CONTENTS

CHAPTER 1  
Read Me First  1

CHAPTER 2  
IPsec Anti-Replay Window Expanding and Disabling  3
  Finding Feature Information  3
  Prerequisites for IPsec Anti-Replay Window Expanding and Disabling  4
  Information About IPsec Anti-Replay Window Expanding and Disabling  4
  IPsec Anti-Replay Window  4
  How to Configure IPsec Anti-Replay Window Expanding and Disabling  4
    Configuring IPsec Anti-Replay Window Expanding and Disabling Globally  4
    Configuring IPsec Anti-Replay Window Expanding and Disabling on a Crypto Map  5
    Troubleshooting Tips  7
  Configuration Examples for IPsec Anti-Replay Window Expanding and Disabling  7
    Global Expanding and Disabling of an Anti-Replay Window Example  7
    Expanding and Disabling of an Anti-Replay Window for Crypto Maps or Crypto Profiles Example  8
  Additional References  9
  Feature Information for IPsec Anti-Replay Window Expanding and Disabling  10

CHAPTER 3  
Pre-Fragmentation for IPsec VPNs  11
  Finding Feature Information  11
  Restrictions for Pre-Fragmentation for IPsec VPNs  11
  Information About Pre-Fragmentation for IPsec VPNs  13
    Pre-fragmentation for IPsec VPNs  13
  How to Configure Pre-Fragmentation for IPsec VPNs  13
    Configuring Pre-Fragmentation for IPsec VPNs  13
  Additional References  14
  Feature Information for Pre-Fragmentation for IPsec VPNs  15
CHAPTER 7  DF Bit Override Functionality with IPsec Tunnels  59

Finding Feature Information  59
Prerequisites for DF Bit Override Functionality with IPsec Tunnels  59
Restrictions for DF Bit Override Functionality with IPsec Tunnels  60
Information About DF Bit Override Functionality with IPsec Tunnels  60
Feature Overview  60
How to Configure DF Bit Override Functionality with IPsec Tunnels  61
Configuring the DF Bit for the Encapsulating Header in Tunnel Mode  61
Verifying DF Bit Setting  61
Configuration Examples for DB Bit Override Functionality with IPsec Tunnels  62
DF Bit Setting Configuration Example  62
Additional References  62
Related Documents  63
Standards  63
MIBs  63
RFCs  63
Technical Assistance  64
Feature Information for DF Bit Override Functionality with IPsec Tunnels  64

CHAPTER 8  IPsec Security Association Idle Timers  65

Finding Feature Information  65
Prerequisites for IPsec Security Association Idle Timers  66
Information About IPsec Security Association Idle Timers  66
Lifetimes for IPsec Security Associations  66
IPsec Security Association Idle Timers  66
How to Configure IPsec Security Association Idle Timers  66
Configuring the IPsec SA Idle Timer Globally  66
Configuring the IPsec SA Idle Timer per Crypto Map  67
Configuration Examples for IPsec Security Association Idle Timers  68
CHAPTER 9  IPv6 IPsec Quality of Service  73
  Finding Feature Information  73
  Information About IPv6 IPsec QoS  73
    IPv6 IPsec QoS Overview  73
  How to Configure IPv6 IPsec QoS  74
    Configuring Crypto LLQ QoS  74
    Configuring QoS Pre-classify  75
      Configuring Pre-classify on the Crypto Map  75
      Configuring Pre-classify on the Tunnel Interface  76
    Configuring LLQ QoS Group  78
  Configuration Examples for QoS  79
    Example: Configuring Crypto LLQ QoS  79
    Example: Configuring Pre-classify on the Crypto Map  79
    Example: Configuring Pre-classify on the Tunnel Interface  79
    Example: Configuring LLQ QoS Group  80
  Additional References for IPv6 IPsec QoS  80
  Feature Information for IPv6 IPsec QoS  81

CHAPTER 10  IPv6 Virtual Tunnel Interface  83
  Finding Feature Information  83
  Information About IPv6 Virtual Tunnel Interface  84
    IPsec for IPv6  84
      IPv6 IPsec Site-to-Site Protection Using Virtual Tunnel Interface  84
  How to Configure IPv6 Virtual Tunnel Interface  85
    Configuring a VTI for Site-to-Site IPv6 IPsec Protection  85
      Defining an IKE Policy and a Preshared Key in IPv6  85
      Configuring ISAKMP Aggressive Mode  89
      Defining an IPsec Transform Set and IPsec Profile  90
      Defining an ISAKMP Profile in IPv6  91
      Configuring IPv6 IPsec VTI  92
Verifying IPsec Tunnel Mode Configuration 95
Troubleshooting IPsec for IPv6 Configuration and Operation 97
Configuration Examples for IPv6 Virtual Tunnel Interface 98
Example: Configuring a VTI for Site-to-Site IPv6 IPsec Protection 98
Additional References 98
Feature Information for IPv6 Virtual Tunnel Interface 99
Read Me First

Important Information about Cisco IOS XE 16

Effective Cisco IOS XE Release 3.7.0E (for Catalyst Switching) and Cisco IOS XE Release 3.17S (for Access and Edge Routing) the two releases evolve (merge) into a single version of converged release—the Cisco IOS XE 16—providing one release covering the extensive range of access and edge products in the Switching and Routing portfolio.

Note

The Feature Information table in the technology configuration guide mentions when a feature was introduced. It may or may not mention when other platforms were supported for that feature. To determine if a particular feature is supported on your platform, look at the technology configuration guides posted on your product landing page. When a technology configuration guide is displayed on your product landing page, it indicates that the feature is supported on that platform.
 CHAPTER 2

IPsec Anti-Replay Window Expanding and Disabling

Cisco IP security (IPsec) authentication provides anti-replay protection against an attacker duplicating encrypted packets by assigning a unique sequence number to each encrypted packet. The decryptor keeps track of which packets it has seen on the basis of these numbers. Currently, the default window size is 64 packets. Generally, this number (window size) is sufficient, but there are times when you may want to expand this window size. The IPsec Anti-Replay Window: Expanding and Disabling feature allows you to expand the window size, allowing the decryptor to keep track of more than 64 packets.

- Finding Feature Information, page 3
- Prerequisites for IPsec Anti-Replay Window Expanding and Disabling, page 4
- Information About IPsec Anti-Replay Window Expanding and Disabling, page 4
- How to Configure IPsec Anti-Replay Window Expanding and Disabling, page 4
- Configuration Examples for IPsec Anti-Replay Window Expanding and Disabling, page 7
- Additional References, page 9
- Feature Information for IPsec Anti-Replay Window Expanding and Disabling, page 10

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest caveats and feature information, see Bug Search Tool and 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 module.

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 IPsec Anti-Replay Window Expanding and Disabling

- Before configuring this feature, you should have already created a crypto map or crypto profile.
- To configure the IPsec Anti-Replay Window: Expanding and Disabling feature, you should understand the following concept: IPsec Anti-Replay Window, on page 4

Information About IPsec Anti-Replay Window Expanding and Disabling

IPsec Anti-Replay Window

Cisco IPsec authentication provides anti-replay protection against an attacker duplicating encrypted packets by assigning a unique sequence number to each encrypted packet. (Security association [SA] anti-replay is a security service in which the receiver can reject old or duplicate packets to protect itself against replay attacks.) The decryptor checks off the sequence numbers that it has seen before. The encryptor assigns sequence numbers in an increasing order. The decryptor remembers the value X of the highest sequence number that it has already seen. N is the window size, and the decryptor also remembers whether it has seen packets having sequence numbers from X-N+1 through X. Any packet with the sequence number X-N is discarded. Currently, N is set at 64, so only 64 packets can be tracked by the decryptor.

At times, however, the 64-packet window size is not sufficient. For example, Cisco quality of service (QoS) gives priority to high-priority packets, which could cause some low-priority packets to be discarded when they arrive outside of the 64 packet replay window at the decryptor. The IPsec Anti-Replay Window: Expanding and Disabling feature allows you to expand the window size, allowing the decryptor to keep track of more than 64 packets.

Increasing the anti-replay window size has no impact on throughput and security. The impact on memory is insignificant because only an extra 128 bytes per incoming IPsec SA is needed to store the sequence number on the decryptor. It is recommended that you use the full 1024 window size to eliminate any future anti-replay problems.

How to Configure IPsec Anti-Replay Window Expanding and Disabling

Configuring IPsec Anti-Replay Window Expanding and Disabling Globally

To configure IPsec Anti-Replay Window: Expanding and Disabling globally (so that it affects all SAs that are created), perform the following steps.
SUMMARY STEPS

1. enable
2. configure terminal
3. crypto ipsec security-association replay window-size \[N\]
4. crypto ipsec security-association replay disable

DETAILED STEPS

<table>
<thead>
<tr>
<th>Command or Action</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Step 1</strong></td>
<td>enable</td>
</tr>
<tr>
<td>Example:</td>
<td>Router&gt; enable</td>
</tr>
<tr>
<td><strong>Step 2</strong></td>
<td>configure terminal</td>
</tr>
<tr>
<td>Example:</td>
<td>Router# configure terminal</td>
</tr>
<tr>
<td><strong>Step 3</strong></td>
<td>crypto ipsec security-association replay window-size [N]</td>
</tr>
<tr>
<td>Example:</td>
<td>Router (config)# crypto ipsec security-association replay window-size 256</td>
</tr>
<tr>
<td><strong>Step 4</strong></td>
<td>crypto ipsec security-association replay disable</td>
</tr>
<tr>
<td>Example:</td>
<td>Router (config)# crypto ipsec security-association replay disable</td>
</tr>
</tbody>
</table>

Configuring IPsec Anti-Replay Window Expanding and Disabling on a Crypto Map

To configure IPsec Anti-Replay Window: Expanding and Disabling on a crypto map so that it affects those SAs that have been created using a specific crypto map or profile, perform the following steps.
### SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **crypto map map-name seq-num [ipsec-isakmp]**
4. **set security-association replay window-size [N]**
5. **set security-association replay disable**

### 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><em>Example:</em> Router&gt; enable</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><em>Example:</em> Router# configure terminal</td>
<td></td>
</tr>
<tr>
<td><strong>Step 3</strong> crypto map map-name seq-num [ipsec-isakmp]</td>
<td>Enters crypto map configuration mode and creates a crypto profile that provides a template for configuration of dynamically created crypto maps.</td>
</tr>
<tr>
<td><em>Example:</em> Router (config)# crypto map ETHO 17 ipsec-isakmp</td>
<td></td>
</tr>
<tr>
<td><strong>Step 4</strong> set security-association replay window-size [N]</td>
<td>Controls the SAs that are created using the policy specified by a particular crypto map, dynamic crypto map, or crypto profile.</td>
</tr>
<tr>
<td><em>Example:</em> Router (crypto-map)# set security-association replay window-size 128</td>
<td></td>
</tr>
<tr>
<td><strong>Step 5</strong> set security-association replay disable</td>
<td>Disables replay checking for a particular crypto map, dynamic crypto map, or crypto profile.</td>
</tr>
<tr>
<td><em>Example:</em> Router (crypto-map)# set security-association replay disable</td>
<td>Configure this command or the <code>set security-association replay disable</code> command. The two commands are not used at the same time.</td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Troubleshooting Tips

• If your replay window size has not been set to a number that is high enough for the number of packets received, you will receive a system message such as the following:

*Nov 17 19:27:32.279: %CRYPTO-4-PKT_REPLAY_ERR: decrypt: replay check failed connection id=1
The above message is generated when a received packet is judged to be outside the anti-replay window.

Configuration Examples for IPsec Anti-Replay Window Expanding and Disabling

Global Expanding and Disabling of an Anti-Replay Window Example

The following example shows that the anti-replay window size has been set globally to 1024:

```
version 2.1
service timestamps debug datetime msec
service timestamps log datetime msec
no service password-encryption
!
hostname VPN-Gateway1
!
boot-start-marker
boot-end-marker
!
!
clock timezone EST 0
no aaa new-model
ip subnet-zero
!
!
ip audit po max-events 100
no ftp-server write-enable
!
!
crypto isakmp policy 10
  authentication pre-share
  crypto isakmp key cisco123 address 192.165.201.2 !
crypto ipsec security-association replay window-size 1024 !
crypto ipsec transform-set basic esp-des esp-md5-hmac !
crypto map mymap 10 ipsec-isakmp
  set peer 192.165.201.2
  set transform-set basic
  match address 101
!
!
interface FastEthernet0/0
  ip address 192.168.1.1 255.255.255.0
!
interface Serial1/0
  ip address 192.165.200.2 255.255.255.252 serial restart-delay 0 crypto map mymap !
  ip classless
  ip route 0.0.0.0 0.0.0.0 192.165.200.1
  no ip http server
  no ip http secure-server
!
!
access-list 101 permit ip 192.168.1.0 0.0.0.255 172.16.2.0 0.0.0.255 access-list 101 remark
```
Expanding and Disabling of an Anti-Replay Window for Crypto Maps or Crypto Profiles Example

The following example shows that anti-replay checking is disabled for IPsec connections to 172.17.150.2 but enabled (and the default window size is 64) for IPsec connections to 172.17.150.3 and 172.17.150.4:

service timestamps debug uptime
service timestamps log uptime
no service password-encryption
!
hostname networkserver1
!
enable secret 5 $1$KxKv$cbqRsZtQTLJLGPN.tErFZ1 enable password ww
!
ip subnet-zero
!
cns event-service server
crypto isakmp policy 1
authentication pre-share
crypto isakmp key cisco170 address 172.17.150.2 crypto isakmp key cisco180 address 172.17.150.3 crypto isakmp key cisco190 address 172.17.150.4
crypto ipsec transform-set 170cisco esp-des esp-md5-hmac crypto ipsec transform-set 180cisco esp-des esp-md5-hmac crypto ipsec transform-set 190cisco esp-des esp-md5-hmac
crypto map ETH0 17 ipsec-isakmp
set peer 172.17.150.2
set security-association replay disable set transform-set 170cisco match address 170 crypto map ETH0 18 ipsec-isakmp set peer 192.168.1.3 set transform-set 180cisco match address 180 crypto map ETH0 19 ipsec-isakmp set peer 192.168.1.4 set transform-set 190cisco match address 190 !
interface FastEthernet0
ip address 172.17.150.1 255.255.255.0
no ip directed-broadcast
no ip route-cache
no ip mroute-cache
no mop enabled
crypto map ETH0
!
interface Serial0
ip address 172.16.160.1 255.255.255.0
no ip directed-broadcast
no ip mroute-cache
no fair-queue
!
ip classless
ip route 172.18.170.0 255.255.255.0 172.17.150.2 ip route 172.19.180.0 255.255.255.0 172.17.150.4 no ip http server
access-list 170 permit ip 172.16.160.0 0.0.0.255 172.18.170.0 0.0.0.255 access-list 180 permit ip 172.16.160.0 0.0.0.255 172.19.180.0 0.0.0.255 access-list 190 permit ip 172.16.160.0 0.0.0.255 172.20.190.0 0.0.0.255 !
dialer-list 1 protocol ip permit
dialer-list 1 protocol ipx permit
!
line con 0
transport input none
line aux 0
### Additional References

The following sections provide references related to IPsec Anti-Replay Window: Expanding and Disabling.

#### 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 Security Command Reference</td>
</tr>
<tr>
<td>IP security and encryption</td>
<td>Configuring Security for VPNs with IPsec</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>No new or modified RFCs are supported by this feature, and support for existing RFCs has not been modified by this feature.</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/techsupport">http://www.cisco.com/techsupport</a></td>
</tr>
</tbody>
</table>

Feature Information for IPsec Anti-Replay Window Expanding and Disabling

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 IPsec Anti-Replay Window: Expanding and Disabling**

<table>
<thead>
<tr>
<th>Feature Name</th>
<th>Releases</th>
<th>Feature Information</th>
</tr>
</thead>
<tbody>
<tr>
<td>IPsec Anti-Replay Window: Expanding and Disabling</td>
<td>Cisco IOS XE Release 2.1</td>
<td>The following commands were introduced or modified: <code>crypto ipsec security-association replay disable</code>, <code>ipsec security-association replay window-size</code>, <code>security-association replay disable</code>, <code>security-association replay window-size</code>.</td>
</tr>
</tbody>
</table>
Pre-Fragmentation for IPsec VPNs

The Pre-Fragmentation for IPsec VPNs feature increases performance between Cisco IOS XE routers and VPN clients by delivering encryption throughput at maximum encryption hardware accelerator speeds for packets that are near the maximum transmission unit (MTU) size. Packets are fragmented into equally sized units to prevent further downstream fragmentation.

• Finding Feature Information, page 11
• Restrictions for Pre-Fragmentation for IPsec VPNs, page 11
• Information About Pre-Fragmentation for IPsec VPNs, page 13
• How to Configure Pre-Fragmentation for IPsec VPNs, page 13
• Additional References, page 14
• Feature Information for Pre-Fragmentation for IPsec VPNs, page 15

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest caveats and feature information, see Bug Search Tool and 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 module.

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 Pre-Fragmentation for IPsec VPNs

Take the following information into consideration before this feature is configured:

• Pre-fragmentation for IPsec VPNs operates in IPsec tunnel mode and IPsec tunnel mode with GRE, but not with IPsec transport mode.
• Pre-fragmentation for IPsec VPNs configured on the decrypting router in a unidirectional traffic scenario does not improve the performance or change the behavior of either of the peers.
• Pre-fragmentation for IPsec VPNs occurs before the transform is applied if compression is turned on for outgoing packets.

