Changing a Password, page 52  
- Deleting a Password, page 52  
- Unconfiguring Password Encryption, page 52  
- Storing Passwords, page 52  
- Configuring New or Unknown Passwords, page 53

### Changing a Password

If the password (master key) is changed, or reencrypted, using the key config-key password-  
encrytption command, the list registry passes the old key and the new key to the application modules that  
are using type 6 encryption.

### Deleting a Password

If the master key that was configured using the key config-key password-encryption command is deleted  
from the system, a warning is printed (and a confirm prompt is issued) that states that all type 6 passwords  
will become useless. As a security measure, after the passwords have been encrypted, they will never be  
decrypted in the Cisco IOS XE software. However, passwords can be reencrypted as explained in the  
previous paragraph.

| Caution |

If the password configured using the key config-key password-encryption command is lost, it cannot be  
recovered. The password should be stored in a safe location.

### Unconfiguring Password Encryption

If you later unconfigure password encryption using the no password encryption aes command, all existing  
type 6 passwords are left unchanged, and as long as the password (master key) that was configured using  
the key config-key password-encryption command exists, the type 6 passwords will be decrypted as and  
when required by the application.

### Storing Passwords

Because no one can “read” the password (configured using the key config-key password-  
encription command), there is no way that the password can be retrieved from the router. Existing  
management stations cannot “know” what it is unless the stations are enhanced to include this key  
somewhere, in which case the password needs to be stored securely within the management system. If
Configurations are stored using TFTP, the configurations are not standalone, meaning that they cannot be loaded onto a router. Before or after the configurations are loaded onto a router, the password must be manually added (using the `key config-key password-encryption` command). The password can be manually added to the stored configuration but is not recommended because adding the password manually allows anyone to decrypt all passwords in that configuration.

**Configuring New or Unknown Passwords**

If you enter or cut and paste cipher text that does not match the master key, or if there is no master key, the cipher text is accepted or saved, but an alert message is printed. The alert message is as follows:

"ciphertext> [for username bar>] is incompatible with the configured master key."

If a new master key is configured, all the plain keys are encrypted and made type 6 keys. The existing type 6 keys are not encrypted. The existing type 6 keys are left as is.

If the old master key is lost or unknown, you have the option of deleting the master key using the `no key config-key password-encryption` command. Deleting the master key using the `no key config-key password-encryption` command causes the existing encrypted passwords to remain encrypted in the router configuration. The passwords will not be decrypted.

**Enabling the Encrypted Preshared Key**

The `password encryption aes` command is used to enable the encrypted password.

**How to Configure an Encrypted Preshared Key**

- Configuring an Encrypted Preshared Key, page 53
- Monitoring Encrypted Preshared Keys, page 54
- Configuring an ISAKMP Preshared Key, page 56
- Configuring an ISAKMP Preshared Key in ISAKMP Keyrings, page 57
- Configuring ISAKMP Aggressive Mode, page 58
- Configuring a Unity Server Group Policy, page 60
- Configuring an Easy VPN Client, page 61

**Configuring an Encrypted Preshared Key**

To configure an encrypted preshared key, perform the following steps.

**SUMMARY STEPS**

1. `enable`
2. `configure terminal`
3. `key config-key password-encryption [text]`
4. `password encryption aes`
### DETAILED STEPS

<table>
<thead>
<tr>
<th>Command or Action</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Step 1</strong> enable</td>
<td>Enables privileged EXEC mode.</td>
</tr>
<tr>
<td><strong>Example:</strong> Router&gt; enable</td>
<td></td>
</tr>
<tr>
<td><strong>Step 2</strong> configure terminal</td>
<td>Enters global configuration mode.</td>
</tr>
<tr>
<td><strong>Example:</strong> Router# configure terminal</td>
<td></td>
</tr>
<tr>
<td><strong>Step 3</strong> key config-key password-encryption [text]</td>
<td>Stores a type 6 encryption key in private NVRAM.</td>
</tr>
<tr>
<td><strong>Example:</strong> Router (config)# key config-key password-encryption</td>
<td></td>
</tr>
<tr>
<td><strong>Step 4</strong> password encryption aes</td>
<td>Enables the encrypted preshared key.</td>
</tr>
<tr>
<td><strong>Example:</strong> Router (config)# password-encryption aes</td>
<td></td>
</tr>
</tbody>
</table>

- Troubleshooting Tips, page 54

### Troubleshooting Tips

If you see the warning message "ciphertext >[for username bar]> is incompatible with the configured master key," you have entered or cut and pasted cipher text that does not match the master key or there is no master key. (The cipher text will be accepted or saved.) The warning message will allow you to locate the broken configuration line or lines.

### Monitoring Encrypted Preshared Keys

To get logging output for encrypted preshared keys, perform the following steps.

1. enable
2. password logging
SUMMARY STEPS

1. enable
2. password logging

DETAILED STEPS

<table>
<thead>
<tr>
<th>Command or Action</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Step 1</strong> enable</td>
<td>Enables privileged EXEC mode.</td>
</tr>
<tr>
<td><strong>Example:</strong></td>
<td>Enter your password if prompted.</td>
</tr>
<tr>
<td><strong>Step 2</strong> password logging</td>
<td>Provides a log of debugging output for a type 6 password operation.</td>
</tr>
<tr>
<td><strong>Example:</strong></td>
<td></td>
</tr>
</tbody>
</table>

Examples

The following password logging debug output shows that a new master key has been configured and that the keys have been encrypted with the new master key:

Router (config)# key config-key password-encrypt
New key:
Confirm key:
Router (config)#
01:40:57: TYPE6_PASS: New Master key configured, encrypting the keys with the new master key
Router (config)# key config-key password-encrypt
Old key:
New key:
Confirm key:
Router (config)#
01:42:11: TYPE6_PASS: Master key change heralded, re-encrypting the keys with the new master key
01:42:11: TYPE6_PASS: Mac verification successful
01:42:11: TYPE6_PASS: Mac verification successful
01:42:11: TYPE6_PASS: Mac verification successful

- Examples, page 55
- What To Do Next, page 56

Examples

The following password logging debug output shows that a new master key has been configured and that the keys have been encrypted with the new master key:

Router (config)# key config-key password-encrypt
New key:
Confirm key:
Router (config)#
01:40:57: TYPE6_PASS: New Master key configured, encrypting the keys with
the new master keypas
Router (config)# key config-key password-encrypt

Old key:
New key:
Confirm key:
Router (config)#
01:42:11: TYPE6_PASS: Master key change heralded, re-encrypting the keys
with the new master key
01:42:11: TYPE6_PASS: Mac verification successful
01:42:11: TYPE6_PASS: Mac verification successful
01:42:11: TYPE6_PASS: Mac verification successful

**What To Do Next**

You can perform any of the following procedures. Each procedure is independent of the others.

**Configuring an ISAKMP Preshared Key**

To configure an ISAKMP preshared key, perform the following procedure.

**SUMMARY STEPS**

1. enable
2. configure terminal
3. crypto isakmp key keystring address peer-address
4. crypto isakmp key keystring hostname hostname

**DETAILED STEPS**

<table>
<thead>
<tr>
<th>Command or Action</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Step 1</strong> enable</td>
<td>Enables privileged EXEC mode.</td>
</tr>
<tr>
<td>Example:</td>
<td>Enter your password if prompted.</td>
</tr>
<tr>
<td>Router# enable</td>
<td></td>
</tr>
<tr>
<td><strong>Step 2</strong> configure terminal</td>
<td>Enters global configuration mode.</td>
</tr>
<tr>
<td>Example:</td>
<td></td>
</tr>
<tr>
<td>Router# configure terminal</td>
<td></td>
</tr>
<tr>
<td><strong>Step 3</strong> crypto isakmp key keystring address peer-address</td>
<td>Configures a preshared authentication key.</td>
</tr>
<tr>
<td>Example:</td>
<td>The peer-address argument specifies the IP address of the remote peer.</td>
</tr>
<tr>
<td>Router (config)# crypto isakmp key cisco address 10.2.3.4</td>
<td></td>
</tr>
</tbody>
</table>
## Configuring an ISAKMP Preshared Key in ISAKMP Keyrings

To configure an ISAKMP preshared key in ISAKMP keyrings, which are used in IPSec Virtual Route Forwarding (VRF) configurations, perform the following procedure.

### SUMMARY STEPS
1. `enable`
2. `configure terminal`
3. `crypto keyring keyring-name`
4. `pre-shared-key address address key key`
5. `pre-shared-key hostname hostname key key`

### DETAILED STEPS

<table>
<thead>
<tr>
<th>Command or Action</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Step 1</strong> <code>enable</code></td>
<td>Enables privileged EXEC mode.</td>
</tr>
<tr>
<td><strong>Example:</strong></td>
<td></td>
</tr>
<tr>
<td><code>Router# enable</code></td>
<td></td>
</tr>
<tr>
<td><strong>Step 2</strong> <code>configure terminal</code></td>
<td>Enters global configuration mode.</td>
</tr>
<tr>
<td><strong>Example:</strong></td>
<td></td>
</tr>
<tr>
<td><code>Router# configure terminal</code></td>
<td></td>
</tr>
</tbody>
</table>

### Configuring an ISAKMP Preshared Key in ISAKMP Keyrings

The following sample output shows that an encrypted preshared key has been configured:

```
crypto isakmp key 6 \_Hg\^[\ECgLGGPF\^RXTQfDDWQ\]}YAAB address 10.2.3.4
crypto isakmp key 6 `eR\eTRaKCUZPYQfDgXRWi_AAB hostname example.com
```

### Example

The following sample output shows that an encrypted preshared key has been configured:

```
crypto isakmp key keystring1
hostname keystring1.com
```

- **Example, page 57**

### Command or Action | Purpose
---|---
**Step 4** `crypto isakmp key keystring hostname hostname` | Configures a preshared authentication key. |
| | - The `hostname` argument specifies the fully qualified domain name (FQDN) of the peer. |

- **Example:**
  
  `Router (config)# crypto isakmp key keystring1
  hostname keystring1.com`
<table>
<thead>
<tr>
<th>Command or Action</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Step 3</strong> crypto keyring <strong>keyring-name</strong></td>
<td>Defines a crypto keyring to be used during Internet Key Exchange (IKE) authentication and enters keyring configuration mode.</td>
</tr>
<tr>
<td><strong>Example:</strong></td>
<td></td>
</tr>
<tr>
<td>Router (config)# crypto keyring keyring1</td>
<td></td>
</tr>
<tr>
<td><strong>Step 4</strong> pre-shared-key <strong>address</strong> <strong>address</strong> <strong>key</strong> <strong>key</strong></td>
<td>Defines a preshared key to be used for IKE authentication.</td>
</tr>
<tr>
<td><strong>Example:</strong></td>
<td></td>
</tr>
<tr>
<td>Router (config-keyring)# pre-shared-key address 10.2.3.5 key cisco</td>
<td></td>
</tr>
<tr>
<td><strong>Step 5</strong> pre-shared-key <strong>hostname</strong> <strong>hostname</strong> <strong>key</strong> <strong>key</strong></td>
<td>Defines a preshared key to be used for IKE authentication.</td>
</tr>
<tr>
<td><strong>Example:</strong></td>
<td></td>
</tr>
<tr>
<td>Router (config-keyring)# pre-shared-key hostname example.com key cisco</td>
<td></td>
</tr>
</tbody>
</table>

- Example, page 58

**Example**

The following is sample output from the `show-running-config` command. The output shows that an encrypted preshared key in ISAKMP keyrings has been configured.

```
crypto keyring keyring1
  pre-shared-key address 10.2.3.5 key 6 `WHCJYR_Z|GRFF^RXTQfDcfZ]GPAAB
  pre-shared-key hostname keyring1.com key 6 aE_REHdcOfYCPF"RXTQfDJYVNSAAB
```

## Configuring ISAKMP Aggressive Mode

To configure ISAKMP aggressive mode, perform the following steps.

**SUMMARY STEPS**

1. enable
2. configure terminal
3. crypto isakmp peer ip-address **ip-address**
4. set aggressive-mode client-endpoint **client-endpoint**
5. set aggressive-mode password **password**
### DETAILED STEPS

<table>
<thead>
<tr>
<th>Command or Action</th>
<th>Purpose</th>
</tr>
</thead>
</table>
| **Step 1 enable**                          | Enables privileged EXEC mode.  
  - Enter your password if prompted. |
| **Example:**                               |         |
| ```                             |         |
| Router# enable                         |         |
| **Step 2 configure terminal**            | Enters global configuration mode. |
| **Example:**                               |         |
| ```                             |         |
| Router# configure terminal               |         |
| **Step 3 crypto isakmp peer ip-address ip-address** | To enable an IP Security (IPSec) peer for IKE querying of authentication, authorization, and accounting (AAA) for tunnel attributes in aggressive mode and to enter ISAKMP peer configuration mode. |
| **Example:**                               |         |
| ```                             |         |
| Router (config)# crypto isakmp peer ip-address 10.2.3.4 |         |
| **Step 4 set aggressive-mode client-endpoint client-endpoint** | Specifies the Tunnel-Client-Endpoint attribute within an ISAKMP peer configuration. |
| **Example:**                               |         |
| ```                             |         |
| Router (config-isakmp-peer)# set aggressive-mode client-endpoint fqdn cisco.com |         |
| **Step 5 set aggressive-mode password password** | Specifies the Tunnel-Password attribute within an ISAKMP peer configuration. |
| **Example:**                               |         |
| ```                             |         |
| Router (config-isakmp-peer)# set aggressive-mode password cisco |         |

- Example, page 59

### Example

The following `show-running-config` sample output shows that an encrypted preshared key in ISAKMP aggressive mode has been configured.

```
crypto isakmp peer address 10.2.3.4
set aggressive-mode password 6 ^aKPIQ_KJE_PPF^RXTQfDTIaLNeAAB
set aggressive-mode client-endpoint fqdn cisco.com
```
Configuring a Unity Server Group Policy

To configure a unity server group policy, perform the following steps.

**SUMMARY STEPS**

1. enable
2. configure terminal
3. crypto isakmp client configuration group *group-name*
4. pool *name*
5. domain name
6. key *name*

**DETAILED STEPS**

<table>
<thead>
<tr>
<th>Command or Action</th>
<th>Purpose</th>
</tr>
</thead>
</table>
| **Step 1** enable | Enables privileged EXEC mode.  
  - Enter your password if prompted. |
| **Example:** | | Router# enable |
| **Step 2** configure terminal | Enters global configuration mode. |
| **Example:** | | Router# configure terminal |
| **Step 3** crypto isakmp client configuration group *group-name* | Specifies the policy profile of the group that will be defined and enters ISAKMP group configuration mode. |
| **Example:** | | Router (config)# crypto isakmp client configuration group group1 |
| **Step 4** pool *name* | Defines a local pool address. |
| **Example:** | | Router (config-isakmp-group)# pool group1 |
| **Step 5** domain name | Specifies the Domain Name Service (DNS) domain to which a group belongs. |
| **Example:** | | Router (config-isakmp-group)# domain cisco.com |
Configuring an Easy VPN Client

To configure an Easy VPN client, perform the following steps.

**SUMMARY STEPS**

1. enable
2. configure terminal
3. `crypto ipsec client ezvpn name`
4. peer `ipaddress`
5. mode `client`
6. group `group-name key group-key`
7. connect `manual`

**DETAILED STEPS**

<table>
<thead>
<tr>
<th>Command or Action</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Step 1</strong> enable</td>
<td>Enables privileged EXEC mode.</td>
</tr>
<tr>
<td></td>
<td>• Enter your password if prompted.</td>
</tr>
</tbody>
</table>

Example:

```console
Router# enable
```

---

### Command or Action | Purpose
---|---
**Step 6** key `name` | Specifies the IKE preshared key for group policy attribute definition.

Example:

```console
Router (config-isakmp-group)# key cisco
```

- Example, page 61

---

**Example**

The following `show-running-config` sample output shows that an encrypted key has been configured for a unity server group policy:

```console
crypto isakmp client configuration group group1
token 6 cZZgDZPOE\dTDPF^RXTQfDTLaLNeAAB
domain cisco.com
pool group1
```

---

**Configuring an Easy VPN Client**

To configure an Easy VPN client, perform the following steps.
<table>
<thead>
<tr>
<th>Command or Action</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Step 2</strong> configure terminal</td>
<td>Enters global configuration mode.</td>
</tr>
<tr>
<td><strong>Example:</strong></td>
<td></td>
</tr>
<tr>
<td>Router# configure terminal</td>
<td></td>
</tr>
<tr>
<td><strong>Step 3</strong> crypto ipsec client ezvpn <code>name</code></td>
<td>Creates a Cisco Easy VPN remote configuration and enters Cisco Easy VPN remote configuration mode.</td>
</tr>
<tr>
<td><strong>Example:</strong></td>
<td></td>
</tr>
<tr>
<td>Router (config)# crypto ipsec client ezvpn ezvpnnamel</td>
<td></td>
</tr>
<tr>
<td><strong>Step 4</strong> peer <code>ipaddress</code></td>
<td>Sets the peer IP address for the VPN connection.</td>
</tr>
<tr>
<td><strong>Example:</strong></td>
<td></td>
</tr>
<tr>
<td>Router (config-isakmp-peer)# peer 10.2.3.4</td>
<td></td>
</tr>
<tr>
<td><strong>Step 5</strong> mode client</td>
<td>Automatically configures the router for Cisco Easy VPN Client mode operation, which uses Network Address Translation (NAT) or Peer Address Translation (PAT) address translations.</td>
</tr>
<tr>
<td><strong>Example:</strong></td>
<td></td>
</tr>
<tr>
<td>Router (config-isakmp-ezvpv)# mode client</td>
<td></td>
</tr>
<tr>
<td><strong>Step 6</strong> group <code>group-name</code> key <code>group-key</code></td>
<td>Specifies the group name and key value for the VPN connection.</td>
</tr>
<tr>
<td><strong>Example:</strong></td>
<td></td>
</tr>
<tr>
<td>Router (config-isakmp-ezvpvn)# group ezvpnnamel key cisco</td>
<td></td>
</tr>
<tr>
<td><strong>Step 7</strong> connect manual</td>
<td>Specifies the manual setting for directing the Cisco Easy VPN remote client to wait for a command or application program interface (API) call before attempting to establish the Cisco Easy VPN remote connection.</td>
</tr>
<tr>
<td><strong>Example:</strong></td>
<td></td>
</tr>
<tr>
<td>Router (config-isakmp-ezvpvn)# connect manual</td>
<td></td>
</tr>
</tbody>
</table>

- Example, page 62

**Example**

The following `show-running-config` sample output shows that an Easy VPN client has been configured. The key has been encrypted.

```
crypto ipsec client ezvpn ezvpnnamel
```
**Configuration Examples for Encrypted Preshared Key**

- Encrypted Preshared Key Example, page 63
- No Previous Key Present Example, page 63
- Key Already Exists Example, page 63
- Key Already Exists But the User Wants to Key In Interactively Example, page 64
- No Key Present But the User Wants to Key In Interactively Example, page 64
- Removal of the Password Encryption Example, page 64

**Encrypted Preshared Key Example**

The following is an example of a configuration for which a type 6 preshared key has been encrypted. It includes the prompts and messages that a user might see.

```
Router (config)# crypto isakmp key cisco address 10.0.0.2
Router (config)# exit
Router# configure terminal
Enter configuration commands, one per line. End with CNTL/Z.
Router (config)# password encryption aes
Router (config)# key config-key password-encrypt
New key:
Confirm key:
01:46:40: TYPE6_PASS: New Master key configured, encrypting the keys with
the new master key
Router (config)# exit
Router # show running-config | include crypto isakmp key
crypto isakmp key 6 CXWdhVTZvB^cIHDOahiFTa address 10.0.0.2
```

**No Previous Key Present Example**

In the following configuration example, no previous key is present:

```
Router (config)# key config-key password-encryption testkey 123
```

**Key Already Exists Example**

In the following configuration example, a key already exists:

```
Router (config)# key config-key password-encryption testkey123
Old key:
Router (config)#
```
Key Already Exists But the User Wants to Key In Interactively Example

In the following configuration example, the user wants to key in interactively, but a key already exists. The Old key, New key, and Confirm key prompts will show on your screen if you enter the `key config-key password-encryption` command and press the enter key to get into interactive mode.

```
Router (config)# key config-key password-encryption
Old key:
New key:
Confirm key:
```

No Key Present But the User Wants to Key In Interactively Example

In the following example, the user wants to key in interactively, but no key is present. The New key and Confirm key prompts will show on your screen if you are in interactive mode.

```
Router (config)# key config-key password-encryption
New key:
Confirm key:
```

Removal of the Password Encryption Example

In the following configuration example, the user wants to remove the encrypted password. The “WARNING: All type 6 encrypted keys will become unusable. Continue with master key deletion? [yes/no]” prompt will show on your screen if you are in interactive mode.

```
Router (config)# no key config-key password-encryption
WARNING: All type 6 encrypted keys will become unusable. Continue with master key deletion? [yes/no]: y
```

Where to Go Next

Configure any other preshared keys.

Additional References

- Related Documents, page 65
- Standards, page 47
- MIBs, page 65
- RFCs, page 48
- Technical Assistance, page 65
- Feature Information for Encrypted Preshared Key, page 65
# Related Documents

<table>
<thead>
<tr>
<th>Related Topic</th>
<th>Document Title</th>
</tr>
</thead>
<tbody>
<tr>
<td>Configuring passwords</td>
<td><em>Cisco IOS Security Command Reference</em></td>
</tr>
</tbody>
</table>

## Standards

<table>
<thead>
<tr>
<th>Standards</th>
<th>Title</th>
</tr>
</thead>
<tbody>
<tr>
<td>None</td>
<td>--</td>
</tr>
</tbody>
</table>

## MIBs

<table>
<thead>
<tr>
<th>MIBs</th>
<th>MIBs Link</th>
</tr>
</thead>
<tbody>
<tr>
<td>No new or modified MIBs are supported by this feature, and support for existing MIBs has not been modified by this feature.</td>
<td>To locate and download MIBs for selected platforms, Cisco IOS XE software releases, and feature sets, use Cisco MIB Locator found at the following URL: <a href="http://www.cisco.com/go/mibs">http://www.cisco.com/go/mibs</a></td>
</tr>
</tbody>
</table>

## RFCs

<table>
<thead>
<tr>
<th>RFCs</th>
<th>Title</th>
</tr>
</thead>
<tbody>
<tr>
<td>None</td>
<td>--</td>
</tr>
</tbody>
</table>

## Technical Assistance

<table>
<thead>
<tr>
<th>Description</th>
<th>Link</th>
</tr>
</thead>
<tbody>
<tr>
<td>The Cisco Support and Documentation website provides online resources to download documentation, software, and tools. Use these resources to install and configure the software and to troubleshoot and resolve technical issues with Cisco products and technologies. Access to most tools on the Cisco Support and Documentation website requires a Cisco.com user ID and password.</td>
<td><a href="http://www.cisco.com/cisco/web/support/index.html">http://www.cisco.com/cisco/web/support/index.html</a></td>
</tr>
</tbody>
</table>

## Feature Information for Encrypted Preshared Key

*GUID-83AF864B-57CA-4EDB-B1F1-84F130A4433E1* lists the release history for this feature.
Use Cisco Feature Navigator to find information about platform support and software image support. Cisco Feature Navigator enables you to determine which Cisco IOS XE software images support a specific software release, feature set, or platform. To access Cisco Feature Navigator, go to http://tools.cisco.com/ITDIT/CFN/jsp/index.jsp. An account on Cisco.com is not required.

**Note**

GUID-83AF864B-57CA-4EDB-B1F1-84F130A4433E lists only the Cisco IOS XE software release that introduced support for a given feature in a given Cisco IOS XE software release train. Unless noted otherwise, subsequent releases of that Cisco IOS XE software release train also support that feature.

**Table 4 Feature Information for Encrypted Preshared Key**

<table>
<thead>
<tr>
<th>Feature Name</th>
<th>Releases</th>
<th>Feature Information</th>
</tr>
</thead>
<tbody>
<tr>
<td>Encrypted Preshared Key</td>
<td>Cisco IOS XE Release 2.1</td>
<td>The Encrypted Preshared Key feature allows you to securely store plain text passwords in type 6 (encrypted) format in NVRAM. The following commands were introduced or modified: crypto ipsec client ezvpn (global), crypto isakmp client configuration group, crypto isakmp key, key config-key password-encryption, password encryption aes, password logging, pre-shared-key, set aggressive-mode password.</td>
</tr>
</tbody>
</table>

Cisco and the Cisco logo are trademarks or registered trademarks of Cisco and/or its affiliates in the U.S. and other countries. To view a list of Cisco trademarks, go to this URL: [www.cisco.com/go/trademarks](http://www.cisco.com/go/trademarks). Third-party trademarks mentioned are the property of their respective owners. The use of the word partner does not imply a partnership relationship between Cisco and any other company. (1110R)

Any Internet Protocol (IP) addresses and phone numbers used in this document are not intended to be actual addresses and phone numbers. Any examples, command display output, network topology diagrams, and other figures included in the document are shown for illustrative purposes only. Any use of actual IP addresses or phone numbers in illustrative content is unintentional and coincidental.
Distinguished Name Based Crypto Maps

The Distinguished Name Based Crypto Maps feature allows you to set restrictions in the router configuration that prevent peers with specific certificates—especially certificates with particular DNs—from having access to selected encrypted interfaces.

- Finding Feature Information, page 67
- Prerequisites for Distinguished Name Based Crypto Maps, page 67
- Restrictions for Distinguished Name Based Crypto Maps, page 67
- Information About Distinguished Name Based Crypto Maps, page 68
- How to Configure Distinguished Name Based Crypto Maps, page 68
- Configuration Examples for Distinguished Name Based Crypto Maps, page 72
- Additional References, page 73
- Feature Information for Distinguished Name Based Crypto Maps, page 74

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest feature information and caveats, see the release notes for your platform and software release. To find information about the features documented in this module, and to see a list of the releases in which each feature is supported, see the Feature Information Table at the end of this document.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to www.cisco.com/go/cfn. An account on Cisco.com is not required.

Prerequisites for Distinguished Name Based Crypto Maps

Before configuring a distinguished name (DN) based crypto map, you must perform the following tasks:

- Create an Internet Key Exchange (IKE) policy at each peer.

For more information on creating IKE policies and crypto map entries, refer to Configuring Internet Key Exchange for IPsec VPNs

Restrictions for Distinguished Name Based Crypto Maps
System Requirements
To configure this feature, your router must support IP Security (IPsec).

Performance Impact
If you restrict access to a large number of DNs, it is recommended that you specify a few number of crypto maps referring to large identity sections instead of specifying a large number of crypto maps referring to small identity sections.

Information About Distinguished Name Based Crypto Maps

- Feature Overview, page 68

Feature Overview
The Distinguished Name Based Crypto Maps feature allows you to configure the router to restrict access to selected encrypted interfaces for those peers with specific certificates, especially certificates with particular Distinguished Names (DNs).

This feature allows you to configure which crypto maps are usable to a peer based on the DN that a peer used to authenticate itself, thereby enabling you to control which encrypted interfaces a peer with a specified DN can access.

How to Configure Distinguished Name Based Crypto Maps

- Configuring DN Based Crypto Maps (authenticated by DN), page 68
- Configuring DN Based Crypto Maps (authenticated by hostname), page 69
- Applying Identity to DN Based Crypto Maps, page 70
- Verifying DN Based Crypto Maps, page 71
- Troubleshooting Tips, page 72

Configuring DN Based Crypto Maps (authenticated by DN)
To configure a DN based crypto map that can be used only by peers that have been authenticated by a DN, perform the following steps.

SUMMARY STEPS
1. enable
2. configure terminal
3. crypto identity name
4. dn name =string [ ,name=string]
## DETAILED STEPS

<table>
<thead>
<tr>
<th>Command or Action</th>
<th>Purpose</th>
</tr>
</thead>
</table>
| **Step 1** enable | Enables privileged EXEC mode.  
|                   | • Enter your password if prompted. |
|                   | **Example:**  
|                   | Router> enable |
| **Step 2** configure terminal | Enters global configuration mode. |
|                   | **Example:**  
|                   | Router# configure terminal |
| **Step 3** crypto identity name | Configures the identity of a router with the given list of DNs in the certificate of the router and enters crypto identity configuration mode. |
|                   | **Example:**  
|                   | Router (config)# crypto identity sample-com |
| **Step 4** dn name=string [name=string] | Associates the identity of the router with the DN in the certificate of the router.  
|                   | The identity of the peer must match the identity in the exchanged certificate. |
|                   | **Example:**  
|                   | Router(crypto-identity)# dn ou=mystring |

## Configuring DN Based Crypto Maps (authenticated by hostname)

To configure a DN based crypto map that can be used only by peers that have been authenticated by a hostname, perform the following steps.

### SUMMARY STEPS

1. enable  
2. configure terminal  
3. crypto identity name  
4. fqdn name
DETAILED STEPS

<table>
<thead>
<tr>
<th>Command or Action</th>
<th>Purpose</th>
</tr>
</thead>
</table>
| **Step 1** enable | Enables privileged EXEC mode.  
  - Enter your password if prompted. |
| **Example:** |  
  Router> enable |
| **Step 2** configure terminal | Enters global configuration mode. |
| **Example:** |  
  Router# configure terminal |
| **Step 3** crypto identity name | Configures the identity of a router with the given list of DNs in the certificate of the router and enters crypto identity configuration mode. |
| **Example:** |  
  Router (config)# crypto identity sample-com |
| **Step 4** fqdn name | Associates the identity of the router with the hostname that the peer used to authenticate itself.  
  **Note** The identity of the peer must match the identity in the exchanged certificate. |
| **Example:** |  
  Router (crypto-identity)# fqdn sample.com |

Applying Identity to DN Based Crypto Maps

To apply the identity (within the crypto map context), perform the following steps.

**SUMMARY STEPS**

1. enable  
2. configure terminal  
3. crypto map map-name seq-num ipsec-isakmp  
4. set identity name
### Detailed Steps

<table>
<thead>
<tr>
<th>Command or Action</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Step 1</strong> enable</td>
<td>Enables privileged EXEC mode.</td>
</tr>
<tr>
<td>Example:</td>
<td></td>
</tr>
<tr>
<td>Router&gt; enable</td>
<td></td>
</tr>
<tr>
<td></td>
<td>• Enter your password if prompted.</td>
</tr>
<tr>
<td><strong>Step 2</strong> configure terminal</td>
<td>Enters global configuration mode.</td>
</tr>
<tr>
<td>Example:</td>
<td></td>
</tr>
<tr>
<td>Router# configure terminal</td>
<td></td>
</tr>
<tr>
<td><strong>Step 3</strong> crypto map map-name seq-num ipsec-isakmp</td>
<td>Creates or modifies a crypto map entry and enters the crypto map configuration mode.</td>
</tr>
<tr>
<td>Example:</td>
<td></td>
</tr>
<tr>
<td>Router (config)# crypto map mymap 10 ipsec-isakmp</td>
<td></td>
</tr>
<tr>
<td><strong>Step 4</strong> set identity name</td>
<td>Applies the identity to the crypto map.</td>
</tr>
<tr>
<td>Example:</td>
<td>When this command is applied, only the hosts that match a configuration listed within the <code>set identity name</code> can use the specified crypto map.</td>
</tr>
<tr>
<td>Router (config-crypto-map)# set identity sample-com</td>
<td>If the <code>set identity</code> command does not appear within the crypto map, the encrypted connection does not have any restrictions other than the IP address of the encrypting peer.</td>
</tr>
</tbody>
</table>

### Verifying DN Based Crypto Maps

To verify that this functionality is properly configured, perform the following steps.