• Pre-fragmentation for IPsec VPNs functionality depends on the egress interface `crypto ipsec df-bit` configuration and the incoming packet "do not fragment" (DF) bit state. See the table below.

**Table 2: Pre-Fragmentation for IPsec VPNs Dependencies**

<table>
<thead>
<tr>
<th>Pre-Fragmentation for IPsec VPNs Feature State (Enabled/Disabled)</th>
<th>Egress Interface “crypto ipsec df-bit” Configuration</th>
<th>Incoming Packet DF Bit State</th>
<th>Result</th>
</tr>
</thead>
<tbody>
<tr>
<td>Enabled</td>
<td>crypto ipsec df-bit clear</td>
<td>0</td>
<td>Fragmentation occurs before encryption.</td>
</tr>
<tr>
<td>Enabled</td>
<td>crypto ipsec df-bit clear</td>
<td>1</td>
<td>Fragmentation occurs before encryption.</td>
</tr>
<tr>
<td>Disabled</td>
<td>crypto ipsec df-bit clear</td>
<td>0</td>
<td>Fragmentation occurs after encryption and packets are reassembled before decryption.</td>
</tr>
<tr>
<td>Disabled</td>
<td>crypto ipsec df-bit clear</td>
<td>1</td>
<td>Fragmentation occurs after encryption and packets are reassembled before decryption.</td>
</tr>
<tr>
<td>Enabled</td>
<td>crypto ipsec df-bit set</td>
<td>0</td>
<td>Fragmentation occurs before encryption.</td>
</tr>
<tr>
<td>Enabled</td>
<td>crypto ipsec df-bit set</td>
<td>1</td>
<td>Packets are dropped.</td>
</tr>
<tr>
<td>Disabled</td>
<td>crypto ipsec df-bit set</td>
<td>0</td>
<td>Fragmentation occurs after encryption and packets are reassembled before decryption.</td>
</tr>
<tr>
<td>Disabled</td>
<td>crypto ipsec df-bit set</td>
<td>1</td>
<td>Packets are dropped.</td>
</tr>
<tr>
<td>Enabled</td>
<td>crypto ipsec df-bit copy</td>
<td>0</td>
<td>Fragmentation occurs before encryption.</td>
</tr>
<tr>
<td>Enabled</td>
<td>crypto ipsec df-bit copy</td>
<td>1</td>
<td>Packets are dropped.</td>
</tr>
<tr>
<td>Disabled</td>
<td>crypto ipsec df-bit copy</td>
<td>0</td>
<td>Fragmentation occurs after encryption, and packets are reassembled before decryption.</td>
</tr>
<tr>
<td>Disabled</td>
<td>crypto ipsec df-bit copy</td>
<td>1</td>
<td>Packets are dropped.</td>
</tr>
</tbody>
</table>
Information About Pre-Fragmentation for IPsec VPNs

Pre-fragmentation for IPsec VPNs

When a packet is nearly the size of the MTU of the outbound link of the encrypting router and it is encapsulated with IPsec headers, it is likely to exceed the MTU of the outbound link. This causes packet fragmentation after encryption. The decrypting router must then reassemble these packets in the process path, which decreases the decrypting router’s performance.

The Pre-fragmentation for IPsec VPNs feature increases the decrypting router’s performance by enabling it to operate in the high-performance CEF path instead of the process path. An encrypting router can predetermine the encapsulated packet size from information available in transform sets, which are configured as part of the IPsec security association (SA). If it is predetermined that the packet exceeds the MTU of the output interface, the packet is fragmented before encryption. This function avoids process-level reassembly before decryption and helps improve decryption performance and overall IPsec traffic throughput.

Note

The pre-fragmentation feature is turned off by default for tunnel interfaces. To receive pre-fragmentation performance benefits, turn pre-fragmentation on after ensuring that the tunnel interfaces have the same MTU on both ends.

Note

Crypto maps are no longer used to define fragmentation behavior that occurred before and after encryption. Now, IPsec Virtual Tunnel Interface (also referred to as Virtual-Template interface) (VTI) fragmentation behavior is determined by the IP MTU settings that are configured on the VTI.

See the IPsec Virtual Tunnel Interface feature document for more information on VTIs.

How to Configure Pre-Fragmentation for IPsec VPNs

Configuring Pre-Fragmentation for IPsec VPNs

Perform this task to configure Pre-Fragmentation for IPsec VPNs.
SUMMARY STEPS

1. enable
2. configure terminal
3. interface type number
4. ip mtu bytes

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></td>
</tr>
<tr>
<td>Router# configure terminal</td>
<td></td>
</tr>
<tr>
<td>Step 3 interface type number</td>
<td>Specifies the interface on which the VTI is configured and enters</td>
</tr>
<tr>
<td>Example:</td>
<td>interface configuration mode.</td>
</tr>
<tr>
<td>Router(config-if)# interface tunnel0</td>
<td></td>
</tr>
<tr>
<td>Step 4 ip mtu bytes</td>
<td>Specifies the VTI MTU size in bytes of IP packets on the egress</td>
</tr>
<tr>
<td>Example:</td>
<td>interface for IPsec VPNS.</td>
</tr>
<tr>
<td>Router(config-if)# ip mtu 1500</td>
<td>Note: If after-encryption fragmentation behavior is desired, then set</td>
</tr>
<tr>
<td></td>
<td>the VTI IP MTU to a value that is greater than the egress router</td>
</tr>
<tr>
<td></td>
<td>interface IP MTU. Use the show ip interface tunnel command to display</td>
</tr>
<tr>
<td></td>
<td>the IP MTU value.</td>
</tr>
</tbody>
</table>

Additional References

**Related Documents**

<table>
<thead>
<tr>
<th>Related Topic</th>
<th>Document Title</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cisco IOS XE commands</td>
<td>Cisco IOS Master Commands List, All Releases</td>
</tr>
</tbody>
</table>
Related Topic | Document Title
--- | ---
Security commands | Cisco IOS Security Command Reference
IPsec | IPsec Virtual Tunnel Interface feature document

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 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

<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 Pre-Fragmentation 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. An account on Cisco.com is not required.
### Table 3: Feature Information for Pre-Fragmentation for IPsec VPNs

<table>
<thead>
<tr>
<th>Feature Name</th>
<th>Releases</th>
<th>Feature Information</th>
</tr>
</thead>
</table>
| Pre-Fragmentation for IPsec VPNs | Cisco IOS XE 2.1 | This feature increases performance between Cisco IOS routers and VPN clients by delivering encryption throughput at maximum encryption hardware accelerator speeds for packets that are near the maximum transmission unit (MTU) size. Packets are fragmented into equally sized units to prevent further downstream fragmentation. The following command was introduced or modified: `ip mtu (interface configuration)`.
Invalid Security Parameter Index Recovery

When an invalid security parameter index error (shown as "Invalid SPI") occurs in IP Security (IPsec) packet processing, the Invalid Security Parameter Index Recovery feature allows for an Internet Key Exchange (IKE) security association (SA) to be established. The "IKE" module sends notification of the "Invalid SPI" error to the originating IPsec peer so that Security Association Databases (SADBs) can be resynchronized and successful packet processing can be resumed.

- Finding Feature Information, page 17
- Prerequisites for Invalid Security Parameter Index Recovery, page 17
- Restrictions for Invalid Security Parameter Index Recovery, page 18
- Information About Invalid Security Parameter Index Recovery, page 18
- How to Configure Invalid Security Parameter Index Recovery, page 18
- Configuration Examples for Invalid Security Parameter Index Recovery, page 25
- Additional References, page 29
- Feature Information for Invalid Security Parameter Index Recovery, page 31

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest caveats and feature information, see Bug Search Tool and 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 module.

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 Invalid Security Parameter Index Recovery

Before configuring the Invalid Security Parameter Index Recovery feature, you must have enabled IKE and IPsec on your router.
Restrictions for Invalid Security Parameter Index Recovery

If an IKE SA is being initiated to notify an IPsec peer of an "Invalid SPI" error, there is the risk that a denial-of-service (DoS) attack can occur. The Invalid Security Parameter Index Recovery feature has a built-in mechanism to minimize such a risk, but because there is a risk, the Invalid Security Parameter Index Recovery feature is not enabled by default. You must enable the command using command-line interface (CLI).

Information About Invalid Security Parameter Index Recovery

How the Feature Works

An IPsec "black hole" occurs when one IPsec peer "dies" (for example, a peer can "die" if a reboot occurs or if an IPsec peer somehow gets reset). Because one of the peers (the receiving peer) is completely reset, it loses its IKE SA with the other peer. Generally, when an IPsec peer receives a packet for which it cannot find an SA, it tries to send an IKE "INVALID SPI NOTIFY" message to the data originator. This notification is sent using the IKE SA. If there is no IKE SA available, the receiving peer drops the packet.

A single SA has only two peers. However, a SADB can have multiple SAs, whereby each SA has an association with a different peer.

When an invalid security parameter index (SPI) is encountered, the Invalid Security Parameter Index feature provides for the setting up of an IKE SA with the originator of the data, and the IKE "INVALID SPI NOTIFY" message is sent. The peer that originated the data "sees" the "INVALID SPI NOTIFY" message and deletes the IPsec SA that has the invalid SPI. If there is further traffic from the originating peer, there will not be any IPsec SAs, and new SAs will be set up. Traffic will flow again. The default behavior (that is, without configuring the Invalid Security Parameter Index Recovery feature) is that the data packet that caused the invalid SPI error is dropped. The originating peer keeps on sending the data using the IPsec SA that has the invalid SPI, and the receiving peer keeps dropping the traffic (thus creating the "black hole").

The IPsec module uses the IKE module to send an IKE "INVALID SPI NOTIFY" message to the other peer. Once the invalid SPI recovery is in place, there should not be any significant dropping of packets although the IPsec SA setup can itself result in the dropping of a few packets.

To configure your router for the Invalid Security Parameter Index Recovery feature, use the `crypto isakmp invalid-spi-recovery` command. The IKE SA will not be initiated unless you have configured this command.

How to Configure Invalid Security Parameter Index Recovery

Configuring Invalid Security Parameter Index Recovery

To configure the Invalid Security Parameter Index Recovery feature, perform the following steps.
SUMMARY STEPS

1. enable
2. configure terminal
3. crypto isakmp invalid-spi-recovery

DETAILED STEPS

<table>
<thead>
<tr>
<th>Command or Action</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Step 1</strong></td>
<td></td>
</tr>
<tr>
<td>enable</td>
<td>Enables privileged EXEC mode.</td>
</tr>
<tr>
<td>• Enter your password if prompted.</td>
<td></td>
</tr>
<tr>
<td><strong>Example:</strong></td>
<td></td>
</tr>
<tr>
<td>Router&gt; enable</td>
<td></td>
</tr>
<tr>
<td><strong>Step 2</strong></td>
<td>Enters global configuration mode.</td>
</tr>
<tr>
<td>configure terminal</td>
<td></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></td>
<td>Initiates the IKE module process whereby the IKE module notifies the</td>
</tr>
<tr>
<td>crypto isakmp invalid-spi-recovery</td>
<td>receiving peer that an &quot;Invalid SPI&quot; error has occurred.</td>
</tr>
<tr>
<td><strong>Example:</strong></td>
<td></td>
</tr>
<tr>
<td>Router (config)# crypto isakmp invalid-spi-recovery</td>
<td></td>
</tr>
</tbody>
</table>

Verifying a Preshared Configuration

To determine the status of the IPsec SA for traffic between two peers, you can use the `show crypto ipsec sa` command. If the IPsec SA is available on one peer and not on the other, there is a “black hole” situation, in which case you will see the invalid SPI errors being logged for the receiving peer. If you turn console logging on or check the syslog server, you will see that these errors are also being logged.
The diagram below shows the topology of a typical preshared configuration setup. Host 1 is the initiating peer (initiator), and Host 2 is the receiving peer (responder).

**Figure 1: Preshared Configuration Topology**

**SUMMARY STEPS**

1. Initiate the IKE and IPsec SAs between Host 1 and Host 2
2. Clear the IKE and IPsec SAs on Router B
3. Send traffic from Host 1 to Host 2 and ensure that new IKE and IPsec SAs are correctly established
4. Check for an invalid SPI message on Router B

**DETAILED STEPS**

**Step 1**
Initiate the IKE and IPsec SAs between Host 1 and Host 2

**Router A**

**Example:**

```
Router# show crypto isakmp sa
f_vrf/i_vrf dst src state conn-id slot
/ 10.2.2.2 10.1.1.1 QM_IDLE 1 0
```

**Router B**

**Example:**

```
Router# show crypto isakmp sa
f_vrf/i_vrf dst src state conn-id slot
/ 10.1.1.1 10.2.2.2 QM_IDLE 1 0
```

**Router A**

**Example:**

```
Router# show crypto ipsec sa interface fastethernet0/0
interface: FastEthernet0/0
    Crypto map tag: testtag1, local addr. 10.1.1.1
    protected vrf: local ident (addr/mask/prot/port): (10.0.0.1/255.255.255.255/0/0)
    remote ident (addr/mask/prot/port): (10.0.2.2/255.255.255.255/0/0)
    current-peer: 10.2.2.2:500
    PERMIT, flags={origin_is_acl,}
    #pkts encaps: 10, #pkts encrypt: 10, #pkts digest: 10
    #pkts decaps: 10, #pkts decrypt: 10, #pkts verify: 10
    #pkts compressed: 0, #pkts decompressed: 0
    #pkts not compressed: 0, #pkts compr. failed: 0
    #pkts not decompressed: 0, #pkts decompress failed: 0
    local crypto endpt.: 10.1.1.1, remote crypto endpt.: 10.2.2.2
    path mtu 1500, media mtu 1500
    current outbound spi: 7AA69CB7
    inbound esp sas:
        spi: 0x249c5062(614223970)
        transform: esp-des esp-sha-hmac ,
        in use settings ={Tunnel, } 
        slot: 0, conn id: 5123, flow_id: 1, crypto map: testtag1
```
crypto engine type: Hardware
sa timing: remaining key lifetime (k/sec): (4537831/3595)
IV size: 8 bytes
replay detection support: Y
inbound ah sas:
  spi: 0xB16D1587(2976716167)
  transform: ah-sha-hmac ,
in use settings ={Tunnel, }
slot: 0, conn id: 5121, flow_id: 1, crypto map: testtag1
  crypto engine type: Hardware
  sa timing: remaining key lifetime (k/sec): (4537831/3595)
  replay detection support: Y
inbound pcp sas:
  spi: 0x7AA69CB7(2057739447)
  transform: esp-des esp-sha-hmac ,
in use settings ={Tunnel, }
slot: 0, conn id: 5124, flow_id: 2, crypto map: testtag1
  crypto engine type: Hardware
  sa timing: remaining key lifetime (k/sec): (4537835/3594)
  IV size: 8 bytes
  replay detection support: Y

outbound ah sas:
  spi: 0x1214F0D(18960141)
  transform: ah-sha-hmac ,
in use settings ={Tunnel, }
slot: 0, conn id: 5122, flow_id: 2, crypto map: testtag1
  crypto engine type: Hardware
  sa timing: remaining key lifetime (k/sec): (4537835/3594)
  replay detection support: Y

outbound pcp sas:
  spi: 0x1214F0D(18960141)
  transform: esp-des esp-sha-hmac ,
in use settings ={Tunnel, }
slot: 0, conn id: 5123, flow_id: 1, crypto map: testtag1
  crypto engine type: Hardware
  sa timing: remaining key lifetime (k/sec): (4421281/3593)
  IV size: 8 bytes
  replay detection support: Y

Router B

Example:

Router# show crypto ipsec sa interface Fastethernet1/0
interface: FastEthernet1/0
  Crypto map tag: testtag1, local addr. 10.2.2.2
  protected vrf:
    local ident (addr/mask/prot/port): (10.0.2.2/255.255.255.255/0/0)
    remote ident (addr/mask/prot/port): (10.0.0.1/255.255.255.255/0/0)
  current_peer: 10.1.1.1:500
    PERMIT, flags={origin_is_acl,}
    #pkts encaps: 10, #pkts encryp: 10, #pkts digest: 10
    #pkts decaps: 10, #pkts decrypt: 10, #pkts verify: 10
    #pkts compressed: 0, #pkts decompressed: 0
    #pkts not compressed: 0, #pkts compr. failed: 0
    #pkts not decompressed: 0, #pkts decompress failed: 0
    send errors 0, recv errors 0
    local crypto endpt.: 10.2.2.2, remote crypto endpt.: 10.1.1.1
    path mtu 1500, media mtu 1500
    current outbound spi: 249C5062
    inbound esp sas:
      spi: 0x7AA69CB7(2057739447)
      transform: esp-des esp-sha-hmac ,
in use settings ={Tunnel, }
slot: 0, conn id: 5123, flow_id: 1, crypto map: testtag1
      crypto engine type: Hardware
      sa timing: remaining key lifetime (k/sec): (4421281/3593)
      IV size: 8 bytes
      replay detection support: Y
    inbound ah sas:
      spi: 0x1214F0D(18960141)
      transform: ah-sha-hmac ,
in use settings ={Tunnel, }
slot: 0, conn id: 5121, flow_id: 1, crypto map: testtag1
      crypto engine type: Hardware
      sa timing: remaining key lifetime (k/sec): (4421281/3593)
replay detection support: Y
inbound pcp sas:
  spi: 0x249C5062(61423970)
  transform: esp-des esp-sha-hmac ,
  in use settings ={Tunnel, }
  slot: 0, conn id: 5124, flow_id: 2, crypto map: testtag1
  crypto engine type: Hardware
  sa timing: remaining key lifetime (k/sec): (4421285/3593)
  IV size: 8 bytes
  replay detection support: Y
outbound esp sas:
  spi: 0x249C5062(61423970)
  transform: esp-des esp-sha-hmac ,
  in use settings ={Tunnel, }
  slot: 0, conn id: 5124, flow_id: 2, crypto map: testtag1
  crypto engine type: Hardware
  sa timing: remaining key lifetime (k/sec): (4421285/3593)
  IV size: 8 bytes
  replay detection support: Y
outbound ah sas:
  spi: 0xB16D1587(2976716167)
  transform: ah-sha-hmac ,
  in use settings ={Tunnel, }
  slot: 0, conn id: 5122, flow_id: 2, crypto map: testtag1
  crypto engine type: Hardware
  sa timing: remaining key lifetime (k/sec): (4421285/3592)
  replay detection support: Y
outbound pcp sas:

**Step 2** Clear the IKE and IPsec SAs on Router B

**Example:**

```
Router# clear crypto isakmp
Router# clear crypto sa
Router# show crypto isakmp sa
   f_vrf/i_vrf dst src state conn-id slot
   / 10.2.2.2. 10.1.1.1 MM_NO_STATE 1 0 (deleted)
Router# show crypto ipsec sa
   interface: FastEthernet1/0
   Crypto map tag: testtag1, local addr. 10.2.2.2
   protected vrf:
   local ident (addr/mask/prot/port): (10.0.2.2/255.255.255.255/0/0)
   remote ident (addr/mask/prot/port): (10.0.0.1/255.255.255.255/0/0)
   current_peer: 10.1.1.1:5000
   PERMIT, flags={origin_is_acl,}
   #pkts encaps: 0, #pkts encrypt: 0, #pkts digest: 0
   #pkts decaps: 0, #pkts decrypt: 0, #pkts verify: 0
   #pkts compressed: 0, #pkts decompressed: 0
   #pkts not compressed: 0, #pkts compr. failed: 0
   #pkts not decompressed: 0, #pkts decompress failed: 0
   local crypto endpt.: 10.2.2.2, remote crypto endpt.: 10.1.1.1
   path mtu 1500, media mtu 1500
   current outbound spi: 0
   inbound esp sas: 0
   inbound ah sas: 0
   inbound pcp sas: 0
   outbound esp sas: 0
   outbound ah sas: 0
   outbound pcp sas: 0
```

**Step 3** Send traffic from Host 1 to Host 2 and ensure that new IKE and IPsec SAs are correctly established

**Example:**

```
ping
Protocol [ip]: ip
Target IP address: 10.0.2.2
Repeat count [5]: 30
Datagram size [100]: 100
Timeout in seconds [2]: 2
Extended commands [n]: 0
Sweep range of sizes [n]: 0
Type escape sequence to abort.
Sending 30, 100-byte ICMP Echos to 10.0.2.2, timeout is 2 seconds:
```
Success rate is 93 percent (28/30), round-trip min/avg/max = 1/3/8 ms

RouterB# show crypto isakmp sa
<table>
<thead>
<tr>
<th>f_vrf/i_vrf</th>
<th>dst</th>
<th>src</th>
<th>state</th>
<th>conn-id</th>
<th>slot</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>10.1.1.1</td>
<td>10.2.2.2</td>
<td>QM_IDLE</td>
<td>3</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td>10.1.1.1</td>
<td>10.2.2.2</td>
<td>MM_NO_STATE</td>
<td>1</td>
<td>0 (deleted)</td>
</tr>
</tbody>
</table>

RouterB# show crypto ipsec sa
interface: FastEthernet1/0
Crypto map tag: testtag1, local addr. 10.2.2.2
protected vrf:
  local ident (addr/mask/prot/port): (10.0.2.2/255.255.255.255/0/0)
  remote ident (addr/mask/prot/port): (10.0.0.1/255.255.255.255/0/0)
current_peer: 10.1.1.1:500
  PERMIT, flags=(origin_is_acl,)
  pkts encaps: 28, pkts encrypt: 28, pkts digest: 28
  pkts decaps: 28, pkts decrypt: 28, pkts verify: 28
  pkts compressed: 0, pkts decompressed: 0
  pkts not compressed: 0, pkts compr. failed: 0
  pkts not decompressed failed: 0
  send errors 0, recv errors 0
  local crypto endpt.: 10.2.2.2, remote crypto endpt.: 10.1.1.1
  path mtu 1500, media mtu 1500
current outbound spi: D763771F
inbound esp sas:
  spi: 0x7E7A4256(3886760534)
    transform: esp-des esp-sha-hmac ,
    in use settings ={Tunnel,}
    slot: 0, conn id: 5127, flow_id: 3, crypto map: testtag1
    crypto engine type: Hardware
    sa timing: remaining key lifetime (k/sec): (4502463/3596)
    IV size: 8 bytes
    replay detection support: Y

inbound ah sas:
  spi: 0xF9205CED(417964725)
    transform: ah-sha-hmac ,
    in use settings ={Tunnel,}
    slot: 0, conn id: 5125, flow_id: 3, crypto map: testtag1
    crypto engine type: Hardware
    sa timing: remaining key lifetime (k/sec): (4502463/3596)
    replay detection support: Y

inbound pcp sas:

outbound esp sas:
  spi: 0xD763771F(3613619999)
    transform: esp-des esp-sha-hmac ,
    in use settings ={Tunnel,}
    slot: 0, conn id: 5128, flow_id: 4, crypto map: testtag1
    crypto engine type: Hardware
    sa timing: remaining key lifetime (k/sec): (4502468/3596)
    IV size: 8 bytes
    replay detection support: Y

outbound ah sas:
  spi: 0xEB95406F(3952427119)
    transform: ah-sha-hmac ,
    in use settings ={Tunnel,}
    slot: 0, conn id: 5126, flow_id: 4, crypto map: testtag1
    crypto engine type: Hardware
    sa timing: remaining key lifetime (k/sec): (4502468/3595)
    replay detection support: Y

outbound pcp sas:

RouterA# show crypto isakmp sa
<table>
<thead>
<tr>
<th>f_vrf/i_vrf</th>
<th>dst</th>
<th>src</th>
<th>state</th>
<th>conn-id</th>
<th>slot</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>10.2.2.2</td>
<td>10.1.1.1</td>
<td>MM_NO_STATE</td>
<td>1</td>
<td>0 (deleted)</td>
</tr>
<tr>
<td></td>
<td>10.2.2.2</td>
<td>10.1.1.1</td>
<td>MM_IDLE</td>
<td>2</td>
<td>0</td>
</tr>
</tbody>
</table>

---

Step 4
Check for an invalid SPI message on Router B

Example:

Router# show logging
Syslog logging: enabled (10 messages dropped, 13 messages rate-limited, 0 flushes, 0 overruns, xml
Invalid Security Parameter Index Recovery Example

The following example shows that invalid security parameter index recovery has been configured on Router A and Router B. Invalid Security Parameter Index Recovery Example, on page 25 shows the topology used for this example.

Router A

Router# show running-config
Building configuration...
Current configuration : 2048 bytes
!
version 2.1
no service pad
service timestamps debug datetime msec localtime
service timestamps log datetime msec localtime
no service password-encryption
service tcp-small-servers
!
hostname ipseca-71a
!
logging queue-limit 100
no logging console
enable secret 5 $1$4GZBS12YOmnenOCNAu0jgFxe8T/
enable password lab
!
clock timezone PST -8
clock summer-time PDT recurring
ip subnet-zero
!
no ip domain lookup
!
ip cef
ip audit notify log
ip audit po max-events 100
mpls ldp logging neighbor-changes
no ftp-server write-enable
!
no voice hpi capture buffer
no voice hpi capture destination
!
crypto isakmp policy 1
 authentication pre-share
 lifetime 180
 crypto isakmp key 0 1234 address 10.2.2.2
 crypto isakmp invalid-spi-recovery

crypto ipsec transform-set auth2 ah-sha-hmac esp-des esp-sha-hmac

crypto map testtag1 10 ipsec-isakmp
 set peer 10.2.2.2
 set transform-set auth2
 match address 150

controller ISA 5/1

interface FastEthernet0/0
 ip address 10.1.1.1 255.0.0.0
 no ip route-cache cef
duplex full
 speed 100
 crypto map testtag1

interface FastEthernet0/1
 ip address 10.0.0.1 255.0.0.0
 no ip route-cache cef
duplex auto
 speed auto

interface Serial1/0
 no ip address
 no ip route-cache
 no ip mroute-cache
 shutdown
 serial restart_delay 0
clockrate 128000

interface Serial1/1
 no ip address
 no ip route-cache
 no ip mroute-cache
 shutdown
 serial restart_delay 0
clockrate 128000

interface Serial1/2
 no ip address
 no ip route-cache
 no ip mroute-cache
 shutdown
 serial restart_delay 0

interface Serial1/3
 no ip address
 no ip route-cache
 no ip mroute-cache
 shutdown
 no keepalive
 serial restart_delay 0
clockrate 128000

ip classless
 ip route 10.3.3.3 255.0.0.0 10.2.0.1
 no ip http server
 no ip http secure-server

access-list 150 permit ip host 10.0.0.1 host 10.0.2.2
dialer-list 1 protocol ip permit
dialer-list 1 protocol ipx permit
call rsvp-sync
mgcp profile default
!
line con 0
  exec-timeout 0 0
line aux 0
line vty 0 4
password lab
login
!
end
ipseca-71a#

Router B

Router# show running-config
Building configuration...
Current configuration : 2849 bytes
!
version 2.1
no service pad
service timestamps debug datetime msec localtime
service timestamps log datetime msec localtime
no service password-encryption
service udp-small-servers
service tcp-small-servers
!
hostname ipseca-72a
!
logging queue-limit 100
no logging console
enable secret 5 $1$kKqL5Th5QhwlubDkkK90KWFxi1
enable password lab
!
clock timezone PST -8
clock summer-time PDT recurring
ip subnet-zero
!
no ip domain lookup
!
ip cef
ip audit notify log
ip audit po max-events 100
mpls ldp logging neighbor-changes
no ftp-server write-enable
!
no voice hpi capture buffer
no voice hpi capture destination
!
mta receive maximum-recipients 0
!
crypto isakmp policy 1
  authentication pre-share
  lifetime 180
crypto isakmp key 0 1234 address 10.1.1.1
crypto isakmp invalid-spi-recovery
!
!crypto ipsec transform-set auth2 ah-sha-hmac esp-des esp-sha-hmac
!crypto map testtag1 10 ipsec-isakmp
set peer 10.1.1.1
set transform-set auth2
match address 150
!
controller ISA 5/1
!
interface FastEthernet0/0
no ip address
no ip route-cache
no ip mrout-cache
shutdown
duplex half
!
interface FastEthernet1/0
ip address 10.2.2.2 255.0.0.0
no ip route-cache cef
duplex half
crypto map testtag1
!
interface FastEthernet1/1
ip address 10.0.2.2 255.0.0.0
no ip route-cache cef
duplex half
!
interface FastEthernet1/2
no ip address
no ip route-cache
no ip mrout-cache
shutdown
duplex half
!
interface FastEthernet1/3
no ip address
no ip route-cache
no ip mrout-cache
shutdown
duplex half
!
interface FastEthernet1/4
no ip address
no ip route-cache
no ip mrout-cache
shutdown
duplex half
!
interface FastEthernet1/5
no ip address
no ip route-cache
no ip mrout-cache
shutdown
duplex half
!
interface FastEthernet1/6
no ip address
no ip route-cache
no ip mrout-cache
shutdown
duplex half
!
interface FastEthernet1/7
no ip address
no ip route-cache
no ip mrout-cache
shutdown
duplex half
!
interface Serial3/0
no ip address
no ip route-cache
no ip mrout-cache
shutdown
serial restart_delay 0
!
interface Serial3/1
no ip address
no ip route-cache
no ip mroute-cache
shutdown
serial restart_delay 0
clockrate 128000
!
interface Serial3/2
no ip address
no ip route-cache
no ip mroute-cache
shutdown
serial restart_delay 0
!
interface Serial3/3
no ip address
no ip route-cache
no ip mroute-cache
shutdown
no keepalive
serial restart_delay 0
clockrate 128000
!
ip classless
ip route 10.0.0.0 255.0.0.0 10.2.0.1	no ip http server
no ip http secure-server
!
access-list 150 permit ip host 10.0.2.2 host 10.0.0.1
dialer-list 1 protocol ip permit
dialer-list 1 protocol ipx permit
!
call rsvp-sync
!
mgcp profile default
!
dial-peer cor custom
!
gatekeeper
  shutdown
!
line con 0
  exec-timeout 0 0
  stopbits 1
line aux 0
  stopbits 1
line vty 0 4
  password lab
  login
!
end

**Additional References**

The following sections provide references relate to Invalid Security Parameter Index Recovery.
Related Documents

<table>
<thead>
<tr>
<th>Related Topic</th>
<th>Document Title</th>
</tr>
</thead>
<tbody>
<tr>
<td>Configuring IKE</td>
<td>Configuring Internet Key Exchange for IPsec VPNs</td>
</tr>
<tr>
<td>Interface commands</td>
<td>Cisco IOS Master Command List</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</td>
<td>--</td>
</tr>
<tr>
<td>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</td>
<td>To locate and download MIBs for selected platforms, Cisco IOS XE software</td>
</tr>
<tr>
<td>existing MIBs has not been modified by this feature.</td>
<td>releases, and feature sets, use Cisco MIB Locator found at the following</td>
</tr>
<tr>
<td></td>
<td>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>No new or modified RFCs are supported by this feature, and support</td>
<td>--</td>
</tr>
<tr>
<td>for existing RFCs has not been modified by this feature.</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/techsupport">http://www.cisco.com/techsupport</a></td>
</tr>
</tbody>
</table>

Feature Information for Invalid Security Parameter Index Recovery

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 4: Feature Information for Invalid Security Parameter Index Recovery

<table>
<thead>
<tr>
<th>Feature Name</th>
<th>Releases</th>
<th>Feature Information</th>
</tr>
</thead>
<tbody>
<tr>
<td>Invalid Special Parameter Index (SPI) Recovery</td>
<td>Cisco IOS XE Release 2.1</td>
<td>When an invalid SPI occurs in IPsec packet processing, the Invalid Security Parameter Index Recovery feature allows for an IKE SA to be established. The &quot;IKE&quot; module sends notification of the &quot;Invalid SPI&quot; error to the originating IPsec peer so that Security Association Databases (SADBs) can be resynchronized and successful packet processing can be resumed. The following command was introduced or modified: <code>crypto isakmp invalid-spi-recovery</code>.</td>
</tr>
</tbody>
</table>
The IPsec Dead Peer Detection Periodic Message Option feature allows you to configure your router to query the liveliness of its Internet Key Exchange (IKE) peer at regular intervals. The benefit of this approach over the default approach (on-demand dead peer detection) is earlier detection of dead peers.

- Finding Feature Information, page 33
- Prerequisites for IPsec Dead Peer Detection Periodic Message Option, page 34
- Restrictions for IPsec Dead Peer Detection Periodic Message Option, page 34
- Information About IPsec Dead Peer Detection Periodic Message Option, page 34
- How to Configure IPsec Dead Peer Detection Periodic Message Option, page 35
- Configuration Examples for IPsec Dead Peer Detection Periodic Message Option, page 40
- Additional References, page 44
- Feature Information for Dead Peer Detection Periodic Message Option, page 45

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest caveats and feature information, see Bug Search Tool and 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 module.

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 IPsec Dead Peer Detection Periodic Message Option

Before configuring the IPsec Dead Peer Detection Periodic Message Option feature, you should have the following:

- Familiarity with configuring IP Security (IPsec).
- An IKE peer that supports DPD (dead peer detection). Implementations that support DPD include the CiscoVPN 3000 concentrator, Cisco PIX Firewall, Cisco VPN Client, and Cisco IOS XE software in all modes of operation--site-to-site, Easy VPN remote, and Easy VPN server.

Restrictions for IPsec Dead Peer Detection Periodic Message Option

Using periodic DPD potentially allows the router to detect an unresponsive IKE peer with better response time when compared to on-demand DPD. However, use of periodic DPD incurs extra overhead. When communicating to large numbers of IKE peers, you should consider using on-demand DPD instead.

Information About IPsec Dead Peer Detection Periodic Message Option

How DPD and Cisco IOS XE Keepalive Features Work

DPD and Cisco IOS XE keepalives function on the basis of the timer. If the timer is set for 10 seconds, the router will send a "hello" message every 10 seconds (unless, of course, the router receives a "hello" message from the peer). The benefit of IOS keepalives and periodic DPD is earlier detection of dead peers. However, IOS keepalives and periodic DPD rely on periodic messages that have to be sent with considerable frequency. The result of sending frequent messages is that the communicating peers must encrypt and decrypt more packets.