**SUMMARY STEPS**

1. enable
2. show crypto identity
DETAILED STEPS

<table>
<thead>
<tr>
<th>Command or Action</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>Step 1 enable</td>
<td>Enables privileged EXEC mode.</td>
</tr>
<tr>
<td>Example:</td>
<td>- Enter your password if prompted.</td>
</tr>
<tr>
<td>Step 2 show crypto identity</td>
<td>Displays the configured identities.</td>
</tr>
<tr>
<td>Example:</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Router&gt; enable</td>
</tr>
<tr>
<td></td>
<td>Router# show crypto identity</td>
</tr>
</tbody>
</table>

Troubleshooting Tips

If an encrypting peer attempts to establish a connection that is blocked by the DN based crypto map configuration, the following error message will be logged:

```plaintext
<time>: %CRYPTO-4-IKE_QUICKMODE_BAD_CERT: encrypted connection attempted with a peer without the configured certificate attributes.
```

Configuration Examples for Distinguished Name Based Crypto Maps

- DN Based Crypto Map Configuration Example, page 72

DN Based Crypto Map Configuration Example

The following example shows how to configure DN based crypto maps that have been authenticated by DN and hostname. Comments are included inline to explain various commands.

```plaintext
! DN based crypto maps require you to configure an IKE policy at each peer.
crypto isakmp policy 15
  encryption 3des
  hash md5
  authentication rsa-sig
  group 2
  lifetime 5000
crypto isakmp policy 20
  authentication pre-share
  lifetime 10000
crypto isakmp key 1234567890 address 172.31.224.33
```

```plaintext
! The following is an IPSec crypto map (part of IPSec configuration). It can be used only!
! by peers that have been authenticated by DN and if the certificate belongs to BigBiz.
crypto map map-to-bigbiz 10 ipsec-isakmp
  set peer 172.21.114.196
  set transform-set my-transformset
  set identity to-bigbiz
```
Distinguished Name Based Crypto Maps

match address 124
!
crypto identity to-bigbiz
dn ou=BigBiz
!
! This crypto map can be used only by peers that have been authenticated by hostname
! and if the certificate belongs to little.com.
crypto map map-to-little-com 10 ipsec-isakmp
set peer 172.21.115.119
set transform-set my-transformset
match address 125
identity to-little-com
!
crypto identity to-little-com
fqdn little.com
!

Additional References

Related Documents

<table>
<thead>
<tr>
<th>Related Topic</th>
<th>Document Title</th>
</tr>
</thead>
<tbody>
<tr>
<td>Internet Key Exchange and IPsec</td>
<td>Configuring Internet Key Exchange for IPsec VPNs</td>
</tr>
<tr>
<td>Security commands</td>
<td>Cisco IOS Security Command Reference</td>
</tr>
</tbody>
</table>

Standards

<table>
<thead>
<tr>
<th>Standard</th>
<th>Title</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>No new or modified standards are supported by this feature, and support for existing standards has not been modified by this feature. --</td>
</tr>
</tbody>
</table>

MIBs

<table>
<thead>
<tr>
<th>MIB</th>
<th>MIBs Link</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>To locate and download MIBs for selected platforms, Cisco IOS XE software releases, and feature sets, use Cisco MIB Locator found at the following URL: <a href="http://www.cisco.com/go/mibs">http://www.cisco.com/go/mibs</a></td>
</tr>
</tbody>
</table>

RFCs

<table>
<thead>
<tr>
<th>RFC</th>
<th>Title</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>No new or modified RFCs are supported by this feature, and support for existing RFCs has not been modified by this feature. --</td>
</tr>
</tbody>
</table>
Technical Assistance

<table>
<thead>
<tr>
<th>Description</th>
<th>Link</th>
</tr>
</thead>
<tbody>
<tr>
<td>The Cisco Support website provides extensive online resources, including documentation and tools for troubleshooting and resolving technical issues with Cisco products and technologies. To receive security and technical information about your products, you can subscribe to various services, such as the Product Alert Tool (accessed from Field Notices), the Cisco Technical Services Newsletter, and Really Simple Syndication (RSS) Feeds. Access to most tools on the Cisco Support website requires a Cisco.com user ID and password.</td>
<td><a href="http://www.cisco.com/techsupport">http://www.cisco.com/techsupport</a></td>
</tr>
</tbody>
</table>

Feature Information for Distinguished Name Based Crypto Maps

The following table provides release information about the feature or features described in this module. This table lists only the software release that introduced support for a given feature in a given software release train. Unless noted otherwise, subsequent releases of that software release train also support that feature.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to www.cisco.com/go/cfn. An account on Cisco.com is not required.

Table 5 Feature Information for Distinguished Name Based Crypto Maps

<table>
<thead>
<tr>
<th>Feature Name</th>
<th>Releases</th>
<th>Feature Information</th>
</tr>
</thead>
<tbody>
<tr>
<td>Distinguished Name Based Crypto Maps</td>
<td>Cisco IOS XE Release 2.1</td>
<td>This feature allows you to configure the router to restrict access to selected encrypted interfaces for those peers with specific certificates, especially certificates with particular Distinguished Names (DNs). The following commands were introduced or modified: crypto identity, dn, fqdn, and identity.</td>
</tr>
</tbody>
</table>

Cisco and the Cisco logo are trademarks or registered trademarks of Cisco and/or its affiliates in the U.S. and other countries. To view a list of Cisco trademarks, go to this URL: www.cisco.com/go/trademarks.
Third-party trademarks mentioned are the property of their respective owners. The use of the word partner does not imply a partnership relationship between Cisco and any other company. (1110R)

Any Internet Protocol (IP) addresses and phone numbers used in this document are not intended to be actual addresses and phone numbers. Any examples, command display output, network topology diagrams, and other figures included in the document are shown for illustrative purposes only. Any use of actual IP addresses or phone numbers in illustrative content is unintentional and coincidental.
VRF-Aware IPsec

The VRF-Aware IPsec feature introduces IP Security (IPsec) tunnel mapping to Multiprotocol Label Switching (MPLS) Virtual Private Networks (VPNs). By using the VRF-Aware IPsec feature, you can map IPsec tunnels to Virtual Routing and Forwarding (VRF) instances using a single public-facing address.

- Finding Feature Information, page 77
- Restrictions for VRF-Aware IPsec, page 77
- Information About VRF-Aware IPsec, page 78
- How to Configure VRF-Aware IPsec, page 80
- Configuration Examples for VRF-Aware IPsec, page 97
- Additional References, page 103
- Feature Information for VRF-Aware IPsec, page 104
- Glossary, page 105

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest feature information and caveats, see the release notes for your platform and software release. To find information about the features documented in this module, and to see a list of the releases in which each feature is supported, see the Feature Information Table at the end of this document.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to www.cisco.com/go/cfn. An account on Cisco.com is not required.

Restrictions for VRF-Aware IPsec

- If you are configuring VRF-Aware IPsec using a crypto map configuration and the Inside VRF (IVRF) is not the same as the Front Door VRF (FVRF), this feature is not interoperable with unicast reverse path forwarding (uRPF) if uRPF is enabled on the crypto map interface. If your network requires uRPF, it is recommended that you use Virtual Tunnel Interface (VTI) for IPsec instead of crypto maps.
- The VRF-Aware IPsec feature does not allow IPsec tunnel mapping between VRFs. For example, it does not allow IPsec tunnel mapping from VRF vpn1 to VRF vpn2.
- When the VRF-Aware IPsec feature is used with a crypto map, this crypto map cannot use the global VRF as the IVRF and a non-global VRF as the FVRF. However, configurations based on virtual tunnel interfaces do not have that limitation.
Information About VRF-Aware IPsec

- VRF Instance, page 78
- MPLS Distribution Protocol, page 78
- VRF-Aware IPsec Functional Overview, page 78

VRF Instance

A VRF instance is a per-VPN routing information repository that defines the VPN membership of a customer site attached to the Provider Edge (PE) router. A VRF comprises an IP routing table, a derived Cisco Express Forwarding (CEF) table, a set of interfaces that use the forwarding table, and a set of rules and routing protocol parameters that control the information that is included in the routing table. A separate set of routing and CEF tables is maintained for each VPN customer.

MPLS Distribution Protocol

The MPLS distribution protocol is a high-performance packet-forwarding technology that integrates the performance and traffic management capabilities of data link layer switching with the scalability, flexibility, and performance of network-layer routing.

VRF-Aware IPsec Functional Overview

Front Door VRF (FVRF) and Inside VRF (IVRF) are central to understanding the feature.

Each IPsec tunnel is associated with two VRF domains. The outer encapsulated packet belongs to one VRF domain, which we shall call the FVRF, while the inner, protected IP packet belongs to another domain called the IVRF. Another way of stating the same thing is that the local endpoint of the IPsec tunnel belongs to the FVRF while the source and destination addresses of the inside packet belong to the IVRF.

One or more IPsec tunnels can terminate on a single interface. The FVRF of all these tunnels is the same and is set to the VRF that is configured on that interface. The IVRF of these tunnels can be different and depends on the VRF that is defined in the Internet Security Association and Key Management Protocol (ISAKMP) profile that is attached to a crypto map entry.
GUID-640C027D-F465-4B58-93B1-1DC0CDC5A91AE is an illustration of a scenario showing IPsec to MPLS and Layer 2 VPNs.

**Figure 2 IPsec to MPLS and Layer 2 VPNs**

- Packet Flow into the IPsec Tunnel, page 79
- Packet Flow from the IPsec Tunnel, page 79

**Packet Flow into the IPsec Tunnel**

- A VPN packet arrives from the Service Provider MPLS backbone network to the PE and is routed through an interface facing the Internet.
- The packet is matched against the Security Policy Database (SPD), and the packet is IPsec encapsulated. The SPD includes the IVRF and the access control list (ACL).
- The IPsec encapsulated packet is then forwarded using the FVRF routing table.

**Packet Flow from the IPsec Tunnel**

- An IPsec-encapsulated packet arrives at the PE router from the remote IPsec endpoint.
- IPsec performs the Security Association (SA) lookup for the Security Parameter Index (SPI), destination, and protocol.
- The packet is decapsulated using the SA and is associated with IVRF.
- The packet is further forwarded using the IVRF routing table.
How to Configure VRF-Aware IPsec

- Configuring Crypto Keyrings, page 80
- Configuring ISAKMP Profiles, page 83
- Configuring an ISAKMP Profile on a Crypto Map, page 86
- Configuring to Ignore Extended Authentication During IKE Phase 1 Negotiation, page 88
- Verifying VRF-Aware IPsec, page 88
- Clearing Security Associations, page 89
- Troubleshooting VRF-Aware IPsec, page 90

Configuring Crypto Keyrings

A crypto keyring is a repository of preshared and Rivest, Shamir, and Adelman (RSA) public keys. There can be zero or more keyrings on the Cisco IOS router.

Perform the following optional task to configure a crypto keyring.

**SUMMARY STEPS**

1. enable
2. configure terminal
3. crypto keyring keyring-name [vrf fvrf-name]
4. description string
5. pre-shared-key {address address [mask] | hostname hostname} key key
6. rsa-pubkey {address address | name fqdn} [encryption | signature]
7. address ip-address
8. serial-number serial-number
9. key-string
10. text
11. quit
12. exit
13. exit

**DETAILED STEPS**

<table>
<thead>
<tr>
<th>Command or Action</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Step 1</strong> enable</td>
<td>Enables privileged EXEC mode.</td>
</tr>
<tr>
<td>Example:</td>
<td></td>
</tr>
<tr>
<td>Router&gt; enable</td>
<td></td>
</tr>
</tbody>
</table>

• Enter your password if prompted.
<table>
<thead>
<tr>
<th>Command or Action</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Step 2</strong> configure terminal</td>
<td>Enters global configuration mode.</td>
</tr>
<tr>
<td>Example:</td>
<td></td>
</tr>
<tr>
<td>Router# configure terminal</td>
<td></td>
</tr>
<tr>
<td><strong>Step 3</strong> crypto keyring keyring-name [vrf fvrf-name]</td>
<td>Defines a keyring with <em>keyring-name</em> as the name of the keyring and enters keyring configuration mode.</td>
</tr>
<tr>
<td>Example:</td>
<td></td>
</tr>
<tr>
<td>Router (config)# crypto keyring VPN1</td>
<td></td>
</tr>
<tr>
<td><strong>Step 4</strong> description string</td>
<td>(Optional) Specifies a one-line description of the keyring.</td>
</tr>
<tr>
<td>Example:</td>
<td></td>
</tr>
<tr>
<td>Router (config-keyring)# description The keys for VPN1</td>
<td></td>
</tr>
<tr>
<td><strong>Step 5</strong> pre-shared-key {address address [mask]</td>
<td>(Optional) Defines a preshared key by address or host name.</td>
</tr>
<tr>
<td>hostname hostname} key key</td>
<td></td>
</tr>
<tr>
<td>Example:</td>
<td></td>
</tr>
<tr>
<td>Router (config-keyring)# pre-shared-key address 10.72.23.11 key VPN1</td>
<td></td>
</tr>
<tr>
<td><strong>Step 6</strong> rsa-pubkey {address address</td>
<td>(Optional) Defines a Rivest, Shamir, and Adelman (RSA) public key by address or host name and enters rsa-pubkey configuration mode.</td>
</tr>
<tr>
<td>name fqdn} [encryption</td>
<td>signature]</td>
</tr>
<tr>
<td>Example:</td>
<td></td>
</tr>
<tr>
<td>Router(config-keyring)# rsa-pubkey name host.vpn.com</td>
<td></td>
</tr>
</tbody>
</table>
### How to Configure VRF-Aware IPsec

#### Command or Action

<table>
<thead>
<tr>
<th>Step</th>
<th>Command or Action</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>Step 7</td>
<td>address ip-address</td>
<td>(Optional) Defines the RSA public key IP address.</td>
</tr>
<tr>
<td></td>
<td>Example:</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Router(config-pubkey-key)# address 10.5.5.1</td>
<td></td>
</tr>
<tr>
<td>Step 8</td>
<td>serial-number serial-number</td>
<td>(Optional) Specifies the serial number of the public key. The value is from 0 through infinity.</td>
</tr>
<tr>
<td></td>
<td>Example:</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Router(config-pubkey-key)# serial-number 1000000</td>
<td></td>
</tr>
<tr>
<td>Step 9</td>
<td>key-string</td>
<td>Enters into the text mode in which you define the public key.</td>
</tr>
<tr>
<td></td>
<td>Example:</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Router(config-pubkey-key)# key-string</td>
<td></td>
</tr>
<tr>
<td>Step 10</td>
<td>text</td>
<td>Specifies the public key.</td>
</tr>
<tr>
<td></td>
<td>Note</td>
<td>Only one public key may be added in this step.</td>
</tr>
<tr>
<td></td>
<td>Example:</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Router(config-pubkey)# 00302017 4A7D385B 1234EF29 335FC973</td>
<td></td>
</tr>
<tr>
<td>Step 11</td>
<td>quit</td>
<td>Quits to the public key configuration mode.</td>
</tr>
<tr>
<td></td>
<td>Example:</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Router(config-pubkey)# quit</td>
<td></td>
</tr>
<tr>
<td>Step 12</td>
<td>exit</td>
<td>Exits to the keyring configuration mode.</td>
</tr>
<tr>
<td></td>
<td>Example:</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Router(config-pubkey)# exit</td>
<td></td>
</tr>
<tr>
<td>Step 13</td>
<td>exit</td>
<td>Exits to global configuration mode.</td>
</tr>
<tr>
<td></td>
<td>Example:</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Router(config-keyring)# exit#</td>
<td></td>
</tr>
</tbody>
</table>
Configuring ISAKMP Profiles

An ISAKMP profile is a repository for IKE Phase 1 and IKE Phase 1.5 configuration for a set of peers. An ISAKMP profile defines items such as keepalive, trustpoints, peer identities, and XAUTH AAA list during the IKE Phase 1 and Phase 1.5 exchange. There can be zero or more ISAKMP profiles on the Cisco IOS router.

**Note**

If traffic from the router to a certification authority (CA) (for authentication, enrollment, or for obtaining a certificate revocation list [CRL]) or to an Lightweight Directory Access Protocol (LDAP) server (for obtaining a CRL) needs to be routed via a VRF, the `vrf` command must be added to the trustpoint. Otherwise, such traffic will use the default routing table.

- If a profile does not specify one or more trustpoints, all trustpoints in the router will be used to attempt to validate the certificate of the peer (Internet Key Exchange (IKE) main mode or signature authentication). If one or more trustpoints are specified, only those trustpoints will be used.

**Note**

A router initiating IKE and a router responding to the IKE request should have symmetrical trustpoint configurations. For example, a responding router (in IKE Main Mode) performing RSA signature encryption and authentication might use trustpoints that were defined in the global configuration when sending the CERT-REQ payloads. However, the router might use a restricted list of trustpoints that were defined in the ISAKMP profile for the certificate verification. If the peer (the IKE initiator) is configured to use a certificate whose trustpoint is in the global list of the responding router but not in ISAKMP profile of the responding router, the certificate will be rejected. (However, if the initiating router does not know about the trustpoints in the global configuration of the responding router, the certificate can still be authenticated.)

**SUMMARY STEPS**

1. `enable`
2. `configure terminal`
3. `crypto isakmp profile profile-name`
4. `description string`
5. `vrf ivrf-name`
6. `keepalive seconds retry retry-seconds`
7. `self-identity {address | fqdn| user-fqdn user-fqdn}
8. `keyring keyring-name`
9. `ca trust-point {trustpoint-name}
10. `match identity {group group-name | address address [mask] [ivrf] | host host-name | host domain domain-name | user user-fqdn | user domain domain-name}
11. `client configuration address {initiate | respond}
12. `client authentication list list-name`
13. `isamkp authorization list list-name`
14. `initiate mode aggressive`
15. `exit`
## DETAILED STEPS

<table>
<thead>
<tr>
<th>Command or Action</th>
<th>Purpose</th>
</tr>
</thead>
</table>
| **Step 1** enable | Enables privileged EXEC mode.  
- Enter your password if prompted. |
| **Example:** |  
Router> enable |
| **Step 2** configure terminal | Enters global configuration mode. |
| **Example:** |  
Router# configure terminal |
| **Step 3** crypto isakmp profile *profile-name* | Defines an Internet Security Association and Key Management Protocol (ISAKMP) profile and enters into isakmp profile configuration mode. |
| **Example:** |  
Router (config)# crypto isakmp profile vpnprofile |
| **Step 4** description *string* | (Optional) Specifies a one-line description of an ISAKMP profile. |
| **Example:** |  
Router (config-isa-prof)# description configuration for VPN profile |
| **Step 5** vrf *ivrf-name* | (Optional) Maps the IPsec tunnel to a Virtual Routing and Forwarding (VRF) instance.  
**Note**  
The VRF also serves as a selector for matching the Security Policy Database (SPD). If the VRF is not specified in the ISAKMP profile, the IVRF of the IPsec tunnel will be the same as its FVRF. |
| **Example:** |  
Router (config-isa-prof)# vrf VPN1 |
| **Step 6** keepalive *seconds* retry *retry-seconds* | (Optional) Allows the gateway to send dead peer detection (DPD) messages to the peer.  
- If not defined, the gateway uses the globally configured value.  
- *seconds* --Number of seconds between DPD messages. The range is from 10 to 3600 seconds.  
- *retry retry-seconds* --Number of seconds between retries if the DPD message fails. The range is from 2 to 60 seconds. |
| **Example:** |  
Router (config-isa-prof)# keepalive 60 retry 5 |
### Command or Action

| Step 7 | self-identity {address | fqdn | user-fqdn user-fqdn} |
|--------|--------------------------|
|        | (Optional) Specifies the identity that the local Internet Key Exchange (IKE) should use to identify itself to the remote peer. |
|        | • If not defined, IKE uses the global configured value. |
|        | • address --Uses the IP address of the egress interface. |
|        | • fqdn-- Uses the fully qualified domain name (FQDN) of the router. |
|        | • user-fqdn --Uses the specified value. |

**Example:**

Router (config-isa-prof)# self-identity address

### Step 8

**keyring keyring-name**

((Optional) Specifies the keyring to use for Phase 1 authentication.

**Example:**

**Example:**

Router (config-isa-prof)# keyring VPN1

### Step 9

**ca trust-point {trustpoint-name}**

(Optional) Specifies a trustpoint to validate a Rivest, Shamir, and Adelman (RSA) certificate.

**Example:**

Router (config-isa-prof)# ca trustpoint VPN1-trustpoint

### Step 10

**match identity {group group-name | address address [mask] [fvrf] | host hostname | host domain domain-name | user user-fqdn | user domain domain-name}**

Specifies the client IKE Identity (ID) that is to be matched.

**Example:**

Router (config-isa-prof)# match identity address 10.1.1.1

- group group-name --Matches the group-name with the ID type ID_KEY_ID. It also matches the group-name with the Organizational Unit (OU) field of the Distinguished Name (DN).
- address address [mask] fvrf--Matches the address with the ID type ID_IPV4_ADDR. The mask argument can be used to specify a range of addresses. The fvrf argument specifies that the address is in Front Door Virtual Routing and Forwarding (FVRF)
- host hostname --Matches the hostname with the ID type ID_FQDN.
- host domain domain-name --Matches the domain-name to the ID type ID_FQDN whose domain name is the same as the domain-name. Use this command to match all the hosts in the domain.
- user username --Matches the username with the ID type ID_USER_FQDN
- user domain domain-name --Matches the ID type ID_USER_FQDN whose domain name matches the domain-name.
### Command or Action

| Step 11 | client configuration address [initiate | respond] | Purpose |
|---------|--------------------------------------|---------|
|         | (Optional) Specifies whether to initiate the mode configuration exchange or responds to mode configuration requests. |

**Example:**

```plaintext
Router (config-isa-prof)# client configuration address initiate
```

<table>
<thead>
<tr>
<th>Step 12</th>
<th>client authentication list list-name</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>(Optional) Authentication, authorization, and accounting (AAA) to use for authenticating the remote client during the extended authentication (XAUTH) exchange.</td>
<td></td>
</tr>
</tbody>
</table>

**Example:**

```plaintext
Router (config-isa-prof)# client authentication list xauthlist
```

<table>
<thead>
<tr>
<th>Step 13</th>
<th>isakmp authorization list list-name</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>(Optional) Network authorization server for receiving the Phase 1 preshared key and other attribute-value (AV) pairs.</td>
<td></td>
</tr>
</tbody>
</table>

**Example:**

```plaintext
Router (config-isa-prof)# isakmp authorization list ikessaaalist
```

<table>
<thead>
<tr>
<th>Step 14</th>
<th>initiate mode aggressive</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>(Optional) Initiates aggressive mode exchange.</td>
<td></td>
</tr>
<tr>
<td></td>
<td>· If not specified, IKE always initiates Main Mode exchange.</td>
<td></td>
</tr>
</tbody>
</table>

**Example:**

```plaintext
Router (config-isa-prof)# initiate mode aggressive
```

<table>
<thead>
<tr>
<th>Step 15</th>
<th>exit</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Exits to global configuration mode.</td>
<td></td>
</tr>
</tbody>
</table>

**Example:**

```plaintext
Router (config-isa-prof)# exit
```

- What to Do Next, page 86

### What to Do Next

Go to the section “Configuring an ISAKMP Profile on a Crypto Map, page 86.”

### Configuring an ISAKMP Profile on a Crypto Map

An ISAKMP profile must be applied to the crypto map. The IVRF on the ISAKMP profile is used as a selector when matching the VPN traffic. If there is no IVRF on the ISAKMP profile, the IVRF will be equal to the FVRF. Perform this required task to configure an ISAKMP profile on a crypto map.
Before configuring an ISAKMP profile on a crypto map, you must first have configured your router for basic IPsec.

**SUMMARY STEPS**

1. enable
2. configure terminal
3. crypto map *map-name isakmp-profile isakmp-profile-name*
4. set isakmp-profile *profile-name*
5. exit

**DETAILED STEPS**

<table>
<thead>
<tr>
<th>Command or Action</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>Step 1 enable</td>
<td>Enables privileged EXEC mode.</td>
</tr>
<tr>
<td>Example:</td>
<td>Enter your password if prompted.</td>
</tr>
<tr>
<td>Router&gt; enable</td>
<td></td>
</tr>
<tr>
<td>Step 2 configure terminal</td>
<td>Enters global configuration mode.</td>
</tr>
<tr>
<td>Example:</td>
<td>Router# configure terminal</td>
</tr>
<tr>
<td>Step 3 crypto map <em>map-name isakmp-profile isakmp-profile-name</em></td>
<td>(Optional) Specifies the Internet Key Exchange and Key Management Protocol (ISAKMP) profile for the crypto map set and enters crypto map configuration mode.</td>
</tr>
<tr>
<td>Example:</td>
<td>The ISAKMP profile will be used during IKE exchange.</td>
</tr>
<tr>
<td>Router (config)# crypto map vpnmap isakmp-profile vpnprofile</td>
<td></td>
</tr>
<tr>
<td>Step 4 set isakmp-profile <em>profile-name</em></td>
<td>(Optional) Specifies the ISAKMP profile to use when the traffic matches the crypto map entry.</td>
</tr>
<tr>
<td>Example:</td>
<td>Router (config-crypto-map)# set isakmp-profile vpnprofile</td>
</tr>
<tr>
<td>Step 5 exit</td>
<td>Exits to global configuration mode.</td>
</tr>
<tr>
<td>Example:</td>
<td>Router (config-crypto-map)# exit</td>
</tr>
</tbody>
</table>
Configuring to Ignore Extended Authentication During IKE Phase 1 Negotiation

To ignore XAUTH during an IKE Phase 1 negotiation, use the `no crypto xauth` command. Use the `no crypto xauth` command if you do not require extended authentication for the Unity clients.

**SUMMARY STEPS**

1. `enable`
2. `configure terminal`
3. `no crypto xauth interface`

**DETAILED STEPS**

<table>
<thead>
<tr>
<th>Command or Action</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Step 1</strong> enable</td>
<td>Enables privileged EXEC mode.</td>
</tr>
<tr>
<td></td>
<td>- Enter your password if prompted.</td>
</tr>
<tr>
<td>Example:</td>
<td></td>
</tr>
<tr>
<td><code>Router&gt; enable</code></td>
<td></td>
</tr>
<tr>
<td><strong>Step 2</strong> configure terminal</td>
<td>Enters global configuration mode.</td>
</tr>
<tr>
<td>Example:</td>
<td></td>
</tr>
<tr>
<td><code>Router# configure terminal</code></td>
<td></td>
</tr>
<tr>
<td><strong>Step 3</strong> no crypto xauth interface</td>
<td>Ignores XAUTH proposals for requests that are destined to the IP address of the interface. By default, Internet Key Exchange (IKE) processes XAUTH proposals.</td>
</tr>
<tr>
<td>Example:</td>
<td></td>
</tr>
<tr>
<td><code>Router(config)# no crypto xauth ethernet0</code></td>
<td></td>
</tr>
</tbody>
</table>

Verifying VRF-Aware IPsec

To verify your VRF-Aware IPsec configurations, use the following `show` commands. These `show` commands allow you to list configuration information and security associations (SAs):

**SUMMARY STEPS**

1. `enable`
2. `show crypto ipsec sa [map map-name | address | identity | interface interface | peer [vrf fvrf-name] address | vrf fvrf-name] [detail]`
3. `show crypto isakmp key`
4. `show crypto isakmp profile`
5. `show crypto key pubkey-chain rsa`
## DETAILED STEPS

<table>
<thead>
<tr>
<th>Command or Action</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Step 1</strong> enable</td>
<td>Enables privileged EXEC mode.</td>
</tr>
<tr>
<td>Example:</td>
<td>• Enter your password if prompted.</td>
</tr>
<tr>
<td>Router&gt; enable</td>
<td></td>
</tr>
<tr>
<td><strong>Step 2</strong> show crypto ipsec sa</td>
<td>Allows you to view the settings used by current security associations (SAs).</td>
</tr>
<tr>
<td>Example:</td>
<td></td>
</tr>
<tr>
<td>Router# show crypto ipsec sa vrf vpn1</td>
<td></td>
</tr>
<tr>
<td><strong>Step 3</strong> show crypto isakmp key</td>
<td>Lists all the keyrings and their preshared keys.</td>
</tr>
<tr>
<td>Example:</td>
<td>• Use this command to verify your crypto keyring configuration.</td>
</tr>
<tr>
<td>Router# show crypto isakmp key</td>
<td></td>
</tr>
<tr>
<td><strong>Step 4</strong> show crypto isakmp profile</td>
<td>Lists all ISAKMP profiles and their configurations.</td>
</tr>
<tr>
<td>Example:</td>
<td></td>
</tr>
<tr>
<td>Router# show crypto isakmp profile</td>
<td></td>
</tr>
<tr>
<td><strong>Step 5</strong> show crypto key pubkey-chain rsa</td>
<td>Views the Rivest, Shamir, and Adelman (RSA) public keys of the peer that are stored on your router.</td>
</tr>
<tr>
<td>Example:</td>
<td>• The output is extended to show the keyring to which the public key belongs.</td>
</tr>
<tr>
<td>Router# show crypto key pubkey-chain rsa</td>
<td></td>
</tr>
</tbody>
</table>

## Clearing Security Associations

The following `clear` commands allow you to clear SAs.

### SUMMARY STEPS

1. `enable`
2. `clear crypto sa [counters | map map-name | peer [vrf vrf-name] address | spi address {ah | esp} spi | vrf ivrf-name]`
### DETAILED STEPS

<table>
<thead>
<tr>
<th>Command or Action</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Step 1</strong> enable</td>
<td>Enables privileged EXEC mode.</td>
</tr>
<tr>
<td><strong>Example:</strong></td>
<td></td>
</tr>
<tr>
<td>Router&gt; enable</td>
<td></td>
</tr>
</tbody>
</table>

| **Step 2** clear crypto sa [counters | map map-name | peer[vrf vrf-name] address | spi address {ah | esp} spi | vrf ivrf-name] | Clears the IPsec security associations (SAs). |
|-----------------|-----------|---------------------|-----------------|---------------------|
| **Example:** | |
| Router# clear crypto sa vrf VPN1 | |

### Troubleshooting VRF-Aware IPsec

To troubleshoot VRF-Aware IPsec, use the following **debug** commands:

### SUMMARY STEPS

1. enable
2. debug crypto ipsec
3. debug crypto isakmp

### DETAILED STEPS

<table>
<thead>
<tr>
<th>Command or Action</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Step 1</strong> enable</td>
<td>Enables privileged EXEC mode.</td>
</tr>
<tr>
<td><strong>Example:</strong></td>
<td></td>
</tr>
<tr>
<td>Router&gt; enable</td>
<td></td>
</tr>
</tbody>
</table>

| **Step 2** debug crypto ipsec | Displays IP security (IPsec) events. |
| **Example:** | |
| Router# debug crypto ipsec | |

| **Step 3** debug crypto isakmp | Displays messages about Internet Key Exchange (IKE) events. |
| **Example:** | |
| Router(config)# debug crypto isakmp | |
Debug Examples for VRF-Aware IPsec

The following sample debug outputs are for a VRF-aware IPsec configuration:

**IPsec PE**

Router# debug crypto ipsec
Crypto IPSEC debugging is on
IPSEC-PE#debug crypto isakmp
Crypto ISAKMP debugging is on
IPSEC-PE#debug crypto isakmp detail
Crypto ISAKMP internals debugging is on