DPD also has an on-demand approach. The contrasting on-demand approach is the default. With on-demand DPD, messages are sent on the basis of traffic patterns. For example, if a router has to send outbound traffic and the liveliness of the peer is questionable, the router sends a DPD message to query the status of the peer. If a router has no traffic to send, it never sends a DPD message. If a peer is dead, and the router never has any traffic to send to the peer, the router will not find out until the IKE or IPsec security association (SA) has to be rekeyed (the liveliness of the peer is unimportant if the router is not trying to communicate with the peer). On the other hand, if the router has traffic to send to the peer, and the peer does not respond, the router will initiate a DPD message to determine the state of the peer.
Using the IPsec Dead Peer Detection Periodic Message Option

With the IPsec Dead Peer Detection Periodic Message Option feature, you can configure your router so that DPD messages are “forced” at regular intervals. This forced approach results in earlier detection of dead peers. For example, if a router has no traffic to send, a DPD message is still sent at regular intervals, and if a peer is dead, the router does not have to wait until the IKE SA times out to find out.

If you want to configure the DPD periodic message option, you should use the `crypto isakmp keepalive` command with the `periodic` keyword. If you do not configure the `periodic` keyword, the router defaults to the on-demand approach.

```
Note
When the `crypto isakmp keepalive` command is configured, the Cisco IOS software negotiates the use of Cisco IOS keepalives or DPD, depending on which protocol the peer supports.
```

Using DPD and Cisco IOS XE Keepalive Features with Multiple Peers in the Crypto Map

DPD and Cisco IOS XE keepalive features can be used in conjunction with multiple peers in the crypto map to allow for stateless failover. DPD allows the router to detect a dead IKE peer, and when the router detects the dead state, the router deletes the IPsec and IKE SAs to the peer. If you configure multiple peers, the router will switch over to the next listed peer for a stateless failover.

Using DPD in an Easy VPN Remote Configuration

DPD can be used in an Easy VPN remote configuration. See the section Configuring DPD for an Easy VPN Remote section.

How to Configure IPsec Dead Peer Detection Periodic Message Option

Configuring a Periodic DPD Message

To configure a periodic DPD message, perform the following steps.

```
SUMMARY STEPS

1. enable
2. configure terminal
3. crypto isakmp keepalive seconds [retries] [periodic | on-demand]
```
## 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>• 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><strong>Example:</strong> Router# configure terminal</td>
<td></td>
</tr>
<tr>
<td><strong>Step 3</strong> crypto isakmp keepalive seconds [retries] [periodic</td>
<td>on-demand]</td>
</tr>
<tr>
<td><strong>Example:</strong> Router (config)# crypto isakmp keepalive 10 periodic</td>
<td>• seconds --When the periodic keyword is used, this argument is the number of seconds between DPD messages; the range is from 10 to 3600 seconds.</td>
</tr>
<tr>
<td></td>
<td>When the on-demand keyword is used, this argument is the number of seconds during which traffic is not received from the peer before DPD retry messages are sent if there is data (IPSec) traffic to send; the range is from 10 to 3600 seconds.</td>
</tr>
<tr>
<td></td>
<td><strong>Note</strong> If you do not specify a time interval, an error message appears.</td>
</tr>
<tr>
<td></td>
<td>• retry-seconds --(Optional) Number of seconds between DPD retry messages if the DPD retry message is missed by the peer; the range is from 2 to 60 seconds.</td>
</tr>
<tr>
<td></td>
<td>Once 1 DPD message is missed by the peer, the router moves to a more aggressive state and sends the DPD retry message at the faster retry interval, which is the number of seconds between DPD retries if the DPD message is missed by the peer. The default DPD retry message is sent every 2 seconds. Five aggressive DPD retry messages can be missed before the tunnel is marked as down.</td>
</tr>
<tr>
<td></td>
<td><strong>Note</strong> To configure DPD with IPsec High Availability (HA), the recommendation is to use a value other than the default (which is 2 seconds). A keepalive timer of 10 seconds with 5 retries seems to work well with HA because of the time that it takes for the router to get into active mode.</td>
</tr>
<tr>
<td></td>
<td>• periodic --(Optional) DPD messages are sent at regular intervals.</td>
</tr>
<tr>
<td></td>
<td>• on-demand --(Optional) The default behavior. DPD retries are sent on demand.</td>
</tr>
<tr>
<td></td>
<td><strong>Note</strong> Because this option is the default, the on-demand keyword does not appear in configuration output.</td>
</tr>
</tbody>
</table>
Configuring DPD and Cisco IOS XE Keepalives with Multiple Peers in the Crypto Map

To configure DPD and IOS keepalives to be used in conjunction with the crypto map to allow for stateless failover, perform the following steps. This configuration will cause a router to cycle through the peer list when it detects that the first peer is dead.

**SUMMARY STEPS**

1. enable
2. configure terminal
3. crypto map  map-name  seq-num  ipsec-isakmp
4. set peer  \{host-name [dynamic] | ip-address\}
5. set transform-set  transform-set-name
6. match address  [access-list-id | 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: Router&gt; enable</td>
<td>• Enter your password if prompted.</td>
</tr>
</tbody>
</table>

**Step 2** configure terminal

Example: Router# configure terminal

Enters global configuration mode.

**Step 3** crypto map  map-name  seq-num  ipsec-isakmp

Example: Router (config)# crypto map green 1 ipsec-isakmp

Enters crypto map configuration mode and creates or modifies a crypto map entry.

• The ipsec-isakmp keyword indicates that IKE will be used to establish the IPsec SAs for protecting the traffic specified by this crypto map entry.

**Step 4** set peer  \{host-name [dynamic] | ip-address\}

Example: Router (config-crypto-map)# set peer 10.12.12.12

Specifies an IPsec peer in a crypto map entry.

• You can specify multiple peers by repeating this command.
### Configuring DPD for an Easy VPN Remote

To configure DPD in an Easy VPN remote configuration, perform the following steps. This configuration also will cause a router to cycle through the peer list when it detects that the first peer is dead.

**Note**
Cisco IOS XE keepalives are not supported for Easy VPN remote configurations.

**SUMMARY STEPS**

1. `enable`
2. `configure terminal`
3. `crypto ipsec client ezvpn name`
4. `connect {auto | manual}`
5. `group group-name key group-key`
6. `mode {client | network-extension}`
7. `peer {ipaddress | 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&gt; enable</td>
<td></td>
</tr>
<tr>
<td>Command or Action</td>
<td>Purpose</td>
</tr>
<tr>
<td>-------------------</td>
<td>---------</td>
</tr>
<tr>
<td><strong>Step 2</strong></td>
<td><strong>configure terminal</strong></td>
</tr>
<tr>
<td><strong>Example:</strong></td>
<td>Router# configure terminal</td>
</tr>
<tr>
<td><strong>Step 3</strong></td>
<td><strong>crypto ipsec client ezvpn name</strong></td>
</tr>
<tr>
<td><strong>Example:</strong></td>
<td>Router (config)# crypto ipsec client ezvpn ezvpn-config1</td>
</tr>
<tr>
<td><strong>Step 4</strong></td>
<td>**connect {auto</td>
</tr>
<tr>
<td><strong>Example:</strong></td>
<td>Router (config-crypto-ezvpn)# connect manual</td>
</tr>
<tr>
<td><strong>Step 5</strong></td>
<td><strong>group group-name key group-key</strong></td>
</tr>
<tr>
<td><strong>Example:</strong></td>
<td>Router (config-crypto-ezvpn)# group unity key preshared</td>
</tr>
<tr>
<td><strong>Step 6</strong></td>
<td>**mode {client</td>
</tr>
<tr>
<td><strong>Example:</strong></td>
<td>Router (config-crypto-ezvpn)# mode client</td>
</tr>
<tr>
<td><strong>Step 7</strong></td>
<td>**peer {ipaddress</td>
</tr>
<tr>
<td><strong>Example:</strong></td>
<td>Router (config-crypto-ezvpn)# peer 10.10.10.10</td>
</tr>
</tbody>
</table>

**Verifying That DPD Is Enabled**

DPD allows the router to clear the IKE state when a peer becomes unreachable. If DPD is enabled and the peer is unreachable for some time, you can use the `clear crypto session` command to manually clear IKE and IPsec SAs.

The `debug crypto isakmp` command can be used to verify that DPD is enabled.
SUMMARY STEPS

1. enable
2. clear crypto session [local ip-address [port local-port]] [remote ip-address [port remote-port]] [fvrf vrf-name] [ivrf vrf-name]
3. debug crypto isakmp

DETAILED STEPS

<table>
<thead>
<tr>
<th>Command or Action</th>
<th>Purpose</th>
</tr>
</thead>
</table>
| **Step 1** enable | Enables privileged EXEC mode.  
  Example:  
  Router> enable |
|                    | • Enter your password if prompted. |
| **Step 2** clear crypto session [local ip-address [port local-port]] [remote ip-address [port remote-port]] [fvrf vrf-name] [ivrf vrf-name] | Deletes crypto sessions (IPsec and IKE SAs).  
  Example:  
  Router# clear crypto session |
| **Step 3** debug crypto isakmp | Displays messages about IKE events.  
  Example:  
  Router# debug crypto isakmp |

Configuration Examples for IPsec Dead Peer Detection Periodic Message Option

Site-to-Site Setup with Periodic DPD Enabled Example

The following configurations are for a site-to-site setup with no periodic DPD enabled. The configurations are for the IKE Phase 1 policy and for the IKE preshared key.

IKE Phase 1 Policy

crypto isakmp policy 1  
  encryption aes  
  authentication pre-share
IKE Preshared Key

crypto isakmp key kd94j1ksldz address 10.2.80.209 255.255.255.0

crypto isakmp keepalive 10 periodic
crypto ipsec transform-set Trans1 esp-aes esp-sha-hmac

interface
ip address 10.1.32.14 255.255.255.0
speed auto

Easy VPN Remote with DPD Enabled Example

The following configuration tells the router to send a periodic DPD message every 30 seconds. If the peer fails to respond to the DPD R_U_THERE message, the router will resend the message every 20 seconds (four transmissions altogether).

crypto isakmp keepalive 30 20 periodic

crypto ipsec client ezvpn ezvpn-config
connect auto
group unity key preshared
mode client
peer 10.2.80.209

interface FastEthernet0
ip address 10.2.3.4 255.255.255.0
half-duplex
crypto ipsec client ezvpn ezvpn-config inside

interface FastEthernet0
ip address 10.1.32.14 255.255.255.0
speed auto
crypto ipsec client ezvpn ezvpn-config outside

Verifying DPD Configuration Using the debug crypto isakmp Command Example

The following sample output from the debug crypto isakmp command verifies that IKE DPD is enabled:


To see that IKE DPD is enabled (and that the peer supports DPD): when periodic DPD is enabled, you should see the following debug messages at the interval specified by the command:


The above message corresponds to sending the DPD R_U_THERE message.

*Mar 25 15:18:52.123: ISAKMP (0:268435457): received packet from 10.2.80.209
dport 500 sport 500 Global (I) QM_IDLE
Verifying DPD Configuration Using the debug crypto isakmp Command Example

```
Mar 25 15:18:52.123: ISAKMP: set new node -443923643 to QM_IDLE
Mar 25 15:18:52.131: ISAKMP:(0:1:HW:2): processing NOTIFY R_U_THERE_ACK protocol 1
  spi 0, message ID = -443923643, sa = 81BA4DD4
Mar 25 15:18:52.135: ISAKMP:(0:1:HW:2): DPD/R_U_THERE_ACK received from peer
  10.2.80.209, sequence 0x9
  reason "informational (in) state 1"
The above message corresponds to receiving the acknowledge (ACK) message from the peer.

Router#
```

```
Mar 25 15:47:35.335: ISAKMP: set new node -90798077 to QM_IDLE
  500 peer_port 500 (I) QM_IDLE
Mar 25 15:47:35.347: ISAKMP:(0:1:HW:2):Input = IKE_MESG_FROM_TIMER,
  IKE_TIMER_IM_ALIVE
Mar 25 15:47:36.611: ISAKMP: set new node -1592471565 to QM_IDLE
  500 peer_port 500 (I) QM_IDLE
  IKE_TIMER_PEERS_ALIVE
  500 peer_port 500 (I) QM_IDLE
Mar 25 15:47:37.359: ISAKMP:(0:1:HW:2):incrementing error counter on sa:
  PEERS_ALIVE TIMER
  500 peer_port 500 (I) QM_IDLE
  IKE_TIMER_PEERS_ALIVE
  500 peer_port 500 (I) QM_IDLE
  PEERS_ALIVE TIMER
  500 peer_port 500 (I) QM_IDLE
  IKE_TIMER_PEERS_ALIVE
  500 peer_port 500 (I) QM_IDLE
Mar 25 15:47:43.391: ISAKMP: set new node 1758739401 to QM_IDLE
Mar 25 15:47:43.395: ISAKMP: set new node 1758739401 to QM_IDLE
  500 peer_port 500 (I) QM_IDLE
Mar 25 15:47:43.395: ISAKMP: set new node 1758739401 to QM_IDLE
  500 peer_port 500 (I) QM_IDLE
Mar 25 15:47:43.399: ISAKMP: unlock IPSEC struct 0x81E5C4E8 from delete_siblings, count 0
```

The above message shows what happens when the remote peer is unreachable. The router sends one DPD R_U_THERE message and four retransmissions before it finally deletes the IPsec and IKE SAs.

**DPD and Cisco IOS XE Keepalives Used in Conjunction with Multiple Peers in a Crypto Map Example**

The following example shows that DPD and Cisco IOS XE keepalives are used in conjunction with multiple peers in a crypto map configuration when IKE will be used to establish the security associations (SAs). In this example, an SA could be set up to the IPsec peer at 10.0.0.1, 10.0.0.2, or 10.0.0.3.

```
crypto map green 1 ipsec-ike
  set peer 10.0.0.1
  set peer 10.0.0.2
  set peer 10.0.0.3
  set transform-set txfm
  match address 101
```
DPD Used in Conjunction with Multiple Peers for an Easy VPN Remote Example

The following example shows that DPD is used in conjunction with multiple peers in an Easy VPN remote configuration. In this example, an SA could be set up to the IPsec peer at 10.10.10.10, 10.2.2.2, or 10.3.3.3.

```
crypto ipsec client ezvpn ezvpn-config
   connect auto
   group unity key preshared
   mode client
   peer 10.10.10.10
   peer 10.2.2.2
   peer 10.3.3.3
```

Additional References

The following sections provide references related to IPsec Dead Peer Detection Periodic Message Option.

### Related Documents

<table>
<thead>
<tr>
<th>Related Topic</th>
<th>Document Title</th>
</tr>
</thead>
<tbody>
<tr>
<td>Configuring IPsec</td>
<td>Configuring Security for VPNs with IPsec</td>
</tr>
<tr>
<td>IPsec 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>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 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>DPD conforms to the Internet draft &quot;draft-ietf-ipsec-dpd-04.txt,&quot; which is pending publication as an Informational RFC (a number has not yet been assigned).</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/techsupport">http://www.cisco.com/techsupport</a></td>
</tr>
</tbody>
</table>

Feature Information for Dead Peer Detection Periodic Message Option

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 Dead Peer Detection

<table>
<thead>
<tr>
<th>Feature Name</th>
<th>Releases</th>
<th>Feature Information</th>
</tr>
</thead>
<tbody>
<tr>
<td>Dead Peer Detection Periodic Message Option</td>
<td>Cisco IOS XE Release 2.1</td>
<td>This feature allows you to configure your router to query the liveliness of its IKE peer at regular intervals. The benefit of this approach over the default approach (on-demand dead peer detection) is earlier detection of dead peers. The following command was introduced or modified: <code>crypto isakmp keepalive</code>.</td>
</tr>
</tbody>
</table>
IPsec NAT Transparency

The IPsec NAT Transparency feature introduces support for IP Security (IPsec) traffic to travel through Network Address Translation (NAT) or Port Address Translation (PAT) points in the network by addressing many known incompatibilities between NAT and IPsec.

- Finding Feature Information, page 47
- Restrictions for IPsec NAT Transparency, page 47
- Information About IPsec NAT Transparency, page 48
- How to Configure NAT and IPsec, page 52
- Configuration Examples for IPsec and NAT, page 54
- Additional References, page 54
- Feature Information for IPsec NAT Transparency, page 56
- Glossary, page 57

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest caveats and feature information, see Bug Search Tool and 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 module.

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 IPsec NAT Transparency

Although this feature addresses many incompatibilities between NAT and IPsec, the following problems still exist:
Internet Key Exchange (IKE) IP Address and NAT

This incompatibility applies only when IP addresses are used as a search key to find a preshared key. Modification of the IP source or destination addresses by NAT or reverse NAT results in a mismatch between the IP address and the preshared key.

Embedded IP Addresses and NAT

Because the payload is integrity protected, any IP address enclosed within IPsec packets cannot be translated by NAT. Protocols that use embedded IP addresses include FTP, Internet Relay Chat (IRC), Simple Network Management Protocol (SNMP), Lightweight Directory Access Protocol (LDAP), H.323, and Session Initiation Protocol (SIP).

Information About IPsec NAT Transparency

Benefit of IPsec NAT Transparency

Before the introduction of this feature, a standard IPsec virtual private network (VPN) tunnel would not work if there were one or more NAT or PAT points in the delivery path of the IPsec packet. This feature makes NAT IPsec-aware, thereby, allowing remote access users to build IPsec tunnels to home gateways.