04:32:07: ISAKMP: Deleting peer node by peer_reap for 10.1.1.1: 63C142F8
04:32:55: ISAKMP cookie gen for src 10.1.1.1 dst 172.16.1.1
04:32:55: ISAKMP cookie AA8F7B41 0C07C670
04:32:55: ISAKMP: Setting client config settings 648252B0
04:32:55: ISAKMP: Setting client xauth list and state
04:32:55: ISAKMP: (Re)Setting client xauth preshared
04:32:55: ISAKMP: encryption 3DES-CBC
04:32:55: ISAKMP: hash SHA
04:32:55: ISAKMP: auth XAUTHInitPreShared
04:32:55: ISAKMP: life type in seconds
04:32:55: ISAKMP: life duration (VPI) of 0x0 0x20 0xC4 0x9B
04:32:55: ISAKMP: atts are acceptable. Next payload is 3
04:32:55: ISAKMP: processing vendor id payload
04:32:55: ISAKMP: vendor ID seems Unity/DPD but major 157 mismatch
04:32:55: ISAKMP: vendor ID is NAT-T v3
04:32:55: ISAKMP: processing vendor id payload
04:32:55: ISAKMP: vendor ID seems Unity/DPD but major 123 mismatch
04:32:55: ISAKMP: vendor ID is NAT-T v2
04:32:55: ISAKMP: Authentication by xauth preshared
04:32:55: ISAKMP: Checking ISAKMP transform 1 against priority 1 policy

Internet Key Exchange for IPsec VPNs Configuration Guide Cisco IOS XE Release 2
04:32:55: ISAKMP (0:13): vendor ID seems Unity/DPD but major 123 mismatch
04:32:55: ISAKMP (0:13): vendor ID is NAT-T v2
04:32:55: ISAKMP (0:13): processing KE payload. message ID = 0
04:32:55: ISAKMP (0:13): processing NONCE payload. message ID = 0
04:32:55: ISAKMP (0:13): processing vendor id payload
04:32:55: ISAKMP (0:13): vendor ID is DPD
04:32:55: ISAKMP (0:13): processing vendor id payload
04:32:55: ISAKMP (0:13): claimed IOS but failed authentication
04:32:55: ISAKMP (0:13): processing vendor id payload
04:32:55: ISAKMP (0:13): vendor ID is Unity
04:32:55: ISAKMP (0:13): Input = IKE_MESG_FROM_PEER, IKE_AM_EXCH
04:32:55: ISAKMP cookie gen for src 11.1.1.1 dst 172.16.1.1
04:32:55: ISAKMP cookie AA8F7B41 7AE1E0DF
04:32:55:       crawler his_cookie AA8F7B41 F7ACF384
04:32:55:        crawler his_cookie E46E088D F227FE4D
04:32:55: ISAKMP (0:13): constructed NAT-T vendor-03 ID
04:32:55: ISAKMP (0:13): ID payload
next-payload : 10
type         : 1
addr         : 172.16.1.1
protocol     : 17
port         : 0
length       : 8
04:32:55: ISAKMP (0:13): constructed HIS NAT-D
04:32:55: ISAKMP (0:13): sending packet to 10.1.1.1 my_port 500 peer_port 500 (R)
AG_INIT_EXCH
04:32:55: ISAKMP (0:13): Input = IKE_MESG_FROM_AAA, PRESHARED_KEY_REPLY
04:32:55: ISAKMP (0:13): received payload type 17
04:32:55: ISAKMP (0:13): detected NAT-D payload

04:32:55: ISAKMP (0:13): processor ID seems Unity/DPD but major 175 mismatch
04:32:55: ISAKMP (0:13): vendor ID is XAUTH
04:32:55: ISAKMP (0:13): processing vendor id payload
04:32:55: ISAKMP (0:13): claimed IOS but failed authentication
04:32:55: ISAKMP (0:13): processing vendor id payload
04:32:55: ISAKMP (0:13): vendor ID is Unity
04:32:55: ISAKMP (0:13): Input = IKE_MESG_FROM_PEER, IKE_AM_EXCH
04:32:55: ISAKMP cookie gen for src 11.1.1.1 dst 172.16.1.1
04:32:55: ISAKMP cookie AA8F7B41 7AE1E0DF
04:32:55:       crawler his_cookie AA8F7B41 F7ACF384
04:32:55:        crawler his_cookie E46E088D F227FE4D
04:32:55: ISAKMP (0:13): constructed NAT-T vendor-03 ID
04:32:55: ISAKMP (0:13): ID payload
next-payload : 10
type         : 1
addr         : 172.16.1.1
protocol     : 17
port         : 0
length       : 8
04:32:55: ISAKMP (0:13): constructed HIS NAT-D
04:32:55: ISAKMP (0:13): sending packet to 10.1.1.1 my_port 500 peer_port 500 (R)
AG_INIT_EXCH
04:32:55: ISAKMP (0:13): Input = IKE_MESG_FROM_AAA, PRESHARED_KEY_REPLY
04:32:55: ISAKMP (0:13): received payload type 17
04:32:55: ISAKMP (0:13): detected NAT-D payload
04:32:55: ISAKMP (0:13): recalc my hash for NAT-D
04:32:55: ISAKMP (0:13): NAT match MINE hash
04:32:55: ISAKMP: received payload type 17
04:32:55: ISAKMP (0:13): Detected NAT-D payload
04:32:55: ISAKMP (0:13): recalc his hash for NAT-D
04:32:55: ISAKMP (0:13): NAT match HIS hash
04:32:55: ISAKMP (0:13): processing NOTIFY INITIAL_CONTACT protocol 1
spi 0, message ID = 0, sa = 6482B354
04:32:55: ISAKMP (0:13): Process initial contact, bring down existing phase 1 and 2 SA's with local 172.16.1.1 remote 10.1.1.1 remote port 500
04:32:55: ISAKMP (0:13): returning IP addr to the address pool
04:32:55: ISAKMP cookie gen for src 10.1.1.1 dst 172.16.1.1
04:32:55: ISAKMP cookie AA8F7B41 05D315C5
04:32:55: ISAKMP cookie gen for src 172.16.1.1 dst 10.1.1.1
04:32:55: ISAKMP cookie 3123100B 041A85A6
04:32:55: ISAKMP (0:13): SA has been authenticated with 10.1.1.1
04:32:55: ISAKMP: set new node -803402627 to CONF_XAUTH
04:32:55: IPSEC(key_engine): got a queue event...
04:32:55: ISAKMP (0:13): sending packet to 10.1.1.1 my_port 500 peer_port 500 (R)
QM_IDLE
04:32:55: ISAKMP (0:13): purging node -803402627
04:32:55: ISAKMP: Sending phase 1 responder lifetime 86400
04:32:55: ISAKMP (0:13): Input = IKE_MESG_FROM_PEER, IKE_AM_EXCH
04:32:55: ISAKMP cookie gen for src 10.1.1.1 dst 172.168.1.1
04:32:55: ISAKMP cookie AA8F7B41 25EEF256
04:32:55: crawler my_cookie AA8F7B41 F7ACF384
crawler his_cookie E46E088D F227FE4D
04:32:55: ISAKMP (0:13): initiating peer config to 10.1.1.1. ID = -1447732198
04:32:55: ISAKMP (0:13): sending packet to 10.1.1.1 my_port 500 peer_port 500 (R)
CONF_XAUTH
04:32:55: ISAKMP (0:13): Input = IKE_MESG_FROM_AAA, IKE_AAA_START_LOGIN
04:32:55: ISAKMP cookie gen for src 172.18.1.1 dst 10.1.1.1
04:32:55: ISAKMP cookie 3123100B 124D4618
04:32:55: ISAKMP cookie gen for src 10.1.1.1 dst 172.18.1.1
04:32:55: ISAKMP cookie AA8F7B41 B0C91917
crawler my_cookie AA8F7B41 F7ACF384
crawler his_cookie E46E088D P227FE4D
04:32:55: ISAKMP (0:13): got callback 1
04:32:55: ISAKMP: set new node -1447732198 to CONF_XAUTH
04:32:55: ISAKMP/xauth: request attribute XAUTH_USER_NAME_V2
04:32:55: ISAKMP/xauth: request attribute XAUTH_USER_PASSWORD_V2
04:32:55: ISAKMP (0:13): initiating peer config to 10.1.1.1. ID = -1447732198
04:32:55: ISAKMP (0:13): sending packet to 10.1.1.1 my_port 500 peer_port 500 (R)
CONF_XAUTH
04:32:55: ISAKMP (0:13): Input = IKE_MESG_FROM_AAA, IKE_AAA_START_LOGIN
04:33:00: ISAKMP (0:13): Retransmitting phase 2 CONF_XAUTH -1447732198...
04:33:00: ISAKMP (0:13): incrementing error counter on sa: retransmit phase 2
04:33:00: ISAKMP (0:13): incrementing error counter on sa: retransmit phase 2
04:33:00: ISAKMP (0:13): retransmitting phase 2 -1447732198 CONF_XAUTH
04:33:00: ISAKMP (0:13): sending packet to 10.1.1.1 my_port 500 peer_port 500 (R)
CONF_XAUTH
04:33:03: ISAKMP cookie gen for src 172.18.1.1 dst 10.1.1.1
04:33:03: ISAKMP cookie 3123100B 124D4618
04:33:03: ISAKMP cookie gen for src 10.1.1.1 dst 172.18.1.1
04:33:03: ISAKMP cookie AA8F7B41 B0C91917
04:33:03: ISAKMP: isadb_post_process_list: crawler: 9 27FF 2 (6482B354)
crawler my_cookie AA8F7B41 F7ACF384
crawler his_cookie E46E088D P227FE4D
04:33:03: ISAKMP (0:13): received packet from 10.1.1.1 dport 500 sport 500 Global (R)
04:33:03: ISAKMP: hash from 7292D74 for 92 bytes
04:33:03: ISAKMP: Packet hash:
6421BCC0: 84A1AF24 5D92B116
6421BCC0: FC2C6252 A472C5F8 152AC860 63
04:33:03: ISAKMP (0:13): processing transaction payload from 11.1.1.1. message ID = -144732198
04:33:03: ISAKMP: Config payload REPLY
04:33:03: ISAKMP/xauth: reply attribute XAUTH_USER_NAME_V2
04:33:03: ISAKMP/xauth: reply attribute XAUTH_USER_PASSWORD_V2
04:33:03: ISAKMP (0:13): deleting node -1447732198 error FALSE reason "done with xauth request/reply exchange"
04:33:03: ISAKMP (0:13): Input = IKE_MESG_FROM_PEER, IKE_CFG_REPLY
04:33:03: ISAKMP (0:13): Old State = IKE_XAUTH_REQ_SENT New State = IKE_XAUTH_AAA_CONT_LOGIN_AWAIT
04:33:03: ISAKMP cookie gen for src 10.1.1.1 dst 172.18.1.1
04:33:03: ISAKMP cookie AA8F7B41 A1B3E684
04:33:03: ISAKMP: isadb_post_process_list: crawler: B 27FF 12 (6482B354)
04:33:03: crawler my_cookie AA8F7B41 F7ACF384
04:33:03: crawler his_cookie E46E088D F227FE4D
04:33:03: ISAKMP (0:13): deleting node 524716665 error FALSE reason "done with transaction"
04:33:03: ISAKMP (0:13): Input = IKE_MESG_FROM_AAA, IKE_AAA_CONT_LOGIN
04:33:03: ISAKMP (0:13): Old State = IKE_XAUTH_AAA_CONT_LOGIN_AWAIT New State = IKE_XAUTH_SET_SENT
04:33:03: ISAKMP cookie gen for src 172.18.1.1 dst 10.1.1.1
04:33:03: ISAKMP cookie 3123100B 5C83A09D
04:33:03: ISAKMP cookie gen for src 10.1.1.1 dst 172.18.1.1
04:33:03: ISAKMP cookie AA8F7B41 2BEBEFD4
04:33:03: ISAKMP: isadb_post_process_list: crawler: B 27FF 2 (6482B354)
04:33:03: crawler my_cookie AA8F7B41 F7ACF384
04:33:03: crawler his_cookie E46E088D F227FE4D
04:33:03: ISAKMP (0:13): input packet from 10.1.1.1 dport 500 sport 500 Global (R)
04:33:03: ISAKMP: hash from 7292A34 for 68 bytes
04:33:03: ISAKMP: Packet hash:
6421BCC0: 5034B99E B8BA531F
6421BCC0: 6267B8BD F3006989 DC118796 63
04:33:03: ISAKMP (0:13): processing transaction payload from 11.1.1.1. message ID = 524716665
04:33:03: ISAKMP: Config payload ACK
04:33:03: ISAKMP (0:13): XAUTH ACK Processed
04:33:03: ISAKMP (0:13): deleting node 524716665 error FALSE reason "done with transaction"
04:33:03: ISAKMP (0:13): Input = IKE_MESG_FROM_PEER, IKE_CFG_ACK
04:33:03: ISAKMP (0:13): Old State = IKE_XAUTH_SET_SENT New State = IKE_P1_COMPLETE
04:33:03: ISAKMP cookie gen for src 172.18.1.1 dst 10.1.1.1
04:33:03: ISAKMP cookie AA8F7B41 FDD27773
04:33:03: ISAKMP: isadb_post_process_list: crawler: B 27FF 2 (6482B354)
04:33:03: crawler my_cookie AA8F7B41 F7ACF384
04:33:03: crawler his_cookie E46E088D F227FE4D
04:33:03: ISAKMP (0:13): input packet from 10.1.1.1 dport 500 sport 500 Global (R)
04:33:03: ISAKMP cookie gen for src 172.18.1.1 dst 10.1.1.1
04:33:03: ISAKMP cookie 3123100B 7794EF6E
04:33:03: ISAKMP cookie gen for src 10.1.1.1 dst 172.18.1.1
04:33:03: ISAKMP cookie AA8F7B41 C035AAE5
04:33:03: ISAKMP: isadb_post_process_list: crawler: 9 27FF 2 (6482B354)
04:33:03: crawler my_cookie AA8F7B41 F7ACF384
04:33:03: crawler his_cookie E46E088D F227FE4D
04:33:03: ISAKMP (0:13): Input = IKE_MESG_INTERNAL, IKE_PHASE1_COMPLETE
04:33:03: ISAKMP (0:13): Old State = IKE_P1_COMPLETE New State = IKE_P2_COMPLETE
04:33:03: ISAKMP cookie gen for src 172.18.1.1 dst 10.1.1.1
04:33:03: ISAKMP cookie AA8F7B41 C05AAE5
04:33:03: ISAKMP: isadb_post_process_list: crawler: 9 27FF 2 (6482B354)
04:33:03: crawler my_cookie AA8F7B41 F7ACF384
04:33:03: crawler his_cookie E46E088D F227FE4D
04:33:03: ISAKMP (0:13): Input = IKE_MESG_INTERNAL, IKE_PHASE2_COMPLETE
04:33:03: ISAKMP (0:13): Old State = IKE_P2_COMPLETE New State = IKE_P2_COMPLETE
04:33:03: ISAKMP cookie gen for src 172.18.1.1 dst 10.1.1.1
04:33:03: ISAKMP cookie 3123100B 7794EF6D
04:33:03: ISAKMP cookie gen for src 10.1.1.1 dst 172.18.1.1
04:33:03: ISAKMP cookie 3123100B F1FC25A
04:33:03: ISAKMP cookie gen for src 10.1.1.1 dst 172.18.1.1
04:33:03: ISAKMP cookie AA8F7B41 31744F44
04:33:03: ISAKMP: isadb_post_process_list: crawler: 9 27FF 2 (6482B354)
04:33:03: crawler my_cookie AA8F7B41 F7ACF384
04:33:03: crawler his_cookie E46E088D F227FE4D
04:33:03: ISAKMP (0:13): received packet from 10.1.1.1 dport 500 sport 500 Global (R)
QM_IDLE
04:33:03: ISAKMP: set new node -1639992295 to QM_IDLE
04:33:03: ISAKMP: hash from 7293A74 for 100 bytes
04:33:03: ISAKMP: Packet hash:
64218CC0:                   9D7DF4DF FE3A6403          .}t_~:d.
64218CD0: 3F1D1C59 C5D138CE 50289B79 07        ?..YEQ8NP(.y.
04:33:03: ISAKMP (0:13): processing transaction payload from 10.1.1.1. message ID = -1639992295
04:33:03: ISAKMP (0:13): checking request:
04:33:03: ISAKMP:    IP4_ADDRESS
04:33:03: ISAKMP:    IP4_NETMASK
04:33:03: ISAKMP:    IP4_DNS
04:33:03: ISAKMP:    IP4_NBNS
04:33:03: ISAKMP:    SPLIT_INCLUDE
04:33:03: ISAKMP (0:13): Input = IKE_MESG_FROM_PEER, IKE_CFG_REQUEST
04:33:03: ISAKMP (0:13): Old State = IKE_P1_COMPLETE  New State = IKE_CONFIG_AUTHOR_AAA_AWAIT
04:33:03: ISAKMP cookie gen for src 10.1.1.1 dst 172.18.1.1
04:33:03: ISAKMP cookie AA8F7B41 B02E0D67
04:33:03: ISAKMP (0:13): got callback 1
04:33:03: ISAKMP (0:13): attributes sent in message:
04:33:03: ISAKMP (0:13): allocating address 10.4.1.4
04:33:03: ISAKMP (0:13): Sending private address: 10.4.1.4
04:33:03: ISAKMP (0:13): Sending DEFAULT_DOMAIN default domain name: vpn2.com
04:33:03: ISAKMP (0:13): sending packet to 10.1.1.1 my_port 500 peer_port 500 (R)
CONF_ADDR
04:33:03: ISAKMP (0:13): deleting node -1639992295 error FALSE reason ""
04:33:03: ISAKMP (0:13): Input = IKE_MESG_FROM_AAA, IKE_AAA_GROUP_ATTR
04:33:03: ISAKMP (0:13): Old State = IKE_CONFIG_AUTHOR_AAA_AWAIT  New State = IKE_P1_COMPLETE
04:33:03: ISAKMP cookie gen for src 172.18.1.1 dst 10.1.1.1
04:33:03: ISAKMP cookie 3123100B 881D5411
04:33:03: ISAKMP cookie gen for src 10.1.1.1 dst 172.18.1.1
04:33:03: ISAKMP cookie AA8F7B41 F3BA766D
04:33:03: ISAKMP (0:13): received packet from 10.1.1.1 dport 500 sport 500 Global (R)
QM_IDLE
04:33:03: ISAKMP: set new node 17011691 to QM_IDLE
04:33:03: ISAKMP: hash from 70029F4 for 540 bytes
04:33:03: ISAKMP: Packet hash:
64218CC0:                   AFBA30B2 55F5BC2D          /:02u<-
64218CD0: 3A86B1C9 00D2F5BA 77BF5589 07        :.1I.Ru:w?U..
04:33:03: ISAKMP (0:13): processing HASH payload. message ID = 17011691
04:33:03: ISAKMP (0:13): Checking IPSec proposal 1
04:33:03: ISAKMP (0:13): Attributes in transform:
04:33:03: ISAKMP (0:13): transform 1, ESP_3DES
04:33:03: ISAKMP (0:13): attributes in transform:
04:33:03: ISAKMP (0:13): SA life type in seconds
04:33:03: ISAKMP (0:13): SA life duration (VPLI) of 0x0 0x20 0x4C 0x9B
04:33:03: ISAKMP (0:13): SA life type in kilobytes
04:33:03: ISAKMP: isadb_post_process_list: crawler: 9 27FF 2 (6482B354)
04:33:03: crawler my_cookie AA8F7B41 F7ACF384
04:33:03: crawler his_cookie E46E088D F227FE4D
04:33:03: ISAKMP (0:13): received packet from 10.1.1.1 dport 500 sport 500 Global (R)
QM_IDLE
04:33:03: ISAKMP: set new node 17011691 to QM_IDLE
04:33:03: ISAKMP: hash from 70029F4 for 540 bytes
04:33:03: ISAKMP: Packet hash:
64218CC0:                   AFBA30B2 55F5BC2D          /:02u<-
64218CD0: 3A86B1C9 00D2F5BA 77BF5589 07        :.1I.Ru:w?U..
04:33:03: ISAKMP (0:13): processing HASH payload. message ID = 17011691
04:33:03: ISAKMP (0:13): checking request:
04:33:03: ISAKMP:    IP4_ADDRESS
04:33:03: ISAKMP:    IP4_NETMASK
04:33:03: ISAKMP:    IP4_DNS
04:33:03: ISAKMP:    IP4_NBNS
04:33:03: ISAKMP:    SPLIT_INCLUDE
04:33:03: ISAKMP (0:13): Input = IKE_MESG_FROM_PEER, IKE_CFG_REQUEST
04:33:03: ISAKMP (0:13): Old State = IKE_P1_COMPLETE  New State = IKE_CONFIG_AUTHOR_AAA_AWAIT
04:33:03: ISAKMP cookie gen for src 10.1.1.1 dst 172.18.1.1
04:33:03: ISAKMP cookie AA8F7B41 B02E0D67
04:33:03: ISAKMP (0:13): got callback 1
04:33:03: ISAKMP (0:13): attributes sent in message:
04:33:03: ISAKMP (0:13): allocating address 10.4.1.4
04:33:03: ISAKMP (0:13): Sending private address: 10.4.1.4
04:33:03: ISAKMP (0:13): Sending DEFAULT_DOMAIN default domain name: vpn2.com
04:33:03: ISAKMP (0:13): sending packet to 10.1.1.1 my_port 500 peer_port 500 (R)
04:33:03: ISAKMP:     SA life duration (VPI) of 0x0 0x46 0x50 0x0
04:33:03: ISAKMP:     authenticator is HMAC-SHA
04:33:03: ISAKMP (0:13): atts are acceptable.
04:33:03: IPSEC(validate_proposal_request): proposal part #1,
   (key eng. msg.) INBOUND local= 172.18.1.1, remote= 10.1.1.1,
   local_proxy= 0.0.0.0/0.0.0.0/0/0 (type=4),
   remote_proxy= 10.4.1.4/255.255.255.255/0/0 (type=1),
   protocol= ESP, transform= esp-3des esp-sha-hmac ,
   lifedur= 0s and 0kb, spi= 0x0(0), conn_id= 0, keysize= 0, flags= 0x2
04:33:03: IPSEC(kei_proxy): head = ra, map->ivrf = vpn1, kei->ivrf = vpn2
04:33:03: IPSEC(kei_proxy): head = ra, map->ivrf = vpn2, kei->ivrf = vpn2
04:33:03: IPSEC(validate_transform_proposal): transform proposal not supported for
   identity:
      {esp-3des esp-sha-hmac }
04:33:03: ISAKMP (0:13): IPSec policy invalidated proposal
04:33:03: ISAKMP (0:13): Checking IPSec proposal 2
04:33:03: ISAKMP: transform 1, ESP_3DES
04:33:03: ISAKMP:   attributes in transform:
04:33:03: ISAKMP:      encaps is 1
04:33:03: ISAKMP:      SA life type in seconds
04:33:03: ISAKMP:      SA life duration (VPI) of 0x0 0x20 0xC4 0x9B
04:33:03: ISAKMP:      SA life type in kilobytes
04:33:03: ISAKMP:      SA life duration (VPI) of 0x0 0x46 0x50 0x0
04:33:03: ISAKMP:      authenticator is HMAC-MD5
04:33:03: ISAKMP (0:13): atts are acceptable.
04:33:03: IPSEC(validate_proposal_request): proposal part #1,
   (key eng. msg.) INBOUND local= 172.18.1.1, remote= 10.1.1.1,
   local_proxy= 0.0.0.0/0.0.0.0/0/0 (type=4),
   remote_proxy= 10.4.1.4/255.255.255.255/0/0 (type=1),
   protocol= ESP, transform= esp-3des esp-md5-hmac ,
   lifedur= 0s and 0kb, spi= 0x0(0), conn_id= 0, keysize= 0, flags= 0x2
04:33:03: IPSEC(kei_proxy): head = ra, map->ivrf = vpn1, kei->ivrf = vpn2
04:33:03: IPSEC(kei_proxy): head = ra, map->ivrf = vpn2, kei->ivrf = vpn2
04:33:03: ISAKMP (0:13): IPsec policy invalidated proposal
04:33:03: ISAKMP (0:13): Checking IPSec proposal 2
04:33:03: ISAKMP (0:13): processing NONCE payload. message ID = 17011691
04:33:03: ISAKMP (0:13): processing ID payload. message ID = 17011691
04:33:03: ISAKMP (0:13): asking for 1 spis from ipsec
04:33:03: ISAKMP (0:13): Node 17011691, Input = IKE_MESG_FROM_PEER, IKE_QM_EXCH
04:33:03: ISAKMP (0:13): Old State = IKE_QM_READY New State = IKE_QM_SPI_STARVE
04:33:03: IPSEC(key_engine): got a queue event...
04:33:03: IPSEC(spi_response): getting spi 2749516541 for SA
from 172.18.1.1 to 10.1.1.1 for prot 3
04:33:03: ISAKMP: received ke message (2/1)
04:33:04: ISAKMP (13): ID payload
   next-payload : 5
   type         : 1
   addr         : 10.4.1.4
   protocol     : 0
   port         : 0
04:33:04: ISAKMP (13): ID payload
   next-payload : 11
   type         : 4
   addr         : 0.0.0.0
   protocol     : 0
   port         : 0
04:33:04: ISAKMP (0:13): sending packet to 10.1.1.1 my_port 500 peer_port 500 (R)
QM_IDLE
04:33:04: ISAKMP (0:13): Node 17011691, Input = IKE_MESG_FROM_IPSEC, IKE_SPI_REPLY
04:33:04: ISAKMP (0:13): Old State = IKE_QM_SPI_STARVE New State = IKE_QM_R_QM2
04:33:04: ISAKMP cookie gen for src 172.18.1.1 dst 10.1.1.1
04:33:04: ISAKMP cookie 3123100B 93DE46D2
04:33:04: ISAKMP cookie gen for src 10.1.1.1 dst 172.18.1.1
04:33:04: ISAKMP cookie AA8F7B41 088A0A16
04:33:04: ISAKMP cookie gen for src 172.18.1.1 dst 10.1.1.1
04:33:04: ISAKMP cookie 3123100B A8F23F73
04:33:04: ISAKMP cookie gen for src 10.1.1.1 dst 172.18.1.1
04:33:04: ISAKMP cookie AA8F7B41 93D8D879
04:33:04: ISAKMP:        isadb_post_process_list: crawler: 9 27FF 2 (6482B354)
04:33:04:        crawler my_cookie AA8F7B41 F7ACF384
04:33:04:        crawler his_cookie E46E088D F227FE4D
04:33:04: ISAKMP cookie gen for src 172.18.1.1 dst 10.1.1.1
04:33:04: ISAKMP cookie 3123100B A8F23F73
04:33:04: ISAKMP cookie gen for src 10.1.1.1 dst 172.18.1.1
04:33:04: ISAKMP cookie AA8F7B41 93D8D879
04:33:04: ISAKMP:        isadb_post_process_list: crawler: 9 27FF 2 (6482B354)
04:33:04: crawler my_cookie AA8F7B41 F7ACF384
04:33:04: crawler his_cookie E46E088D F227FE4D
04:33:04: ISAKMP (0:13): received packet from 10.1.1.1 dport 500 sport 500 Global (R) QM_IDLE
04:33:04: ISAKMP: hash from 7290DB4 for 60 bytes
04:33:04: ISAKMP: Packet hash:
64218CC0:                   4BB45A92 7181A2F8          K4Z.q."x
64218CD0: 73CC12F8 091875C0 054F77CD 63        sL.x..u@.OwMc
04:33:04: ISAKMP: Locking peer struct 0x640BBB18, IPSEC refcount 1 for for stuff_ke
04:33:04: ISAKMP (0:13): Creating IPSec SAs
04:33:04:         inbound SA from 10.1.1.1 to 172.18.1.1 (f/i)  0/ 2 (proxy 10.4.1.4 to 0.0.0.0)
04:33:04:         has spi 0xA3E24AFD and conn_id 5127 and flags 2
04:33:04:         lifetime of 2147483 seconds
04:33:04:         lifetime of 4608000 kilobytes
04:33:04:         has client flags 0x0
04:33:04:         outbound SA from 172.18.1.1 to 10.1.1.1 (f/i)  0/ 2 (proxy 0.0.0.0 to 10.4.1.4)
04:33:04:         has spi 1343294712 and conn_id 5128 and flags A
04:33:04:         lifetime of 2147483 seconds
04:33:04:         lifetime of 4608000 kilobytes
04:33:04:         has client flags 0x0
04:33:04: ISAKMP (0:13): deleting node 17011691 error FALSE reason "quick mode done (await)"
04:33:04: ISAKMP (0:13): Node 17011691, Input = IKE_MESG_FROM_PEER, IKE_QM_EXCH
04:33:04: ISAKMP (0:13): Old State = IKE_QM_R_QM2 New State = IKE_QM_PHASE2_COMPLETE
04:33:04: IPSEC(key_engine): got a queue event...
04:33:04: IPSEC(initialize_sas): , (key eng. msg.) INBOUND local= 172.18.1.1, remote= 10.1.1.1, local_proxy= 0.0.0.0/0.0.0.0/0/0 (type=4), remote_proxy= 10.4.1.4/0.0.0.0/0/0 (type=1), protocol= ESP, transform= esp-3des esp-md5-hmac , lifedur= 2147483s and 4608000kb, spi= 0xA3E24AFD(2749516541), conn_id= 5127, keysize= 0, flags= 0x2
04:33:04: IPSEC(initialize_sas): , (key eng. msg.) OUTBOUND local= 172.18.1.1, remote= 10.1.1.1, local_proxy= 0.0.0.0/0.0.0.0/0/0 (type=4), remote_proxy= 10.4.1.4/0.0.0.0/0/0 (type=1), protocol= ESP, transform= esp-3des esp-md5-hmac , lifedur= 2147483s and 4608000kb, spi= 0xA3E24AFD(2749516541), conn_id= 5127, keysize= 0, flags= 0x2
04:33:04: IPSEC(kei_proxy): head = ra, map->ivrf = vpn1, kei->ivrf = vpn2
04:33:04: IPSEC(kei_proxy): head = ra, map->ivrf = vpn2, kei->ivrf = vpn2
04:33:04: IPSEC(kei_proxy): head = ra, map->ivrf = vpn2, kei->ivrf = vpn2
04:33:04: IPSEC(rte_mgr): VPN Route Added 10.4.1.4 255.255.255.255 via 10.1.1.1 in vpn2
04:33:04: IPSEC(add mtree): src 0.0.0.0, dest 10.4.1.4, dest_port 0
04:33:04: IPSEC(create_sa): sa created, (sa) sa_dest= 172.18.1.1, sa_prot= 50, sa_spi= 0xA3E24AFD(2749516541), sa_trans= esp-3des esp-md5-hmac , sa_conn_id= 5127
04:33:04: IPSEC(create_sa): sa created, (sa) sa_dest= 10.1.1.1, sa_prot= 50, sa_spi= 0xA3E24AFD(2749516541), sa_trans= esp-3des esp-md5-hmac , sa_conn_id= 5127
04:33:53: ISAKMP (0:13): purging node -1639992295
04:33:54: ISAKMP (0:13): purging node 17011691

Configuration Examples for VRF-Aware IPsec

- Static IPsec-to-MPLS VPN Example, page 98
- IPsec-to-MPLS VPN Using RSA Encryption Example, page 99
- IPsec-to-MPLS VPN with RSA Signatures Example, page 101
- Upgrade from Previous Versions of the Cisco Network-Based IPsec VPN Solution, page 102
Static IPsec-to-MPLS VPN Example

The following sample shows a static configuration that maps IPsec tunnels to MPLS VPNs. The configurations map IPsec tunnels to MPLS VPNs “VPN1” and “VPN2.” Both of the IPsec tunnels terminate on a single public-facing interface.