Feature Design of IPsec NAT Traversal

The IPsec NAT Transparency feature introduces support for IPsec traffic to travel through NAT or PAT points in the network by encapsulating IPsec packets in a User Datagram Protocol (UDP) wrapper, which allows the packets to travel across NAT devices. The following sections define the details of NAT traversal:

IKE Phase 1 Negotiation NAT Detection

During Internet Key Exchange (IKE) phase 1 negotiation, two types of NAT detection occur before IKE Quick Mode begins--NAT support and NAT existence along the network path.

To detect NAT support, you should exchange the vendor identification (ID) string with the remote peer. During Main Mode (MM) 1 and MM 2 of IKE phase 1, the remote peer sends a vendor ID string payload to its peer to indicate that this version supports NAT traversal. Thereafter, NAT existence along the network path can be determined.

Detecting whether NAT exists along the network path allows you to find any NAT device between two peers and the exact location of NAT. A NAT device can translate the private IP address and port to public value (or from public to private). This translation changes the IP address and port if the packet goes through the device. To detect whether a NAT device exists along the network path, the peers should send a payload with hashes of the IP address and port of both the source and destination address from each end. If both ends calculate the hashes and the hashes match, each peer knows that a NAT device does not exist on the network path between them. If the hashes do not match (that is, someone translated the address or port), then each peer needs to perform NAT traversal to get the IPsec packet through the network.

The hashes are sent as a series of NAT discovery (NAT-D) payloads. Each payload contains one hash; if multiple hashes exist, multiple NAT-D payloads are sent. In most environments, there are only two NAT-D payloads--one for the source address and port and one for the destination address and port. The destination
NAT-D payload is sent first, followed by the source NAT-D payload, which implies that the receiver should expect to process the local NAT-D payload first and the remote NAT-D payload second. The NAT-D payloads are included in the third and fourth messages in Main Mode and in the second and third messages in Aggressive Mode (AM).

**IKE Phase 2 Negotiation NAT Traversal Decision**

While IKE phase 1 detects NAT support and NAT existence along the network path, IKE phase 2 decides whether or not the peers at both ends will use NAT traversal. Quick Mode (QM) security association (SA) payload in QM1 and QM2 is used to for NAT traversal negotiation.

Because the NAT device changes the IP address and port number, incompatibilities between NAT and IPsec can be created. Thus, exchanging the original source address bypasses any incompatibilities.

**UDP Encapsulation of IPsec Packets for NAT Traversal**

In addition to allowing IPsec packets to traverse across NAT devices, UDP encapsulation also addresses many incompatibility issues between IPsec and NAT and PAT. The resolved issues are as follows:

**Incompatibility Between IPsec ESP and PAT Resolved**

If PAT found a legislative IP address and port, it would drop the Encapsulating Security Payload (ESP) packet. To prevent this scenario, UDP encapsulation is used to hide the ESP packet behind the UDP header. Thus, PAT treats the ESP packet as a UDP packet, processing the ESP packet as a normal UDP packet.

**Incompatibility Between Checksums and NAT Resolved**

In the new UDP header, the checksum value is always assigned to zero. This value prevents an intermediate device from validating the checksum against the packet checksum, thereby, resolving the TCP UDP checksum issue because NAT changes the IP source and destination addresses.

**Incompatibility Between Fixed IKE Destination Ports and PAT Resolved**

PAT changes the port address in the new UDP header for translation and leaves the original payload unchanged.
To see how UDP encapsulation helps to send IPSec packets see the figures below.

**Figure 2: Standard IPSec Tunnel Through a NAT/PAT Point (No UDP Encapsulation)**

**Figure 3: IPSec Packet with UDP Encapsulation**
UDP Encapsulated Process for Software Engines Transport Mode and Tunnel Mode ESP Encapsulation

After the IPsec packet is encrypted by a hardware accelerator or a software crypto engine, a UDP header and a non-IKE marker (which is 8 bytes in length) are inserted between the original IP header and ESP header. The total length, protocol, and checksum fields are changed to match this modification. The first figure below shows an IPsec packet before and after transport mode is applied; the second figure below shows an IPsec packet before and after tunnel mode is applied.

![Figure 4: Transport Mode--IPsec Packet Before and After ESP Encapsulation](image)

![Figure 5: Tunnel Mode--IPsec Packet Before and After ESP Encapsulation](image)

**NAT Keepalives**

NAT keepalives are enabled to keep the dynamic NAT mapping alive during a connection between two peers. NAT keepalives are UDP packets with an unencrypted payload of 1 byte. Although the current dead peer detection (DPD) implementation is similar to NAT keepalives, there is a slight difference: DPD is used to detect peer status, while NAT keepalives are sent if the IPsec entity did not send or receive the packet at a specified period of time—valid range is between 5 to 3600 seconds.

If NAT keepalives are enabled (via the `crypto isakmp nat keepalive` command), users should ensure that the idle value is shorter than the NAT mapping expiration time, which is 20 seconds.
How to Configure NAT and IPsec

Configuring NAT Traversal

NAT Traversal is a feature that is auto detected by VPN devices. There are no configuration steps for a router running Cisco IOS XE Release 2.1. If both VPN devices are NAT-T capable, NAT Traversal is auto detected and auto negotiated.

Disabling NAT Traversal

You may wish to disable NAT traversal if you already know that your network uses IPsec-awareness NAT (spi-matching scheme). To disable NAT traversal, use the following commands:

**SUMMARY STEPS**

1. enable
2. configure terminal
3. no crypto ipsec nat-transparency udp-encapsulation

**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 higher privilege levels, such as privileged EXEC mode.</td>
</tr>
<tr>
<td><strong>Example:</strong></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><strong>Example:</strong></td>
<td></td>
</tr>
<tr>
<td>Router# configure terminal</td>
<td></td>
</tr>
<tr>
<td><strong>Step 3</strong> no crypto ipsec nat-transparency udp-encapsulation</td>
<td>Disables NAT traversal.</td>
</tr>
<tr>
<td><strong>Example:</strong></td>
<td></td>
</tr>
<tr>
<td>Router(config)# no crypto ipsec nat-transparency udp-encapsulation</td>
<td></td>
</tr>
</tbody>
</table>
Configuring NAT Keepalives

To configure your router to send NAT keepalives, use the following commands:

SUMMARY STEPS

1. enable
2. configure terminal
3. crypto isakmp nat keepalive seconds

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 higher privilege levels, such as privileged EXEC mode. Enter your password if prompted. Example: Router&gt; enable</td>
</tr>
<tr>
<td><strong>Step 2</strong> configure terminal</td>
<td>Enters global configuration mode. Example: Router# configure terminal</td>
</tr>
<tr>
<td><strong>Step 3</strong> crypto isakmp nat keepalive seconds</td>
<td>Allows an IPsec node to send NAT keepalive packets. Example: Router(config)# crypto isakmp nat keepalive 20</td>
</tr>
</tbody>
</table>

- seconds --The number of seconds between keepalive packets; range is between 5 to 3,600 seconds.

**Note** When the timer is modified, it is modified for every Internet Security Association Key Management Protocol (ISAKMP) security association (SA) when the keepalive for that SA is sent based on the existing timer. A five-percent jitter mechanism value is applied to the timer to avoid security association rekey collisions. If there are many peer routers, and the timer is configured too low, then the router can experience high CPU usage.

Verifying IPsec Configuration

To verify your configuration, perform the following optional steps:

SUMMARY STEPS

1. enable
2. show crypto ipsec sa [map map-name | address | identity] [detail]
## DETAILED STEPS

<table>
<thead>
<tr>
<th>Step 1</th>
<th>Command or Action</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>enable</td>
<td>Enables higher privilege levels, such as privileged EXEC mode. Enter your password if prompted.</td>
</tr>
<tr>
<td></td>
<td>Example:</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Router&gt; enable</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Step 2</th>
<th>Command or Action</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>show crypto ipsec sa [map map-name</td>
<td>address</td>
</tr>
<tr>
<td></td>
<td>Example:</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Router# show crypto ipsec sa</td>
<td></td>
</tr>
</tbody>
</table>

## Configuration Examples for IPsec and NAT

### NAT Keepalives Configuration Example

The following example shows how to enable NAT keepalives to be sent every 20 seconds:

```
crypto isakmp policy 1
  authentication pre-share
crypto isakmp key 1234 address 10.0.0.1
crypto isakmp nat keepalive 20
!
crypto ipsec transform-set t2 esp-des esp-sha-hmac
!
crypto map test2 10 ipsec-isakmp
  set peer 10.0.0.1
  set transform-set t2
  match address 101
```

## Additional References

The following sections provide references related to the IPsec NAT Transparency feature.
## Related Documents

<table>
<thead>
<tr>
<th>Related Topic</th>
<th>Document Title</th>
</tr>
</thead>
<tbody>
<tr>
<td>Additional NAT configuration tasks</td>
<td>• “Configuring NAT for IP Address Conservation” module in the <em>Cisco IOS XE IP Addressing Services Configuration Guide</em></td>
</tr>
<tr>
<td></td>
<td>• “Using Application Level Gateways with NAT” module in the <em>Cisco IOS XE IP Addressing Services Configuration Guide</em></td>
</tr>
<tr>
<td></td>
<td>• “Configuring NAT for High Availability” module in the <em>Cisco IOS XE IP Addressing Services Configuration Guide</em></td>
</tr>
<tr>
<td></td>
<td>• “Integrating NAT with MPLS VPNs” module in the <em>Cisco IOS XE IP Addressing Services Configuration Guide</em></td>
</tr>
<tr>
<td>Additional NAT commands</td>
<td><em>Cisco IOS IP Addressing Services Command Reference</em></td>
</tr>
<tr>
<td>Additional IPsec configuration tasks</td>
<td>“Configuring Security for VPNs with IPsec” module in the <em>Cisco IOS XE Security Configuration Guide: Secure Connectivity</em></td>
</tr>
<tr>
<td>Additional IPsec commands</td>
<td><em>Cisco IOS Security Command Reference</em></td>
</tr>
<tr>
<td>Information on IKE</td>
<td>“Configuring Internet Key Exchange for IPsec VPNs” module in the <em>Cisco IOS XE Security Configuration Guide: Secure Connectivity</em></td>
</tr>
<tr>
<td>Additional information on IKE dead peer detection</td>
<td>“Easy VPN Server” module in the <em>Cisco IOS XE Security Configuration Guide: Secure Connectivity</em></td>
</tr>
</tbody>
</table>

## Standards

<table>
<thead>
<tr>
<th>Standards</th>
<th>Title</th>
</tr>
</thead>
<tbody>
<tr>
<td>No new or modified MIBs are supported by this feature, and support for</td>
<td>--</td>
</tr>
<tr>
<td>existing MIBs 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 software releases, and feature sets, use Cisco MIB Locator found at the following URL: <a href="http://tools.cisco.com/ITDIT/MIBS/servlet/index">http://tools.cisco.com/ITDIT/MIBS/servlet/index</a></td>
</tr>
</tbody>
</table>

RFCs

<table>
<thead>
<tr>
<th>RFCs</th>
<th>Title</th>
</tr>
</thead>
<tbody>
<tr>
<td>RFC 2402</td>
<td>IP Authentication Header</td>
</tr>
<tr>
<td>RFC 2406</td>
<td>IP Encapsulating Security Payload (ESP)</td>
</tr>
</tbody>
</table>

1 Not all supported RFCs are listed.

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 IPsec NAT Transparency

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 IPsec NAT Transparency

<table>
<thead>
<tr>
<th>Feature Name</th>
<th>Releases</th>
<th>Feature Information</th>
</tr>
</thead>
<tbody>
<tr>
<td>IPsec NAT Transparency</td>
<td>Cisco IOS XE Release 2.1</td>
<td>The IPsec NAT Transparency feature introduces support for IP Security (IPsec) traffic to travel through Network Address Translation (NAT) or Port Address Translation (PAT) points in the network by addressing many known incompatibilities between NAT and IPsec. The following commands were introduced or modified: <code>crypto isamkp nat keepalive</code>, <code>access-list (IP extended)</code>, <code>show crypto ipsec sa</code></td>
</tr>
</tbody>
</table>

Glossary

**IKE** --Internet Key Exchange. Hybrid protocol that implements Oakley key exchange and Skeme key exchange inside the Internet Security Association Key Management Protocol (ISAKMP) framework. Although IKE can be used with other protocols, its initial implementation is with IPsec. IKE provides authentication of the IPsec peers, negotiates IPsec keys, and negotiates IPsec security associations (SAs).

**IPsec** --IP Security. Framework of open standards developed by the Internet Engineering Task Force (IETF). IPsec provides security for transmission of sensitive information over unprotected networks such as the Internet. IPsec acts at the network layer, protecting and authenticating IP packets between participating IPsec devices (“peers”), such as Cisco routers.

**NAT** --Network Address Translation. Translates a private IP address used inside the corporation to a public, routable address for use on the outside of the corporation, such as the Internet. NAT is considered a one-to-one mapping of addresses from private to public.

**PAT** --Port Address Translation. Like NAT, PAT also translated private IP address to public, routable addresses. Unlike NAT, PAT provides a many-to-one mapping of private addresses to a public address; each instance of the public address is associated with a particular port number to provide uniqueness. PAT can be used in environments where the cost of obtaining a range of public addresses is too expensive for an organization.
CHAPTER 7

DF Bit Override Functionality with IPsec Tunnels

The DF Bit Override Functionality with IPsec Tunnels feature allows you to configure the setting of the DF bit when encapsulating tunnel mode IPsec traffic on a global or per-interface level. Thus, if the DF bit is set to clear, routers can fragment packets regardless of the original DF bit setting.

- Finding Feature Information, page 59
- Prerequisites for DF Bit Override Functionality with IPsec Tunnels, page 59
- Restrictions for DF Bit Override Functionality with IPsec Tunnels, page 60
- Information About DF Bit Override Functionality with IPsec Tunnels, page 60
- How to Configure DF Bit Override Functionality with IPsec Tunnels, page 61
- Configuration Examples for DB Bit Override Functionality with IPsec Tunnels, page 62
- Additional References, page 62
- Feature Information for DF Bit Override Functionality with IPsec Tunnels, page 64

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest caveats and feature information, see Bug Search Tool and 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 module.

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 DF Bit Override Functionality with IPsec Tunnels

IPsec must be enabled on your router.
Restrictions for DF Bit Override Functionality with IPsec Tunnels

Performance Impact
Because each packet is reassembled at the process level, a significant performance impact occurs at a high data rate. Two major caveats are as follows:

- The reassemble queue can fill up and force fragments to be dropped.
- The traffic is slower because of the process switching.

DF Bit Setting Requirement
If several interfaces share the same crypto map using the local address feature, these interfaces must share the same DF bit setting.

Feature Availability
This feature is available only for IPsec tunnel mode. (IPsec transport mode is not affected because it does not provide an encapsulating IP header.)

Information About DF Bit Override Functionality with IPsec Tunnels

Feature Overview
The DF Bit Override Functionality with IPsec Tunnels feature allows you to specify whether your router can clear, set, or copy the Don’t Fragment (DF) bit from the encapsulated header. A DF bit is a bit within the IP header that determines whether a router is allowed to fragment a packet.

Some user configurations have hosts that perform the following functions:

- Set the DF bit in packets they send
- Use firewalls that block Internet Control Message Protocol (ICMP) errors from outside the firewall, preventing hosts from learning about the maximum transmission unit (MTU) size outside the firewall
- Use IP Security (IPsec) to encapsulate packets, reducing the available MTU size

If your configurations have hosts that prevent you from learning about the available MTU size, you can configure your router to clear the DF bit and fragment the packet.

Note
In compliance with RFC 2401, this feature can be configured globally or per interface. If both levels are configured, the interface configuration will override the global configuration.
How to Configure DF Bit Override Functionality with IPsec Tunnels

Configuring the DF Bit for the Encapsulating Header in Tunnel Mode

To set the DF bit for the encapsulating header in tunnel mode, perform the following steps.

SUMMARY STEPS

1. enable
2. configure terminal
3. crypto ipsec df-bit [clear | set | copy]

DETAILED STEPS

<table>
<thead>
<tr>
<th>Command or Action</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>Step 1</td>
<td></td>
</tr>
<tr>
<td>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</td>
<td></td>
</tr>
<tr>
<td>configure terminal</td>
<td>Enters global configuration mode.</td>
</tr>
<tr>
<td>Example:</td>
<td>Router# configure terminal</td>
</tr>
<tr>
<td>Step 3</td>
<td></td>
</tr>
<tr>
<td>crypto ipsec df-bit</td>
<td>Sets the DF bit for the encapsulating header in tunnel mode for all</td>
</tr>
<tr>
<td>[clear</td>
<td>set</td>
</tr>
<tr>
<td>Example:</td>
<td>To set the DF bit for a specified interface, use the crypto ipsec df-bit</td>
</tr>
<tr>
<td>Router (config)# crypto</td>
<td>command in interface configuration mode.</td>
</tr>
<tr>
<td>Step 3</td>
<td>set</td>
</tr>
</tbody>
</table>

DF bit interface configuration settings override all DF bit global configuration settings.