**IPsec PE Configuration**

```plaintext
ip vrf vpn1
  rd 100:1
  route-target export 100:1
  route-target import 100:1

ip vrf vpn2
  rd 101:1
  route-target export 101:1
  route-target import 101:1

crypto keyring vpn1
  pre-shared-key address 172.16.1.1 key vpn1

crypto keyring vpn2
  pre-shared-key address 10.1.1.1 key vpn2

crypto isakmp policy 1
  encri 3des
  authentication pre-share
  group 2

crypto isakmp profile vpn1
  vrf vpn1
  keyring vpn1
  match identity address 172.16.1.1 255.255.255.255

crypto isakmp profile vpn2
  vrf vpn2
  keyring vpn2
  match identity address 10.1.1.1 255.255.255.255

crypto ipsec transform-set vpn1 esp-3des esp-sha-hmac

crypto ipsec transform-set vpn2 esp-3des esp-md5-hmac

crypto map crypmap 1 ipsec-isakmp
  set peer 172.16.1.1
  set transform-set vpn1
  set isakmp-profile vpn1
  match address 101

crypto map crypmap 3 ipsec-isakmp
  set peer 10.1.1.1
  set transform-set vpn2
  set isakmp-profile vpn2
  match address 102

interface Ethernet1/1
  ip address 172.17.1.1 255.255.0.0
tag-switching ip

interface Ethernet1/2
  ip address 172.18.1.1 255.255.255.0

crypto map crypmap

ip route 172.16.1.1 255.255.255.255 172.18.1.2
ip route 10.1.1.1 255.255.255.255 172.18.1.2
ip route vrf vpn1 10.2.0.0 255.255.0.0 172.18.1.2 global
ip route vrf vpn2 10.2.0.0 255.255.0.0 172.18.1.2 global

access-list 101 permit ip 10.1.0.0 0.0.255.255 10.2.0.0 0.0.255.255
access-list 102 permit ip 10.1.0.0 0.0.255.255 10.2.0.0 0.0.255.255
```
IPsec Customer Provided Edge (CPE) Configuration for VPN1

crypto isakmp policy 1
  encr 3des
  authentication pre-share
  group 2
  crypto isakmp key vpn1 address 172.18.1.1
  !
  crypto ipsec transform-set vpn1 esp-3des esp-sha-hmac
  !
  crypto map vpn1 1 ipsec-isakmp
  set peer 172.18.1.1
  set transform-set vpn1
  match address 101
  !
  interface FastEthernet1/0
  ip address 172.16.1.1 255.255.255.0
  crypto map vpn1
  !
  interface FastEthernet1/1
  ip address 10.2.1.1 255.255.0.0
  !
  access-list 101 permit ip 10.2.0.0 0.0.255.255 10.1.0.0 0.0.255.255
  !

IPsec CPE Configuration for VPN2

crypto isakmp policy 1
  encr 3des
  authentication pre-share
  group 2
  !
  crypto isakmp key vpn2 address 172.18.1.1
  !
  crypto ipsec transform-set vpn2 esp-3des esp-md5-hmac
  !
  crypto map vpn2 1 ipsec-isakmp
  set peer 172.18.1.1
  set transform-set vpn2
  match address 101
  !
  interface FastEthernet0
  ip address 10.1.1.1 255.255.255.0
  crypto map vpn2
  !
  interface FastEthernet1
  ip address 10.2.1.1 255.255.0.0
  !
  access-list 101 permit ip 10.2.0.0 0.0.255.255 10.1.0.0 0.0.255.255

IPsec-to-MPLS VPN Using RSA Encryption Example

The following example shows an IPsec-to-MPLS configuration using RSA encryption:

PE Router Configuration

ip vrf vpn1
  rd 100:1
  route-target export 100:1
  route-target import 100:1
  !
  crypto isakmp policy 10
  authentication rsa-encr
  !
  crypto keyring vpn1
rsa-pubkey address 172.16.1.1 encryption
key-string 305C300D 06092A86 4886F70D 01010105 00034B00 30480241 00DBF381 00DD0CC8
DC4A490 40320C52 99128276 EB6F717C 63DCA95C 7E5EC0A2 84F276CE 292B42D7
D664F324 3726F4E0 39D33039 ECB81B95 482511A5 F064C4B3 D5020301 0001 quit

! crypto isakmp profile vpn1
  vrf vpn1
  keyring vpn1
  match identity address 172.16.1.1 255.255.255.255
! crypto ipsec transform-set vpn1 esp-3des esp-sha-hmac
! crypto map cryptomap 1 ipsec-isakmp
  set peer 172.16.1.1
  set transform-set vpn1
  set isakmp-profile vpn1
  match address 101
! interface Ethernet1/1
  ip address 172.17.1.1 255.255.0.0
tag-switching ip
! interface Ethernet1/2
  ip address 172.18.1.1 255.255.255.0
crypto map cryptomap
  ip route 172.16.1.1 255.255.255.255 172.18.1.2
  ip route vrf vpn1 10.2.0.0 255.255.0.0 172.18.1.2 global
! access-list 101 permit ip 10.1.0.0 0.0.255.255
10.2.0.0 0.0.255.255

IPsec CPE Configuration for VPN1

crypto isakmp policy 10
  authentication rsa-enctr
! crypto key pubkey-chain rsa
  addressed-key 172.18.11.1 encryption
  key-string 3082011B 300D0609 2A864886 F70D0101 01050003 82010800 30820103 0281FB00
  C90CC7BA 6002B0BA 24683396 B7D8477C 16D0B8C47 B00C3C10 63CF13BC 4E09EA23
  992BB8A8 411F35A4 8796CB8E AD7EDC1 3B072B86 71B8CE7C BA082D11 AA9724A4
  4D7FCEA5 562F2225 A2B11F18 E53C4415 61C3B741 3A06E75D B4F9102D 6163EE40
  16C68FD7 6532F660 97B59118 9C8DE3E5 4E2F2925 B8BBB7FC 95223D4E A5E362DB
  215CB35C 26008B05 17B9E1EF C0350E13 031F3D05 5C22D1EC FC8B1E0C 764F70A5
  D050E8C0 7890D9C5 EC20DDF0 173FE2B8 89F5B5F9 2EDAC5BE D461921E 3D5B6016
  ABB86BE9 E2124A21 93F0E8AE B487461B E7F1F1C4 032A0B0E 80DC3E15 CB268EC9
  5D7689DB 3C78CB75 CE9F68C6 484D6573 CBC3EB59 4B5F3999 8F9D0203 010001 quit
! crypto ipsec transform-set vpn1 esp-3des esp-sha-hmac
! crypto map vpng 1 ipsec-isakmp
  set peer 172.18.1.1
  set transform-set vpn1
  match address 101
! interface FastEthernet1/0
  ip address 172.16.1.1 255.255.255.0
crypto map vpn1
! interface FastEthernet1/1
  ip address 10.2.1.1 255.255.0.0
! access-list 101 permit ip 10.2.0.0 0.0.255.255
10.1.0.0 0.0.255.255
IPsec-to-MPLS VPN with RSA Signatures Example

The following shows an IPsec-to-MPLS VPN configuration using RSA signatures:

**PE Router Configuration**

```
pe vrf vpn1
  rd 100:1
  route-target export 100:1
  route-target import 100:1
!
crypto ca trustpoint bombo
  enrollment url http://172.31.68.59:80
crl optional
!
crypto ca certificate chain bombo
  certificate 03C0
  308203BF 308202A7 A0030201 02020203 C0300D06 092A8648 86F7D0D1 01050500
  ...
  quit
!
certificate ca 01
  30820379 30820261 A0030201 02020101 300D0609 2A864886 F7D0D101 05050030
  ...
  quit
!
crypto isakmp profile vpn1
  vrf vpn1
crypto ca trustpoint bombo
  match identity address 172.16.1.1 255.255.255.255
!
crypto ipsec transform-set vpn1 esp-3des esp-sha-hmac
!
crypto map crypmap 1 ipsec-isakmp
  set peer 172.16.1.1
  set transform-set vpn1
  set isakmp-profile vpn1
  match address 101
!
interface Ethernet1/1
  ip address 172.31.1.1 255.255.0.0
tag-switching ip
!
interface Ethernet1/2
  ip address 172.18.1.1 255.255.255.0
crypto map crypmap
!
ip route 172.16.1.1 255.255.255.255 172.18.1.2

```

**IPsec CPE Configuration for VPN1**

```
crypto ca trustpoint bombo
  enrollment url http://172.31.68.59:80
crl optional
!
crypto ca certificate chain bombo
  certificate 03BF
  308203BD 308202A5 A0030201 02020203 BF300D06 092A8648 86F7D0D1 01050500
  ...
  quit
!
certificate ca 01
  30820379 30820261 A0030201 02020101 300D0609 2A864886 F7D0D101 05050030
  ...
  quit
!```
Upgrade from Previous Versions of the Cisco Network-Based IPsec VPN Solution

The VRF-Aware IPSec feature in the Cisco network-based IPsec VPN solution release 1.5 requires that you change your existing configurations.

The following sample configuration indicates the changes you must make to your existing configuration:

- Site-to-Site Configuration Upgrade, page 102

Site-to-Site Configuration Upgrade

The following configurations show the changes that are necessary for a site-to-site configuration upgrade from a previous version of the network-based IPsec VPN solution to the Cisco network-based IPsec VPN solution release 1.5:

- Previous Version Site-to-Site Configuration, page 102
- New Version Site-to-Site Configuration, page 103

Previous Version Site-to-Site Configuration

crypto isakmp key VPN1 address 172.21.25.74
crypto isakmp key VPN2 address 172.21.21.74

crypto ipsec transform-set VPN1 esp-des esp-sha-hmac
crypto ipsec transform-set VPN2 esp-3des esp-sha-hmac

crypto map VPN1 10 ipsec-isakmp
set peer 172.21.25.74
set transform-set VPN1
match address 101

crypto map VPN2 10 ipsec-isakmp
set peer 172.21.21.74
set transform-set VPN2
match address 102

interface FastEthernet0/0.1
capsulation dot1Q 1 native
ip vrf forwarding VPN1
ip address 172.21.25.73 255.255.255.0

crypto map VPN1

interface FastEthernet0/0.2
capsulation dot1Q 2 native
ip vrf forwarding VPN2
New Version Site-to-Site Configuration

The following is an upgraded version of the same site-to-site configuration to the Cisco network-based IPsec VPN solution release 1.5 solution:

Note

You must change to keyrings. The VRF-Aware IPsec feature requires that keys be associated with a VRF if the IKE local endpoint is in the VRF.

crypto keyring VPN1-KEYS vrf VPN1
pre-shared-key address 172.21.25.74 key VPN1
!
crypto keyring VPN2-KEYS vrf VPN2
pre-shared-key address 172.21.21.74 key VPN2
!
crypto ipsec transform-set VPN1 esp-des esp-sha-hmac
crypto ipsec transform-set VPN2 esp-3des esp-sha-hmac
!
crypto map VPN1 10 ipsec-isakmp
set peer 172.21.25.74
set transform-set VPN1
match address 101
!
crypto map VPN2 10 ipsec-isakmp
set peer 172.21.21.74
set transform-set VPN2
match address 102
!
interface FastEthernet0/0.1
encapsulation dot1Q 1 native
ip vrf forwarding VPN1
ip address 172.21.25.73 255.255.255.0
crypto map VPN1
!
interface FastEthernet0/0.2
encapsulation dot1Q 2 native
ip vrf forwarding VPN2
ip address 172.21.21.74 255.255.255.0
crypto map VPN2

Additional References

The following sections provide references related to the VRF-Aware IPsec feature.

Related Documents

<table>
<thead>
<tr>
<th>Related Topic</th>
<th>Document Title</th>
</tr>
</thead>
<tbody>
<tr>
<td>IPsec configuration tasks</td>
<td>The chapter “Configuring Security for VPNs with IPsec” in the Cisco IOS Security Configuration Guide: Secure Connectivity</td>
</tr>
<tr>
<td>IPsec commands</td>
<td>Cisco IOS Security Command Reference</td>
</tr>
</tbody>
</table>
### Related Topic

- IKE Phase 1 and Phase 2, aggressive mode, and main mode
- IKE dead peer detection

### Document Title

- The chapter “Configuring Internet Key Exchange for IPsec VPNs” in the *Cisco IOS Security Configuration Guide: Secure Connectivity*
- Easy VPN Server

### MIBs

<table>
<thead>
<tr>
<th>MIB</th>
<th>MIBs Link</th>
</tr>
</thead>
<tbody>
<tr>
<td>None</td>
<td>To locate and download MIBs for selected platforms, Cisco IOS XE software releases, and feature sets, use Cisco MIB Locator found at the following URL: <a href="http://www.cisco.com/go/mibs">http://www.cisco.com/go/mibs</a></td>
</tr>
</tbody>
</table>

### Technical Assistance

**Description**

The Cisco Support website provides extensive online resources, including documentation and tools for troubleshooting and resolving technical issues with Cisco products and technologies.

To receive security and technical information about your products, you can subscribe to various services, such as the Product Alert Tool (accessed from Field Notices), the Cisco Technical Services Newsletter, and Really Simple Syndication (RSS) Feeds.

Access to most tools on the Cisco Support website requires a Cisco.com user ID and password.

**Link**


### Feature Information for VRF-Aware IPsec

The following table provides release information about the feature or features described in this module. This table lists only the software release that introduced support for a given feature in a given software release train. Unless noted otherwise, subsequent releases of that software release train also support that feature.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to www.cisco.com/go/cfn. An account on Cisco.com is not required.
Table 6  Feature Information for VRF-Aware IPsec

<table>
<thead>
<tr>
<th>Feature Name</th>
<th>Releases</th>
<th>Feature Information</th>
</tr>
</thead>
<tbody>
<tr>
<td>VRF-Aware IPsec</td>
<td>XE 2.6</td>
<td>The VRF-Aware IPsec feature introduces IP Security (IPsec) tunnel mapping to Multiprotocol Label Switching (MPLS) Virtual Private Networks (VPNs). Using the VRF-Aware IPsec feature, you can map IPsec tunnels to Virtual Routing and Forwarding (VRF) instances using a single public-facing address. This feature was introduced in Cisco IOS Release 12.2(15)T. The following commands were introduced or modified: address, ca trust-point, client authentication list, client configuration address, crypto isakmp profile, crypto keyring, crypto map isakmp-profile, initiate-mode, isakmp authorization list, keepalive (isakmp profile), keyring, key-string, match identity, no crypto xauth, pre-shared-key, quit, rsa-pubkey, self-identity, serial-number, set isakmp-profile, show crypto isakmp key, show crypto isakmp profile, vrf, clear crypto sa, crypto isakmp peer, crypto map isakmp-profile, show crypto ipsec sa, show crypto isakmp sa, show crypto map (IPsec).</td>
</tr>
</tbody>
</table>

Glossary

**CA** -- certification authority. CA is an entity that issues digital certificates (especially X.509 certificates) and vouches for the binding between the data items in a certificate.

**CLI** -- command-line-interface. CLI is an interface that allows the user to interact with the operating system by entering commands and optional arguments. The UNIX operating system and DOS provide CLIs.

**client** -- Corresponding IPsec IOS peer of the UUT in the Multi Protocol Label Switching (MPLS) network.

**dead peer** -- IKE peer that is no longer reachable.
DN -- Distinguished Name. A DN is the global, authoritative name of an entry in the Open System Interconnection (OSI Directory [X.500]).

FQDN -- fully qualified domain name. A FQDN is the full name of a system rather than just its host name. For example, aldebaran is a host name, and aldebaran.interop.com is an FQDN.

FR -- Frame Relay. FR is an industry-standard, switch-data-link-layer protocol that handles multiple virtual circuits using high-level data link (HDLC) encapsulation between connected devices. Frame Relay is more efficient than X.25, the protocol for which it generally is considered a replacement.

FVRF -- Front Door Virtual Routing and Forwarding (VRF) repository. FVRF is the VRF used to route the encrypted packets to the peer.

IDB -- Interface descriptor block. An IDB subblock is an area of memory that is private to an application. This area stores private information and states variables that an application wants to associate with an IDB or an interface. The application uses the IDB to register a pointer to its subblock, not to the contents of the subblock itself.

IKE -- Internet Key Exchange. IKE establishes a shared security policy and authenticates keys for services (such as IPsec) that require keys. Before any IPsec traffic can be passed, each router, firewall, and host must verify the identity of its peer. This can be done by manually entering preshared keys into both hosts or by a CA service.

IKE keepalive -- Bidirectional mechanism for determining the liveliness of an IKE peer.

IPsec -- Security protocol for IP.

IVRF -- Inside Virtual Routing and Forwarding. IVRF is the VRF of the plaintext packets.

MPLS -- Multiprotocol Label Switching. MPLS is a switching method that forwards IP traffic using a label. This label instructs the routers and the switches in the network where to forward the packets based on preestablished IP routing information.

RSA -- Rivest, Shamir, and Adelman are the inventors of the RSA technique. The RSA technique is a public-keycryptographic system that can be used for encryption and authentication.

SA -- Security Association. SA is an instance of security policy and keying material applied to a data flow.

VPN -- Virtual Private Network. A VPN enables IP traffic to travel securely over a public TCP or IP network by encrypting all traffic from one network to another. A VPN uses "tunneling" to encrypt all information at the IP level.

VRF -- Virtual Route Forwarding. VRF is a VPN routing and forwarding instance. A VRF consists of an IP routing table, a derived forwarding table, a set of interfaces that use the forwarding table, and a set of rules and routing protocols that determine what goes into the forwarding table. In general, a VRF includes the routing information that defines a customer VPN site that is attached to a PE router.

XAUTH -- Extended authentication. XAUTH is an optional exchange between IKE Phase 1 and IKE Phase 2, in which the router demands additional authentication information in an attempt to authenticate the actual user (as opposed to authenticating the peer).
and other figures included in the document are shown for illustrative purposes only. Any use of actual IP addresses or phone numbers in illustrative content is unintentional and coincidental.
IKE Initiate Aggressive Mode

The IKE: Initiate Aggressive Mode feature allows you to specify RADIUS tunnel attributes for an IP security (IPsec) peer and to initiate an Internet Key Exchange (IKE) aggressive mode negotiation with the tunnel attributes. This feature is best implemented in a crypto hub-and-spoke scenario, by which the spokes initiate IKE aggressive mode negotiation with the hub by using the preshared keys that are specified as tunnel attributes and stored on the AAA server. This scenario is scalable because the preshared keys are kept at a central repository (the AAA server) and more than one hub router and one application can use the information.

- Finding Feature Information, page 109
- Prerequisites for IKE Initiate Aggressive Mode, page 109
- Restrictions for IKE Initiate Aggressive Mode, page 110
- Information About IKE Initiate Aggressive Mode, page 110
- How to Configure IKE Initiate Aggressive Mode, page 110
- Configuration Examples for IKE Initiate Aggressive Mode, page 113
- Additional References, page 114
- Feature Information for IKE Initiate Aggressive Mode, page 115

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest feature information and caveats, see the release notes for your platform and software release. To find information about the features documented in this module, and to see a list of the releases in which each feature is supported, see the Feature Information Table at the end of this document.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to www.cisco.com/go/cfn. An account on Cisco.com is not required.

Prerequisites for IKE Initiate Aggressive Mode

Before configuring the Initiate Aggressive Mode IKE feature, you must perform the following tasks:

- Configure AAA
- Configure an IPsec Transform
- Configure a static crypto map
- Configure an Internet Security Association and Key Management Protocol (ISAKMP) policy
- Configure a dynamic crypto map
Restrictions for IKE Initiate Aggressive Mode

TED Restriction
This feature is not intended to be used with a dynamic crypto map that uses Tunnel Endpoint Discovery (TED) to initiate tunnel setup. TED is useful in configuring a full mesh setup, which requires an AAA server at each site to store the preshared keys for the peers; this configuration is not practical for use with this feature.

Tunnel-Client-Endpoint ID Types
Only the following ID types can be used in this feature:
- ID_IPV4 (IPV4 address)
- ID_FQDN (fully qualified domain name, for example “foo.cisco.com”)
- ID_USER_FQDN (e-mail address)

Information About IKE Initiate Aggressive Mode

- Overview, page 110
- RADIUS Tunnel Attributes, page 110

Overview
The IKE: Initiate Aggressive Mode feature allows you to configure IKE preshared keys as RADIUS tunnel attributes for IPsec peers. Thus, you can scale your IKE preshared keys in a hub-and-spoke topology.

Although IKE preshared keys are simple to understand and easy to deploy, they do not scale well with an increasing number of users and are therefore prone to security threats. Instead of keeping your preshared keys on the hub router, this feature allows you to scale your preshared keys by storing and retrieving them from an authentication, authorization, and accounting (AAA) server. The preshared keys are stored in the AAA server as Internet Engineering Task Force (IETF) RADIUS tunnel attributes and are retrieved when a user tries to “speak” to the hub router. The hub router retrieves the preshared key from the AAA server and the spokes (the users) initiate aggressive mode to the hub by using the preshared key that is specified in the Internet Security Association Key Management Policy (ISAKMP) peer policy as a RADIUS tunnel attribute.

RADIUS Tunnel Attributes
To initiate an IKE aggressive mode negotiation, the Tunnel-Client-Endpoint (66) and Tunnel-Password (69) attributes must be configured in the ISAKMP peer policy. The Tunnel-Client-Endpoint attribute will be communicated to the server by encoding it in the appropriate IKE identity payload; the Tunnel-Password attribute will be used as the IKE preshared key for the aggressive mode negotiation.

How to Configure IKE Initiate Aggressive Mode

- Configuring RADIUS Tunnel Attributes, page 111
• Verifying RADIUS Tunnel Attribute Configurations, page 112
• Troubleshooting Tips, page 112

Configuring RADIUS Tunnel Attributes

To configure the Tunnel-Client-Endpoint and Tunnel-Password attributes within the ISAKMP peer configuration, perform the following steps.

**SUMMARY STEPS**

1. enable
2. configure terminal
3. crypto map map-name isakmp authorization list list-name
4. crypto isakmp peer {ip-address ip-address | fqdn fqdn}
5. set aggressive-mode client-endpoint client-endpoint
6. set aggressive-mode password password

**DETAILED STEPS**

<table>
<thead>
<tr>
<th>Command or Action</th>
<th>Purpose</th>
</tr>
</thead>
</table>
| **Step 1** enable | Enables privileged EXEC mode.  
• Enter your password if prompted. |
| **Example:** Router> enable | |
| **Step 2** configure terminal | Enters global configuration mode. |
| **Example:** Router# configure terminal | |
| **Step 3** crypto map map-name isakmp authorization list list-name | Enables IKE querying of AAA for tunnel attributes in aggressive mode. |
| **Example:** Router (config)# crypto map testmap10 isakmp authorization list list ike | |
| **Step 4** crypto isakmp peer {ip-address ip-address | fqdn fqdn} | Enables an IPsec peer for IKE querying of AAA for tunnel attributes in aggressive mode and enters ISAKMP policy configuration mode. |
| **Example:** Router (config)# crypto isakmp peer ip address 10.10.10.1 | |
### Command or Action

<table>
<thead>
<tr>
<th>Step 5 set aggressive-mode client-endpoint client-endpoint</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>Specifies the Tunnel-Client-Endpoint attribute within an ISAKMP peer configuration.</td>
<td></td>
</tr>
</tbody>
</table>

**Example:**

```
Router (config-isakmp)# set aggressive-mode client-endpoint user-fqdn user@cisco.com
```

<table>
<thead>
<tr>
<th>Step 6 set aggressive-mode password password</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>Specifies the Tunnel-Password attribute within an ISAKMP peer configuration.</td>
<td></td>
</tr>
</tbody>
</table>

**Example:**

```
Router (config-isakmp)#set aggressive-mode password cisco123
```

---

### Verifying RADIUS Tunnel Attribute Configurations

To verify that the Tunnel-Client-Endpoint and Tunnel-Password attributes have been configured within the ISAKMP peer policy, use the `show running-config` global configuration command.

### Troubleshooting Tips

To troubleshoot the IKE: Initiate Aggressive Mode feature, perform the following steps.

#### SUMMARY STEPS

1. `enable`
2. `debug aaa authorization`
3. `debug crypto isakmp`
4. `debug radius`

#### DETAILED STEPS

<table>
<thead>
<tr>
<th>Command or Action</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Step 1 enable</strong></td>
<td>Enables privileged EXEC mode.</td>
</tr>
<tr>
<td>Example:</td>
<td>Enter your password if prompted.</td>
</tr>
</tbody>
</table>

```
Router> enable
```

<table>
<thead>
<tr>
<th><strong>Step 2 debug aaa authorization</strong></th>
<th>Displays information about AAA authorization.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Example:</td>
<td></td>
</tr>
</tbody>
</table>

```
Router# debug aaa authorization
```
### Command or Action | Purpose
---|---
**Step 3** debug crypto isakmp | Displays messages about IKE events.

**Example:**
Router# debug crypto isakmp

**Step 4** debug radius | Displays information associated with RADIUS.

**Example:**
Router# debug radius

### Configuration Examples for IKE Initiate Aggressive Mode

- Hub Configuration Example, page 113
- Spoke Configuration Example, page 114
- RADIUS User Profile Example, page 114

### Hub Configuration Example

The following example shows how to configure a hub for a hub-and-spoke topology that supports aggressive mode using RADIUS tunnel attributes:

```
!The AAA configurations are as follows:
aaa new-model
aaa authorization network ike group radius
aaa authentication login default group radius
!
! The Radius configurations are as follows:
radius-server host 10.1.1.1 auth-port 1645 acct-port 1646
radius-server key rad123
!
! The IKE configurations are as follows:
crypto isakmp policy 1
    authentication pre-share
!
! The IPsec configurations are as follows:
crypto ipsec transform-set trans1 esp-3des esp-sha-hmac
!
crypto dynamic-map Dmap 10
    set transform-set trans1
!
crypto map Testtag isakmp authorization list ike
crypto map Testtag 10 ipsec-isakmp dynamic Dmap
!
interface FastEthernet0
    ip address 10.4.4.1 255.255.255.0
    crypto map Testtag
!
interface FastEthernet1
    ip address 10.2.2.1 255.255.255.0
```
Spoke Configuration Example

The following example shows how to configure a spoke for a hub-and-spoke topology that supports aggressive mode using RADIUS tunnel attributes:

```plaintext
!The IKE configurations are as follows:
crypto isakmp policy 1
  authentication pre-share
!
! The IPsec configurations are as follows:
crypto ipsec transform-set trans1 esp-3des esp-sha-hmac
  access-list 101 permit ip 10.3.3.0 0.0.0.255 10.2.2.0 0.0.0.255
!
! Initiate aggressive mode using Radius tunnel attributes
crypto isakmp peer address 10.4.4.1
  set aggressive-mode client-endpoint user-fqdn user@cisco.com
  set aggressive-mode password cisco123
!
crypto map Testtag 10 ipsec-isakmp
  set peer 10.4.4.1
  set transform-set trans1
  match address 101
!
interface FastEthernet0
  ip address 10.5.5.1 255.255.255.0
  crypto map Testtag
!
interface FastEthernet1
  ip address 10.3.3.1 255.255.255.0
```

RADIUS User Profile Example

The following is an example of a user profile on a RADIUS server that supports the Tunnel-Client-Endpoint and Tunnel-Password attributes:

```
user@cisco.com  Password = "cisco", Service-Type = Outbound
  Tunnel-Medium-Type = :1:IP,
  Tunnel-Type = :1:ESP,
  Cisco:Avpair = "ipsec:tunnel-password=cisco123",
  Cisco:Avpair = "ipsec:key-exchange=ike"
```

Additional References

The following sections provide references related to the IKE: Initiate Aggressive Mode feature.

<table>
<thead>
<tr>
<th>Related Topic</th>
<th>Document Title</th>
</tr>
</thead>
<tbody>
<tr>
<td>Security commands</td>
<td>Cisco IOS Security Command Reference</td>
</tr>
<tr>
<td>Configuring authentication</td>
<td>Configuring Authentication</td>
</tr>
<tr>
<td>Configuring IKE</td>
<td>Configuring Internet Key Exchange for IPsec VPNs</td>
</tr>
</tbody>
</table>
Standards

<table>
<thead>
<tr>
<th>Standard</th>
<th>Title</th>
</tr>
</thead>
<tbody>
<tr>
<td>No new or modified standards are supported by this feature, and support for existing standards has not been modified by this feature.</td>
<td>--</td>
</tr>
</tbody>
</table>

MIBs

<table>
<thead>
<tr>
<th>MIB</th>
<th>MIBs Link</th>
</tr>
</thead>
<tbody>
<tr>
<td>No new or modified MIBs are supported by this feature, and support for existing MIBs has not been modified by this feature.</td>
<td>To locate and download MIBs for selected platforms, Cisco IOS XE software releases, and feature sets, use Cisco MIB Locator found at the following URL:  <a href="http://www.cisco.com/go/mibs">http://www.cisco.com/go/mibs</a></td>
</tr>
</tbody>
</table>

RFCs

<table>
<thead>
<tr>
<th>RFC</th>
<th>Title</th>
</tr>
</thead>
<tbody>
<tr>
<td>RFC 2409</td>
<td>RFC 2409, <em>The Internet Key Exchange</em></td>
</tr>
<tr>
<td>RFC 2868</td>
<td>RFC 2868, <em>RADIUS Attributes for Tunnel Protocol Support</em></td>
</tr>
</tbody>
</table>

Technical Assistance

<table>
<thead>
<tr>
<th>Description</th>
<th>Link</th>
</tr>
</thead>
<tbody>
<tr>
<td>The Cisco Support website provides extensive online resources, including documentation and tools for troubleshooting and resolving technical issues with Cisco products and technologies. To receive security and technical information about your products, you can subscribe to various services, such as the Product Alert Tool (accessed from Field Notices), the Cisco Technical Services Newsletter, and Really Simple Syndication (RSS) Feeds. Access to most tools on the Cisco Support website requires a Cisco.com user ID and password.</td>
<td><a href="http://www.cisco.com/techsupport">http://www.cisco.com/techsupport</a></td>
</tr>
</tbody>
</table>

Feature Information for IKE Initiate Aggressive Mode

The following table provides release information about the feature or features described in this module. This table lists only the software release that introduced support for a given feature in a given software release train. Unless noted otherwise, subsequent releases of that software release train also support that feature.
Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to www.cisco.com/go/cfn. An account on Cisco.com is not required.

Table 7  Feature Information for IKE: Initiate Aggressive Mode

<table>
<thead>
<tr>
<th>Feature Name</th>
<th>Releases</th>
<th>Feature Information</th>
</tr>
</thead>
<tbody>
<tr>
<td>IKE: Initiate Aggressive Mode</td>
<td>Cisco IOS XE Release 2.1</td>
<td>The IKE: Initiate Aggressive Mode feature allows you to specify RADIUS tunnel attributes for an IPsec peer and to initiate an IKE aggressive mode negotiation with the tunnel attributes. The following commands were introduced or modified: <code>crypto isakmp peer</code>, <code>set aggressive-mode client-endpoint</code>, <code>set aggressive-mode password</code>.</td>
</tr>
</tbody>
</table>

Cisco and the Cisco logo are trademarks or registered trademarks of Cisco and/or its affiliates in the U.S. and other countries. To view a list of Cisco trademarks, go to this URL: www.cisco.com/go/trademarks. Third-party trademarks mentioned are the property of their respective owners. The use of the word partner does not imply a partnership relationship between Cisco and any other company. (1110R)

Any Internet Protocol (IP) addresses and phone numbers used in this document are not intended to be actual addresses and phone numbers. Any examples, command display output, network topology diagrams, and other figures included in the document are shown for illustrative purposes only. Any use of actual IP addresses or phone numbers in illustrative content is unintentional and coincidental.