Verifying DF Bit Setting

To verify the current DF Bit settings on your router, use the show running-config command in EXEC mode.
Configuration Examples for DB Bit Override Functionality with IPsec Tunnels

DF Bit Setting Configuration Example

In following example, the router is configured to globally clear the setting for the DF bit and copy the DF bit on the interface named FastEthernet. Thus, all interfaces except FastEthernet will allow the router to send packets larger than the available MTU size; FastEthernet will allow the router to fragment the packet.

crypto isakmp policy 1  
  hash md5  
  authentication pre-share  
crypto isakmp key Delaware address 192.168.10.66  
crypto isakmp key Key=What-Key address 192.168.11.19

!  
crypto ipsec transform-set exampleset ah-md5-hmac esp-des  
crypto ipsec df-bit clear

!  
crypto map armadillo 1 ipsec-isakmp  
set peer 192.168.10.66  
set transform-set exampleset  
match address 101

!  
crypto map basilisk 1 ipsec-isakmp  
set peer 192.168.11.19  
set transform-set exampleset  
match address 102

!  
interface FastEthernet  
ip address 192.168.10.38 255.255.255.0  
ip broadcast-address 0.0.0.0  
media-type 10BaseT  
crypto map armadillo  
crypto ipsec df-bit copy

!  
interface FastEthernet1  
ip address 192.168.11.75 255.255.255.0  
ip broadcast-address 0.0.0.0  
media-type 10BaseT  
crypto map basilisk

!  
interface Serial0  
no ip address  
ip broadcast-address 0.0.0.0  
no ip route-cache  
no ip mroute-cache

Additional References

The following sections provide references related to the DF Bit Override Functionality with IPsec Tunnels feature.
### Related Documents

<table>
<thead>
<tr>
<th>Related Topic</th>
<th>Document Title</th>
</tr>
</thead>
<tbody>
<tr>
<td>Internet Key Exchange and IPsec networks</td>
<td>Configuring Internet Key Exchange for IPsec VPNs</td>
</tr>
<tr>
<td>IPsec network 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>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 standards 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>RFC</th>
<th>Title</th>
</tr>
</thead>
<tbody>
<tr>
<td>No new or modified RFCs are supported by this feature, and support for existing standards has not been modified by this feature.</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/techsupport">http://www.cisco.com/techsupport</a></td>
</tr>
</tbody>
</table>

Feature Information for DF Bit Override Functionality with IPsec Tunnels

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>
<thead>
<tr>
<th>Feature Name</th>
<th>Releases</th>
<th>Feature Information</th>
</tr>
</thead>
</table>
| DF Bit Override Functionality with IPsec Tunnels                            | Cisco IOS XE Release 2.1        | This feature allows users to specify whether their router can clear, set, or copy the Don’t Fragment (DF) bit from the encapsulated header. A DF bit is a bit within the IP header that determines whether a router is allowed to fragment a packet. The following commands were introduced or modified: 
**crypto ipsec df-bit.** |

Table 7: Feature Information for DF Bit Override Functionality with IPsec Tunnels
CHAPTER 8

IPsec Security Association Idle Timers

When a router running the Cisco IOS XE software creates an IPsec security association (SA) for a peer, resources must be allocated to maintain the SA. The SA requires both memory and several managed timers. For idle peers, these resources are wasted. If enough resources are wasted by idle peers, the router could be prevented from creating new SAs with other peers. The IPsec Security Association Idle Timers feature introduces a configurable idle timer to monitor SAs for activity, allowing SAs for idle peers to be deleted. Benefits of this feature include:

- Increased availability of resources
- Improved scalability of Cisco IOS XE IPsec deployments. Because this feature prevents the wasting of resources by idle peers, more resources will be available to create new SAs as required.

• Finding Feature Information, page 65
• Prerequisites for IPsec Security Association Idle Timers, page 66
• Information About IPsec Security Association Idle Timers, page 66
• How to Configure IPsec Security Association Idle Timers, page 66
• Configuration Examples for IPsec Security Association Idle Timers, page 68
• Additional References, page 68
• Feature Information for IPsec Security Association Idle Timers, page 70

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest caveats and feature information, see Bug Search Tool and 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 module.

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 IPsec Security Association Idle Timers

You must configure Internet Key Exchange (IKE) as described in the "Configuring Internet Key Exchange Security Protocol" chapter of the Cisco IOS XE Security Configuration Guide.

Information About IPsec Security Association Idle Timers

Lifetimes for IPsec Security Associations

The Cisco IOS software currently allows the configuration of lifetimes for IPsec SAs. Lifetimes can be configured globally or per crypto map. There are two lifetimes: a "timed" lifetime and a "traffic-volume" lifetime. A security association expires after the first of these lifetimes is reached.

IPsec Security Association Idle Timers

The IPsec SA idle timers are different from the global lifetimes for IPsec SAs. The expiration of the global lifetime is independent of peer activity. The IPsec SA idle timer allows SAs associated with inactive peers to be deleted before the global lifetime has expired.

If the IPsec SA idle timers are not configured, only the global lifetimes for IPsec SAs are applied. SAs are maintained until the global timers expire, regardless of peer activity.

Note

If the last IPsec SA to a given peer is deleted due to idle timer expiration, the Internet Key Exchange (IKE) SA to that peer will also be deleted.

How to Configure IPsec Security Association Idle Timers

Configuring the IPsec SA Idle Timer Globally

This task configures the IPsec SA idle timer globally. The idle timer configuration will be applied to all SAs.

SUMMARY STEPS

1. enable
2. configure terminal
3. crypto ipsec security-association idle-time seconds
### DETAILED STEPS

<table>
<thead>
<tr>
<th>Command or Action</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Step 1</strong></td>
<td>Enables privileged EXEC mode.</td>
</tr>
<tr>
<td><code>enable</code></td>
<td>- Enter your password if prompted.</td>
</tr>
<tr>
<td><strong>Example:</strong></td>
<td></td>
</tr>
<tr>
<td><strong>Step 2</strong></td>
<td>Enters global configuration mode.</td>
</tr>
<tr>
<td><code>configure terminal</code></td>
<td></td>
</tr>
<tr>
<td><strong>Example:</strong></td>
<td></td>
</tr>
<tr>
<td><strong>Step 3</strong></td>
<td>Configures the IPsec SA idle timer.</td>
</tr>
<tr>
<td><code>crypto ipsec security-association idle-time seconds</code></td>
<td>- The <code>seconds</code> argument specifies the time, in seconds, that the idle timer will allow an inactive peer to maintain an SA. Valid values for the <code>seconds</code> argument range from 60 to 86400.</td>
</tr>
<tr>
<td><strong>Example:</strong></td>
<td></td>
</tr>
</tbody>
</table>

### Configuring the IPsec SA Idle Timer per Crypto Map

This task configures the IPsec SA idle timer for a specified crypto map. The idle timer configuration will be applied to all SAs under the specified crypto map.

### SUMMARY STEPS

1. `enable`
2. `configure terminal`
3. `crypto map map-name seq-number ipsec-isakmp`
4. `set security-association idle-time seconds`

### DETAILED STEPS

<table>
<thead>
<tr>
<th>Command or Action</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Step 1</strong></td>
<td>Enables privileged EXEC mode.</td>
</tr>
<tr>
<td><code>enable</code></td>
<td>- Enter your password if prompted.</td>
</tr>
<tr>
<td><strong>Example:</strong></td>
<td></td>
</tr>
<tr>
<td>Step</td>
<td>Command or Action</td>
</tr>
<tr>
<td>------</td>
<td>-------------------</td>
</tr>
<tr>
<td>2</td>
<td>configure terminal</td>
</tr>
<tr>
<td></td>
<td>Example:</td>
</tr>
<tr>
<td></td>
<td>Router# configure terminal</td>
</tr>
<tr>
<td>3</td>
<td>crypto map map-name seq-number ipsec-isakmp</td>
</tr>
<tr>
<td></td>
<td>Example:</td>
</tr>
<tr>
<td></td>
<td>Router(config)# crypto map test 1 ipsec-isakmp</td>
</tr>
</tbody>
</table>
| 4    | set security-association idle-time seconds | Specifies the maximum amount of time for which the current peer can be idle before the default peer is used.  
- The seconds argument is the number of seconds for which the current peer can be idle before the default peer is used.  
Valid values are 60 to 86400. |
|      | Example:          |         |
|      | Router(config-crypto-map)# set security-association idle-time 600 | |

### Configuration Examples for IPsec Security Association Idle Timers

#### Configuring the IPsec SA Idle Timer Globally Example

The following example globally configures the IPsec SA idle timer to drop SAs for inactive peers after 600 seconds:

```
crypto ipsec security-association idle-time 600
```

#### Configuring the IPsec SA Idle Timer per Crypto Map Example

The following example configures the IPsec SA idle timer for the crypto map named test to drop SAs for inactive peers after 600 seconds:

```
crypto map test 1 ipsec-isakmp  
set security-association idle-time 600
```

### Additional References

The following sections provide references related to the IPsec Security Association Idle Timers feature.
### Related Documents

<table>
<thead>
<tr>
<th>Related Topic</th>
<th>Document Title</th>
</tr>
</thead>
<tbody>
<tr>
<td>Additional information about configuring IKE</td>
<td>Internet Key Exchange for IPsec VPNs</td>
</tr>
</tbody>
</table>
| Additional information about configuring global lifetimes for IPsec SAs | • Configuring Security for VPNs with IPsec  
  • IPsec Preferred Peer |
| Additional Security commands | Cisco IOS Security Command Reference |

### 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>MIB</th>
<th>MIBs Link</th>
</tr>
</thead>
</table>
| No new or modified MIBs are supported by this feature, and support for existing MIBs has not been modified by this feature. | 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:  
http://www.cisco.com/go/mibs |

### RFCs

<table>
<thead>
<tr>
<th>RFC</th>
<th>Title</th>
</tr>
</thead>
<tbody>
<tr>
<td>No new or modified RFCs are supported by this feature, and support for existing RFCs has not been modified by this feature.</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</td>
<td><a href="http://www.cisco.com/techsupport">http://www.cisco.com/techsupport</a></td>
</tr>
<tr>
<td>documentation and tools for troubleshooting and resolving technical issues</td>
<td></td>
</tr>
<tr>
<td>with Cisco products and technologies.</td>
<td></td>
</tr>
<tr>
<td>To receive security and technical information about your products, you</td>
<td></td>
</tr>
<tr>
<td>can subscribe to various services, such as the Product Alert Tool (accessed</td>
<td></td>
</tr>
<tr>
<td>from Field Notices), the Cisco Technical Services Newsletter, and Really</td>
<td></td>
</tr>
<tr>
<td>Simple Syndication (RSS) Feeds.</td>
<td></td>
</tr>
<tr>
<td>Access to most tools on the Cisco Support website requires a Cisco.com</td>
<td></td>
</tr>
<tr>
<td>user ID and password.</td>
<td></td>
</tr>
</tbody>
</table>

**Feature Information for IPsec Security Association Idle Timers**

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 8: Feature Information for IPsec Security Association Idle Timers

<table>
<thead>
<tr>
<th>Feature Name</th>
<th>Releases</th>
<th>Feature Information</th>
</tr>
</thead>
<tbody>
<tr>
<td>IPsec Security Association Idle Timers</td>
<td>Cisco IOS XE Release 2.1</td>
<td>When a router running the Cisco IOS XE software creates an IPsec security association (SA) for a peer, resources must be allocated to maintain the SA. The SA requires both memory and several managed timers. For idle peers, these resources are wasted. If enough resources are wasted by idle peers, the router could be prevented from creating new SAs with other peers. The IPsec Security Association Idle Timers feature introduces a configurable idle timer to monitor SAs for activity, allowing SAs for idle peers to be deleted. The following command was introduced or modified: <code>crypto ipsec security-association idle-time</code>.</td>
</tr>
<tr>
<td></td>
<td>Cisco IOS XE Release 2.1</td>
<td>The <code>set security-association idle-time</code> command was added, allowing for the configuration of an IPsec idle timer for a specified crypto map. The following command was introduced or modified: <code>set security-association idle-time</code>.</td>
</tr>
</tbody>
</table>
CHAPTER 9
IPv6 IPsec Quality of Service

The IPv6 IPsec QoS feature allows the quality of service (QoS) policies to be applied to IPv6 IPsec.

- Finding Feature Information, page 73
- Information About IPv6 IPsec QoS, page 73
- How to Configure IPv6 IPsec QoS, page 74
- Configuration Examples for QoS, page 79
- Additional References for IPv6 IPsec QoS, page 80
- Feature Information for IPv6 IPsec QoS, page 81

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest caveats and feature information, see Bug Search Tool and 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.

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.

Information About IPv6 IPsec QoS

IPv6 IPsec QoS Overview

The IPv6 IPsec QoS feature applies the quality of service (QoS) policies to IPv6 IPsec. This feature supports the following functionalities:

- Crypto LLQ QoS—Traffic that is classified by QoS and marked as priority level 1 or 2 by traditional Cisco Modular QoS CLI (MQC) QoS configuration, for example PAK priority, is enqueued to the priority queue before the crypto processor. The low latency queuing (LLQ) for IPsec encryption engines helps reduce packet latency for priority traffic.
• IPsec QoS Pre-Classify—QoS pre-classify is configured under a crypto map to enable IPsec to save the original Layer 3 and Layer 4 header before the encryption so that QoS can do the classification using the saved header.

• QoS group-based LLQ—the QoS group-based LLQ feature allows IPsec to check the LLQ QoS group setting to determine whether a packet is a high priority packet before it is enqueued to low latency queuing (LLQ).

How to Configure IPv6 IPsec QoS

Configuring Crypto LLQ QoS

When IPsec and QoS are configured on a physical interface and if the QoS policy has priority class, IPSec will classify the packet based on the policy attached to the interface. It will enqueue the packet matching priority class into Low Latency Queue. The high-priority packet will be enqueued to low latency queuing (LLQ).

Perform this task to attach a service policy to the output interface and enable LLQ for IPsec encryption engines.

SUMMARY STEPS

1. enable
2. configure terminal
3. interface physical-interface-name
4. ipv6 address {ipv6-address /prefix-length | prefix-name sub-bits/prefix-length}
5. service-policy output policy-map
6. ipv6 crypto map map-name
7. end

DETAILED STEPS

<table>
<thead>
<tr>
<th>Command or Action</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Step 1</strong></td>
<td></td>
</tr>
<tr>
<td>enable</td>
<td>Enables privileged EXEC mode.</td>
</tr>
<tr>
<td>Example:</td>
<td>• Enter your password if prompted.</td>
</tr>
<tr>
<td>Device&gt; enable</td>
<td></td>
</tr>
<tr>
<td><strong>Step 2</strong></td>
<td></td>
</tr>
<tr>
<td>configure terminal</td>
<td>Enters global configuration mode.</td>
</tr>
<tr>
<td>Example:</td>
<td></td>
</tr>
<tr>
<td>Device# configure terminal</td>
<td></td>
</tr>
</tbody>
</table>
### Purpose

<table>
<thead>
<tr>
<th>Command or Action</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Step 3</strong></td>
<td><strong>interface</strong> <em>physical-interface-name</em></td>
</tr>
<tr>
<td><strong>Example:</strong></td>
<td>Device(config)# interface GigabitEthernet0/0/1</td>
</tr>
<tr>
<td><strong>Step 4</strong></td>
<td><strong>ipv6 address</strong> `{ipv6-address /prefix-length</td>
</tr>
<tr>
<td><strong>Example:</strong></td>
<td>Device(config-if)# ipv6 address 2001:DB8:FFFF::2/64</td>
</tr>
<tr>
<td><strong>Step 5</strong></td>
<td><strong>service-policy output</strong> <em>policy-map</em></td>
</tr>
<tr>
<td><strong>Example:</strong></td>
<td>Device(config-if)# service-policy output p1</td>
</tr>
<tr>
<td><strong>Step 6</strong></td>
<td><strong>ipv6 crypto map</strong> <em>map-name</em></td>
</tr>
<tr>
<td><strong>Example:</strong></td>
<td>Device(config-if)# ipv6 crypto map CMAP_1</td>
</tr>
<tr>
<td><strong>Step 7</strong></td>
<td><strong>end</strong></td>
</tr>
<tr>
<td><strong>Example:</strong></td>
<td>Device(config-if)# end</td>
</tr>
</tbody>
</table>

### Configuring QoS Pre-classify

#### Configuring Pre-classify on the Crypto Map

The **qos pre-classify** command is applied on the crypto map, allowing configuration on a per-tunnel basis. QoS policy is applied to Packets based on the L3 and L4 Header before encryption.

Perform this task to apply the QoS pre-classify on the crypto map.
### SUMMARY STEPS

1. `enable`
2. `configure terminal`
3. `ipv6 crypto map map-name`
4. `qos pre-classify`
5. `end`

### DETAILED STEPS

<table>
<thead>
<tr>
<th>Step</th>
<th>Command or Action</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>Step 1</td>
<td><code>enable</code></td>
<td>Enables privileged EXEC mode.</td>
</tr>
<tr>
<td></td>
<td>Example: <code>Device&gt; enable</code></td>
<td>- Enter your password if prompted.</td>
</tr>
<tr>
<td>Step 2</td>
<td><code>configure terminal</code></td>
<td>Enters global configuration mode.</td>
</tr>
<tr>
<td></td>
<td>Example: <code>Device# configure terminal</code></td>
<td></td>
</tr>
<tr>
<td>Step 3</td>
<td><code>ipv6 crypto map map-name</code></td>
<td>Enters crypto map configuration mode and specifies the crypto map to be configured.</td>
</tr>
<tr>
<td></td>
<td>Example: <code>Device(config-if)# ipv6 crypto map CM_V6</code></td>
<td></td>
</tr>
<tr>
<td>Step 4</td>
<td><code>qos pre-classify</code></td>
<td>Enables QoS pre-classify on the crypto map.</td>
</tr>
<tr>
<td></td>
<td>Example: <code>Device(config-if)# qos pre-classify</code></td>
<td></td>
</tr>
<tr>
<td>Step 5</td>
<td><code>end</code></td>
<td>Exits interface configuration mode and returns to privileged EXEC mode.</td>
</tr>
<tr>
<td></td>
<td>Example: <code>Device(config-if)# end</code></td>
<td></td>
</tr>
</tbody>
</table>

#### Configuring Pre-classify on the Tunnel Interface

The `qos pre-classify` command is applied on the IPv6 IPsec tunnel interface, making QoS a configuration option on a per-tunnel basis.

Perform this task to apply the QOS pre-classify on the tunnel interface.
**SUMMARY STEPS**

1. enable
2. configure terminal
3. interface tunnel-interface-name
4. ipv6 address \{ipv6-address /prefix-length | prefix-name sub-bits/prefix-length\}
5. qos pre-classify
6. end

**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> Device&gt; enable</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><strong>Example:</strong> Device# configure terminal</td>
<td></td>
</tr>
<tr>
<td><strong>Step 3</strong> interface tunnel-interface-name</td>
<td>Enters interface configuration mode and specifies the tunnel or virtual interface to configure.</td>
</tr>
<tr>
<td><strong>Example:</strong> Device(config)# interface Tunnel1</td>
<td></td>
</tr>
<tr>
<td><strong>Step 4</strong> ipv6 address {ipv6-address /prefix-length</td>
<td>prefix-name sub-bits/prefix-length}</td>
</tr>
<tr>
<td><strong>Example:</strong> Device(config-if)# ipv6 address 2001:DB8::FFFF::2/64</td>
<td></td>
</tr>
<tr>
<td><strong>Step 5</strong> qos pre-classify</td>
<td>Enables QoS pre-classify on the tunnel interface.</td>
</tr>
<tr>
<td><strong>Example:</strong> Device(config-if)# qos pre-classify</td>
<td></td>
</tr>
<tr>
<td><strong>Step 6</strong> end</td>
<td>Exits interface configuration mode and returns to privileged EXEC mode.</td>
</tr>
<tr>
<td><strong>Example:</strong> Device(config-if)# end</td>
<td></td>
</tr>
</tbody>
</table>
Configuring LLQ QoS Group

The `platform ipsec llq qos-group` command enables low latency queuing for traffic that matches the QoS groups configured with this command.

Perform this task to enable LLQ for QoS groups.

**SUMMARY STEPS**

1. `enable`
2. `configure terminal`
3. `platform ipsec llq qos-group group-number`
4. `end`

**DETAILED STEPS**

<table>
<thead>
<tr>
<th>Command or Action</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Step 1</strong></td>
<td><code>enable</code>&lt;br&gt;Example: <code>Device&gt; enable</code>&lt;br&gt;Enables privileged EXEC mode.&lt;br&gt;• Enter your password if prompted.</td>
</tr>
<tr>
<td><strong>Step 2</strong></td>
<td><code>configure terminal</code>&lt;br&gt;Example: <code>Device# configure terminal</code>&lt;br&gt;Enters global configuration mode.</td>
</tr>
<tr>
<td><strong>Step 3</strong></td>
<td><code>platform ipsec llq qos-group group-number</code>&lt;br&gt;Example: <code>Device(config)# platform ipsec llq qos-group 1</code>&lt;br&gt;Specifies the QoS group to enable LLQ. Valid values are from 1 to 99.</td>
</tr>
<tr>
<td><strong>Step 4</strong></td>
<td><code>end</code>&lt;br&gt;Example: <code>Device(config-if)# end</code>&lt;br&gt;Exits interface configuration mode and returns to privileged EXEC mode.</td>
</tr>
</tbody>
</table>
Configuration Examples for QoS

Example: Configuring Crypto LLQ QoS

The following example shows how to specify the service policy map to the output interface and enable an IPv6 crypto map on an interface.

```
! class-map match-all c2
 match precedence 5 6 7
class-map match-all c1
 match precedence 0 1 2 3

policy-map p1
 class c1
 priority percent 10
 class c2
 bandwidth remaining percent 3

crypto map ipv6 CMAP_1 1 ipsec-isakmp
 set peer address 2001:DB8:FFFF::1
 set transform-set ESP-3DES-SHA
 match address 102

interface GigabitEthernet0/0/1
 ipv6 address 2001:DB8:FFFF::2/64
 ipv6 crypto map CMAP_1
 service-policy output p1
```

Example: Configuring Pre-classify on the Crypto Map

The following example shows how to enable QoS pre-classification using the `qos pre-classify` command on the crypto map CM_V6.

```
! crypto map ipv6 CM_V6 10 ipsec-isakmp
 match address ACL_IPV6_1
 set transform-set set1
 set peer 2001:DB8:FFFF::1
 qos pre-classify

interface GigabitEthernet0/0/1
 ipv6 address 2001:DB8:FFFF::2/64
 service-policy output policy1
 ipv6 crypto map CM_V6
```

Example: Configuring Pre-classify on the Tunnel Interface

The following example shows how to enable QoS pre-classification using the `qos pre-classify` command on the tunnel interface tunnel1.

```
interface GigabitEthernet1/1/2
 ipv6 address 2001:DB8:3::F/64
 service-policy output policy1

interface Tunnel1
 ipv6 address 2001:DB8:2::F/64
```
Example: Configuring LLQ QoS Group

The following example shows how to configure low latency queuing on a QoS group.

```
! 
platform ipsec liq qos-group 1
platform ipsec liq qos-group 49
!
crypto map ipv6 cmap 1 ipsec-isakmp
    set peer 2001:DB8:FFFF:F:1::E/64
    set security-association lifetime seconds 600
    set transform-set aes-192
    match address 102
!
class-map match-all c1
    match precedence 5
class-map match-all c2
    match precedence 2
class-map match-all c3
    match precedence 4
class-map match-all c4
    match precedence 3
!
policy-map p1
    class c3
        set qos-group 20
    class c1
        set qos-group 49
    class c4
        set qos-group 77
!
policy-map p2
    class class-default
        set qos-group 1
!
interface GigabitEthernet0/2/0
    ipv6 address
    negotiation auto
    cdp enable
    ipv6 crypto map cmap
    service-policy input p2
!
interface GigabitEthernet0/2/7
    ipv6 address 2001:DB8:FFFF:F:1::F/64
    negotiation auto
    cdp enable
    service-policy input p1
```

Additional References for IPv6 IPsec QoS

<table>
<thead>
<tr>
<th>Related Topic</th>
<th>Document Title</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cisco IOS commands</td>
<td>Cisco IOS Master Command List, All Releases</td>
</tr>
</tbody>
</table>
Feature Information for IPv6 IPsec QoS

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.
The IPv6 IPsec QoS feature allows the QoS policies to be applied to IPv6 IPsec. This feature supports the following functionalities:
- Crypto LLQ QoS
- IPsec QoS Pre-Classify
- QoS group-based LLQ

The following command was modified: `ipv6 crypto map`

<table>
<thead>
<tr>
<th>Feature Name</th>
<th>Releases</th>
<th>Feature Information</th>
</tr>
</thead>
<tbody>
<tr>
<td>IPv6 IPsec QoS</td>
<td>15.4(1)S</td>
<td>The IPv6 IPsec QoS feature allows the QoS policies to be applied to IPv6 IPsec. This feature supports the following functionalities:</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Crypto LLQ QoS</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• IPsec QoS Pre-Classify</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• QoS group-based LLQ</td>
</tr>
<tr>
<td></td>
<td></td>
<td>The following command was modified: <code>ipv6 crypto map</code></td>
</tr>
</tbody>
</table>
IPv6 Virtual Tunnel Interface

Cisco IOS IPv6 security features for your Cisco networking devices can protect your network against degradation or failure and also against data loss or compromise resulting from intentional attacks and from unintended but damaging mistakes by well-meaning network users.

Cisco IOS IPsec functionality provides network data encryption at the IP packet level, offering robust, standards-based security. IPsec provides data authentication and antireplay services in addition to data confidentiality services.

IPsec is a mandatory component of IPv6 specification. IPv6 IPsec tunnel mode and encapsulation is used to protect IPv6 unicast and multicast traffic. This document provides information about implementing IPsec in IPv6 security.

- Finding Feature Information, page 83
- Information About IPv6 Virtual Tunnel Interface, page 84
- How to Configure IPv6 Virtual Tunnel Interface, page 85
- Configuration Examples for IPv6 Virtual Tunnel Interface, page 98
- Additional References, page 98
- Feature Information for IPv6 Virtual Tunnel Interface, page 99

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest caveats and feature information, see Bug Search Tool and 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 module.

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.
IPsec for IPv6

IP Security, or IPsec, is a framework of open standards developed by the Internet Engineering Task Force (IETF) that provide security for transmission of sensitive information over unprotected networks such as the Internet. IPsec acts at the network layer, protecting and authenticating IP packets between participating IPsec devices (peers), such as Cisco routers. IPsec provides the following optional network security services. In general, local security policy will dictate the use of one or more of these services:

- Data confidentiality—The IPsec sender can encrypt packets before sending them across a network.
- Data integrity—The IPsec receiver can authenticate packets sent by the IPsec sender to ensure that the data has not been altered during transmission.
- Data origin authentication—The IPsec receiver can authenticate the source of the IPsec packets sent. This service depends upon the data integrity service.
- Antireplay—The IPsec receiver can detect and reject replayed packets.

With IPsec, data can be sent across a public network without observation, modification, or spoofing. IPsec functionality is similar in both IPv6 and IPv4; however, site-to-site tunnel mode only is supported in IPv6.

In IPv6, IPsec is implemented using the AH authentication header and the ESP extension header. The authentication header provides integrity and authentication of the source. It also provides optional protection against replayed packets. The authentication header protects the integrity of most of the IP header fields and authenticates the source through a signature-based algorithm. The ESP header provides confidentiality, authentication of the source, connectionless integrity of the inner packet, antireplay, and limited traffic flow confidentiality.

The Internet Key Exchange (IKE) protocol is a key management protocol standard that is used in conjunction with IPsec. IPsec can be configured without IKE, but IKE enhances IPsec by providing additional features, flexibility, and ease of configuration for the IPsec standard.

IKE is a hybrid protocol that implements the Oakley key exchange and Skeme key exchange inside the Internet Security Association Key Management Protocol (ISAKMP) framework (ISAKMP, Oakley, and Skeme are security protocols implemented by IKE) (see the figure below). This functionality is similar to the security gateway model using IPv4 IPsec protection.

IPv6 IPsec Site-to-Site Protection Using Virtual Tunnel Interface

The IPsec virtual tunnel interface (VTI) provides site-to-site IPv6 crypto protection of IPv6 traffic. Native IPv6 IPsec encapsulation is used to protect all types of IPv6 unicast and multicast traffic.

The IPsec VTI allows IPv6 routers to work as security gateways, establish IPsec tunnels between other security gateway routers, and provide crypto IPsec protection for traffic from internal networks when it is sent across
the public IPv6 Internet (see the figure below). This functionality is similar to the security gateway model using IPv4 IPsec protection.

Figure 6: IPsec Tunnel Interface for IPv6

When the IPsec tunnel is configured, IKE and IPsec security associations (SAs) are negotiated and set up before the line protocol for the tunnel interface is changed to the UP state. The remote IKE peer is the same as the tunnel destination address; the local IKE peer will be the address picked from tunnel source interface which has the same IPv6 address scope as tunnel destination address.

The following figures shows the IPsec packet format.

Figure 7: IPv6 IPsec Packet Format

<table>
<thead>
<tr>
<th>Outer IPv6 header</th>
<th>AH header</th>
<th>ESP header</th>
<th>Inner IPv6 header</th>
<th>Payload</th>
<th>Padding</th>
<th>ESP Auth</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>ESP encrypted</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>ESP HMAC authenticated</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>AH authenticated (except for mutable fields)</td>
</tr>
</tbody>
</table>

How to Configure IPv6 Virtual Tunnel Interface

Configuring a VTI for Site-to-Site IPv6 IPsec Protection

Defining an IKE Policy and a Preshared Key in IPv6

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).

**Note**

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.

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, and when the remote peer’s policy specifies a lifetime that is less than or equal to the lifetime in the policy being compared. (If the lifetimes are not identical, the shorter lifetime—from the remote peer’s policy—will be used.)

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. 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.

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 hostname or its IPv6 address, depending on how you have set the ISAKMP identity of the router.

By default, a peer’s ISAKMP identity is the IPv6 address of the peer. If appropriate, you could change the identity to be the peer’s hostname instead. As a general rule, set the identities of all peers the same way—either all peers should use their IPv6 addresses or all peers should use their hostnames. If some peers use their hostnames and some peers use their IPv6 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.

Perform this task to create an IKE policy and a preshared key in IPv6.
SUMMARY STEPS

1. enable
2. configure terminal
3. crypto isakmp policy priority
4. authentication \{rsa-sig | rsa-encr | pre-share\}
5. hash \{sha | md5\}
6. group \{1 | 2 | 5\}
7. encryption \{des | 3des | aes | aes 192 | aes 256\}
8. lifetime seconds
9. exit
10. crypto isakmp key password-type keystring keystring \{address peer-address|ipv6 \{ipv6-address | ipv6-prefix\} | hostname hostname\} \{no-xauth\}
11. crypto keyring keyring-name \{vrf fvrf-name\}
12. pre-shared-key \{address address [mask] | hostname hostname | ipv6 \{ipv6-address | ipv6-prefix\}\} key key

DETAILED STEPS

<table>
<thead>
<tr>
<th>Step</th>
<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 policy priority | Defines an IKE policy, and enters ISAKMP policy configuration mode.  
* Policy number 1 indicates the policy with the highest priority. The smaller the priority argument value, the higher the priority. |
| | Example: Router(config)# crypto isakmp policy 15 | |
| Step 4 | authentication \{rsa-sig | rsa-encr | pre-share\} | Specifies the authentication method within an IKE policy.  
* The rsa-sig and rsa-encr keywords are not supported in IPv6. |
<p>| | Example: Router(config-isakmp-policy)# authentication pre-share | |</p>
<table>
<thead>
<tr>
<th>Command or Action</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Step 5</strong> hash {sha</td>
<td>md5}</td>
</tr>
<tr>
<td>Example: Router(config-isakmp-policy)# hash md5</td>
<td></td>
</tr>
<tr>
<td><strong>Step 6</strong> group {1</td>
<td>2</td>
</tr>
<tr>
<td>Example: Router(config-isakmp-policy)# group 2</td>
<td></td>
</tr>
<tr>
<td><strong>Step 7</strong> encryption {des</td>
<td>3des</td>
</tr>
<tr>
<td>Example: Router(config-isakmp-policy)# encryption 3des</td>
<td></td>
</tr>
<tr>
<td><strong>Step 8</strong> lifetime seconds</td>
<td>Specifies the lifetime of an IKE SA.</td>
</tr>
<tr>
<td>Example: Router(config-isakmp-policy)# lifetime 43200</td>
<td>- Setting the IKE lifetime value is optional.</td>
</tr>
<tr>
<td><strong>Step 9</strong> exit</td>
<td>Exits ISAKMP policy configuration mode and enter global configuration mode.</td>
</tr>
<tr>
<td>Example: Router(config-isakmp-policy)# exit</td>
<td></td>
</tr>
<tr>
<td><strong>Step 10</strong> crypto isakmp key password-type keystring keystring / address peer-address ipv6 {ipv6-address / ipv6-prefix}</td>
<td>Configures a preshared authentication key.</td>
</tr>
<tr>
<td>hostname hostname [no-xauth]</td>
<td></td>
</tr>
<tr>
<td>Example: Router(config)# crypto isakmp key 0 my-preshare-key-0 address ipv6 3ffe:1001::2/128</td>
<td></td>
</tr>
<tr>
<td><strong>Step 11</strong> crypto keyring keyring-name [vrf fvrf-name]</td>
<td>Defines a crypto keyring to be used during IKE authentication and enters config-keyring mode.</td>
</tr>
<tr>
<td>Example: Router(config)# crypto keyring keyring1</td>
<td></td>
</tr>
<tr>
<td><strong>Step 12</strong> pre-shared-key {address address [mask]</td>
<td>hostname hostname</td>
</tr>
<tr>
<td>Example: Router (config-keyring)# pre-shared-key ipv6 3FFE:2002::A8BB:CCFF:FEU1:2C02/128</td>
<td></td>
</tr>
</tbody>
</table>
## Configuring ISAKMP Aggressive Mode

You likely do not need to configure aggressive mode in a site-to-site scenario. The default mode is typically used.

### SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **crypto isakmp peer** `{address {ipv4-address | ipv6 ipv6-address ipv6-prefix-length} | hostname fqdn-hostname}`
4. **set aggressive-mode client-endpoint** `{client-endpoint | ipv6 ipv6-address}`
5. **end**

### DETAILED STEPS

<table>
<thead>
<tr>
<th>Command or Action</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Step 1</strong></td>
<td>Enables privileged EXEC mode.</td>
</tr>
<tr>
<td>Example:</td>
<td>Router&gt; enable</td>
</tr>
<tr>
<td></td>
<td>• Enter your password if prompted.</td>
</tr>
<tr>
<td><strong>Step 2</strong></td>
<td>Enters global configuration mode.</td>
</tr>
<tr>
<td>Example:</td>
<td>Router# configure terminal</td>
</tr>
<tr>
<td><strong>Step 3</strong></td>
<td>Enables an IPsec peer for IKE querying for tunnel attributes.</td>
</tr>
<tr>
<td>Example:</td>
<td>Router(config)# crypto isakmp peer address ipv6 3FFE:2002::A8BB:CCFF:FE01:2C02/128</td>
</tr>
<tr>
<td><strong>Step 4</strong></td>
<td>Defines the remote peer’s IPv6 address, which will be used by aggressive mode negotiation. The remote peer’s address is usually the client side’s end-point address.</td>
</tr>
<tr>
<td>Example:</td>
<td>Router(config-isakmp-peer)# set aggressive mode client-endpoint ipv6 3FFE:2002::A8BB:CCFF:FE01:2C02/128</td>
</tr>
</tbody>
</table>
### Purpose

Command or Action | Purpose
--- | ---
**Step 5** | end

**Example:**

```
Router(config-isakmp-peer)# end
```

Exits crypto ISAKMP peer configuration mode and returns to privileged EXEC mode.

---

### Defining an IPsec Transform Set and IPsec Profile

Perform this task to define an IPsec transform set. A transform set is a combination of security protocols and algorithms that is acceptable to the IPsec routers.

**SUMMARY STEPS**

1. enable
2. configure terminal
3. crypto ipsec transform-set  `transform-set-name transform1 [transform2] [transform3] [transform4]
4. crypto ipsec profile  `name
5. set transform-set  `transform-set-name [transform-set-name2...transform-set-name6

**DETAILED STEPS**

<table>
<thead>
<tr>
<th>Command or Action</th>
<th>Purpose</th>
</tr>
</thead>
</table>
| **Step 1** | enable

**Example:**

```
Router> enable
```

- Enter your password if prompted.

| **Step 2** | configure terminal

**Example:**

```
Router# configure terminal
```

Enters global configuration mode.

| **Step 3** | crypto ipsec transform-set  `transform-set-name transform1 [transform2] [transform3] [transform4]

**Example:**

```
Router(config)# crypto ipsec transform-set myset0 ah-sha-hmac esp-3des
```

Defines a transform set, and places the router in crypto transform configuration mode.
### Step 4
**crypto ipsec profile name**  
**Example:**  
Router(config)# crypto ipsec profile profile0  

**Purpose:** Defines the IPsec parameters that are to be used for IPsec encryption between two IPsec routers.

### Step 5
**set transform-set transform-set-name [transform-set-name2...transform-set-name6]**  
**Example:**  
Router(config-crypto-transform)# set-transform-set myset0  

**Purpose:** Specifies which transform sets can be used with the crypto map entry.

---

### Defining an ISAKMP Profile in IPv6

**SUMMARY STEPS**

1. enable
2. configure terminal  
3. crypto isakmp profile profile-name [accounting aaalist]  
4. self-identity {address | address ipv6 | fqdn | user-fqdn user-fqdn}  
5. match identity {group group-name | address {address [mask] [fvrf] | ipv6 ipv6-address} | host host-name | host domain domain-name | user user-fqdn | user domain domain-name}  
6. end

**DETAILED STEPS**

<table>
<thead>
<tr>
<th>Command or Action</th>
<th>Purpose</th>
</tr>
</thead>
</table>
| **Step 1**  
enable  
Example:  
Router> enable | Enables privileged EXEC mode.  
• Enter your password if prompted. |
| **Step 2**  
configure terminal  
Example:  
Router# configure terminal | Enters global configuration mode. |
### Command or Action

#### Step 3
**crypto isakmp profile profile-name [accounting aaalist**

Example:
```
Router(config)# crypto isakmp profile profile1
```

Purpose: Defines an ISAKMP profile and audits IPsec user sessions.

#### Step 4
**self-identity {address | address ipv6 | fqdn | user-fqdn user-fqdn}**

Example:
```
Router(config-isakmp-profile)# self-identity address ipv6
```

Purpose: Defines the identity that the local IKE uses to identify itself to the remote peer.

#### Step 5
**match identity {group group-name | address {address [mask] [fvr|] ipv6 ipv6-address] | host host-name | host domain domain-name | user user-fqdn | user domain domain-name}**

Example:
```
Router(config-isakmp-profile)# match identity address ipv6 3FFE:2002::A8BB:CCFF:FE01:2C02/128
```

Purpose: Matches an identity from a remote peer in an ISAKMP profile.

#### Step 6
**end**

Example:
```
Router(config-isakmp-profile)# end
```

Purpose: Exits ISAKMP profile configuration mode and returns to privileged EXEC mode.

---

### Configuring IPv6 IPsec VTI

**Before You Begin**

Use the *ipv6 unicast-routing* command to enable IPv6 unicast routing.
SUMMARY STEPS

1. enable
2. configure terminal
3. ipv6 unicast-routing
4. interface tunnel tunnel-number
5. ipv6 address ipv6-address/prefix
6. ipv6 enable
7. tunnel source {ip-address | ipv6-address | interface-type interface-number}
8. tunnel destination {host-name | ip-address | ipv6-address}
9. tunnel mode {aurp | cayman | dvmrp | eon | gre | gre multipoint | gre ipv6 | ipip [decapsulate-any] | ipsec ipv4 | iptalk | ipv6 | ipsec ipv6 | mpls | nos | rbscp}
10. tunnel protection ipsec profile name [shared]
11. end

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: 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: Router# configure terminal</td>
<td></td>
</tr>
<tr>
<td><strong>Step 3</strong> ipv6 unicast-routing</td>
<td>Enables IPv6 unicast routing. You only need to enable IPv6 unicast routing once, not matter how many interface tunnels you want to configure.</td>
</tr>
<tr>
<td>Example: Router(config)# ipv6 unicast-routing</td>
<td></td>
</tr>
<tr>
<td><strong>Step 4</strong> interface tunnel tunnel-number</td>
<td>Specifies a tunnel interface and number, and enters interface configuration mode.</td>
</tr>
<tr>
<td>Example: Router(config)# interface tunnel 0</td>
<td></td>
</tr>
<tr>
<td><strong>Step 5</strong> ipv6 address ipv6-address/prefix</td>
<td>Provides an IPv6 address to this tunnel interface, so that IPv6 traffic can be routed to this tunnel.</td>
</tr>
<tr>
<td>Example: Router(config-if)# ipv6 address 3FFE:C000:0:7::/64 eui-64</td>
<td></td>
</tr>
<tr>
<td>Step</td>
<td>Command or Action</td>
</tr>
<tr>
<td>------</td>
<td>------------------</td>
</tr>
<tr>
<td>6</td>
<td>ipv6 enable</td>
</tr>
<tr>
<td></td>
<td>Example:</td>
</tr>
<tr>
<td></td>
<td>Router(config-if)# ipv6 enable</td>
</tr>
<tr>
<td>7</td>
<td>tunnel source</td>
</tr>
<tr>
<td></td>
<td>{ip-address</td>
</tr>
<tr>
<td></td>
<td>Example:</td>
</tr>
<tr>
<td></td>
<td>Router(config-if)# tunnel source ethernet0</td>
</tr>
<tr>
<td>8</td>
<td>tunnel destination</td>
</tr>
<tr>
<td></td>
<td>{host-name</td>
</tr>
<tr>
<td></td>
<td>Example:</td>
</tr>
<tr>
<td></td>
<td>Router(config-if)# tunnel destination 2001:DB8:1111:2222::1</td>
</tr>
<tr>
<td>9</td>
<td>tunnel mode</td>
</tr>
<tr>
<td></td>
<td>{aurp</td>
</tr>
<tr>
<td></td>
<td>Example:</td>
</tr>
<tr>
<td></td>
<td>Router(config-if)# tunnel mode ipsec ipv6</td>
</tr>
<tr>
<td>10</td>
<td>tunnel protection ipsec profile name [shared]</td>
</tr>
<tr>
<td></td>
<td>Example:</td>
</tr>
<tr>
<td></td>
<td>Router(config-if)# tunnel protection ipsec profile profile1</td>
</tr>
<tr>
<td>11</td>
<td>end</td>
</tr>
<tr>
<td></td>
<td>Example:</td>
</tr>
<tr>
<td></td>
<td>Router(config-if)# end</td>
</tr>
</tbody>
</table>
Verifying IPsec Tunnel Mode Configuration

SUMMARY STEPS

1. `show adjacency [summary [interface-type interface-number]] | [prefix] [interface interface-number] [connectionid id] [link {ipv4|ipv6|mpls}] [detail]`
2. `show crypto engine {accelerator | brief | configuration | connections [active | dh | dropped-packet | show] | qos}`
3. `show crypto ipsec sa [ipv6] [interface-type interface-number] [detailed]`
4. `show crypto isakmp peer [config | detail]`
5. `show crypto isakmp policy`
6. `show crypto isakmp profile [tag profilename | vrf vrfname]`
7. `show crypto map [interface interface] [tag map-name]`
8. `show crypto session [detail] | [local ip-address [port local-port]] | [remote ip-address [port remote-port]] [detail] | fvfr vrf-name | ivrf vrf-name`
9. `show crypto socket`
10. `show ipv6 access-list [access-list-name]`
11. `show ipv6 cef [ipv6-prefix | prefix-length] | [interface-type interface-number] [longer-prefixes | similar-prefixes | detail | internal | platform | epoch | source]`
12. `show interface type number stats`

DETAILED STEPS

<table>
<thead>
<tr>
<th>Step</th>
<th>Command or Action</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>Step 1</td>
<td>`show adjacency [summary [interface-type interface-number]]</td>
<td>[prefix] [interface interface-number] [connectionid id] [link {ipv4</td>
</tr>
<tr>
<td>Example:</td>
<td>Router# show adjacency detail</td>
<td></td>
</tr>
<tr>
<td>Step 2</td>
<td>`show crypto engine {accelerator</td>
<td>brief</td>
</tr>
<tr>
<td>Example:</td>
<td>Router# show crypto engine connection active</td>
<td></td>
</tr>
<tr>
<td>Step 3</td>
<td><code>show crypto ipsec sa [ipv6] [interface-type interface-number] [detailed]</code></td>
<td>Displays the settings used by current SAs in IPv6.</td>
</tr>
<tr>
<td>Example:</td>
<td>Router# show crypto ipsec sa ipv6</td>
<td></td>
</tr>
<tr>
<td>Command or Action</td>
<td>Purpose</td>
<td></td>
</tr>
<tr>
<td>-------------------------------------------------------------</td>
<td>-------------------------------------------------</td>
<td></td>
</tr>
<tr>
<td><strong>Step 4</strong> show crypto isakmp peer [config</td>
<td>detail]</td>
<td>Displays peer descriptions.</td>
</tr>
<tr>
<td>Example:</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Router# show crypto isakmp peer detail</td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>Step 5</strong> show crypto isakmp policy</td>
<td>Displays the parameters for each IKE policy.</td>
<td></td>
</tr>
<tr>
<td>Example:</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Router# show crypto isakmp policy</td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>Step 6</strong> show crypto isakmp profile [tag profilename</td>
<td>vrf vrfname]</td>
<td>Lists all the ISAKMP profiles that are defined on a router.</td>
</tr>
<tr>
<td>Example:</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Router# show crypto isakmp profile</td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>Step 7</strong> show crypto map [interface interface</td>
<td>tag map-name]</td>
<td>Displays the crypto map configuration. The crypto maps shown in this command output are dynamically generated. The user does not have to configure crypto maps.</td>
</tr>
<tr>
<td>Example:</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Router# show crypto map</td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>Step 8</strong> show crypto session [detail]</td>
<td>[local ip-address [port local-port]</td>
<td>[remote ip-address [port remote-port]]</td>
</tr>
<tr>
<td>Example:</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Router# show crypto session</td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>Step 9</strong> show crypto socket</td>
<td>Lists crypto sockets.</td>
<td></td>
</tr>
<tr>
<td>Example:</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Router# show crypto socket</td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>Step 10</strong> show ipv6 access-list [access-list-name]</td>
<td>Displays the contents of all current IPv6 access lists.</td>
<td></td>
</tr>
<tr>
<td>Example:</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Router# show ipv6 access-list</td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>Step 11</strong> show ipv6 cef [ipv6-prefix / prefix-length]</td>
<td>[interface-type interface-number]</td>
<td>[longer-prefixes</td>
</tr>
<tr>
<td>Example:</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Router# show ipv6 cef</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
**Step 12**

<table>
<thead>
<tr>
<th>Command or Action</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>show interface type number stats</code></td>
<td>Displays numbers of packets that were process switched, fast switched, and distributed switched.</td>
</tr>
</tbody>
</table>

**Example:**

```
Router# show interface fddi 3/0/0 stats
```

---

**Troubleshooting IPsec for IPv6 Configuration and Operation**

**SUMMARY STEPS**

1. `enable`
2. `debug crypto ipsec`
3. `debug crypto engine packet [detail]`

**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: <code>Router# enable</code></td>
<td></td>
</tr>
<tr>
<td><strong>Step 2</strong> <code>debug crypto ipsec</code></td>
<td>Displays IPsec network events.</td>
</tr>
<tr>
<td>Example: <code>Router# debug crypto ipsec</code></td>
<td></td>
</tr>
<tr>
<td><strong>Step 3</strong> <code>debug crypto engine packet [detail]</code></td>
<td>Displays the contents of IPv6 packets.</td>
</tr>
<tr>
<td>Example: <code>Router# debug crypto engine packet</code></td>
<td><strong>Caution</strong> Using this command could flood the system and increase CPU usage if several packets are being encrypted.</td>
</tr>
</tbody>
</table>
Configuration Examples for IPv6 Virtual Tunnel Interface

Example: Configuring a VTI for Site-to-Site IPv6 IPsec Protection

```plaintext
crypto isakmp policy 1
  encryption aes
  authentication pre-share
  group 14
!
crypto isakmp key myPreshareKey0 address ipv6 3FFE:2002::A8BB:CCFF:FE01:2C02/128
!
crypto isakmp keepalive 30 30
!
crypto ipsec transform-set Trans1 ah-sha-hmac esp-aes
!
crypto ipsec profile profile0
  set transform-set Trans1
!
ipv6 cef
!
tunnel source Ethernet2/0
  tunnel destination 3FFE:2002::A8BB:CCFF:FE01:2C02
  tunnel mode ipsec ipv6
  tunnel protection ipsec profile profile0
```

Additional References

<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>Security commands</td>
<td>Cisco IOS Security Command Reference</td>
</tr>
<tr>
<td>QoS Commands</td>
<td>Cisco IOS Quality of Service Solutions Command Reference</td>
</tr>
<tr>
<td>Weighted Fair Queueing</td>
<td>Configuring Weighted Fair Queueing feature module.</td>
</tr>
</tbody>
</table>
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 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

<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 IPv6 Virtual Tunnel Interface

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 10: Feature Information for IPv6 Virtual Tunnel Interface

<table>
<thead>
<tr>
<th>Feature Name</th>
<th>Releases</th>
<th>Feature Information</th>
</tr>
</thead>
<tbody>
<tr>
<td>IPv6 Virtual Tunnel Interface</td>
<td>Cisco IOS XE Release 2.4</td>
<td>IPsec is a framework of open standards that provide security for transmission of sensitive information over unprotected networks such as the Internet. IPsec acts at the network layer, protecting and authenticating IP packets between participating IPsec devices (peers), such as Cisco routers. The following commands were introduced or modified: authentication (IKE policy), crypto ipsec profile, crypto isakmp key, crypto isakmp peer, crypto isakmp policy, crypto isakmp profile, crypto keyring, debug crypto ipv6 ipsec, encryption (IKE policy), group (IKE policy), hash (IKE policy), lifetime (IKE policy), match identity, pre-shared-key, self-identity, set aggressive-mode client-endpoint, set transform-set, show adjacency, show crypto engine, show crypto ipsec sa, show crypto isakmp peers, show crypto isakmp policy, show crypto isakmp profile, show crypto map, show crypto session, show crypto socket, show ipv6 access-list, show ipv6 cef, tunnel destination, tunnel mode, tunnel source.</td>
</tr>
</tbody>
</table>
## Index

### D

DF Bit Override Functionality with IPsec Tunnels 59, 60, 62
  - Additional references 62
  - Prerequisites 59
  - Restrictions 60

### I

Invalid security parameter index recovery 17, 18, 19, 29
  - Additional references 29
  - Prerequisites 17
  - Restrictions 18
  - Verifying 19

IP multicast routing 95
  - MDS 95
    - Packet statistics, displaying 95
IPSec 84
IPsec Anti-Replay Window 10
  - Expanding and Disabling 10
  - Expanding and Disabling 7
    - Configuration examples 7
IPSec dead peer detection periodic message option 34, 44
  - Additional references 44
  - Prerequisites 34
  - Restrictions 34