



Invalid Security Parameter Index Recovery

First Published: July 25, 2003

Last Updated: May 5, 2011

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

Your software release may not support all the features documented in this module. For the latest feature information and caveats, see the release notes for your platform and software release. To find information about the features documented in this module, and to see a list of the releases in which each feature is supported, see the [“Feature Information for Invalid Security Parameter Index Recovery”](#) section on page 17.

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

Contents

- [Prerequisites for Invalid Security Parameter Index Recovery, page 2](#)
- [Restrictions for Invalid Security Parameter Index Recovery, page 2](#)
- [Information About Invalid Security Parameter Index Recovery, page 2](#)
- [How to Configure Invalid Security Parameter Index Recovery, page 3](#)
- [Configuration Examples for Invalid Security Parameter Index Recovery, page 10](#)
- [Additional References, page 16](#)
- [Feature Information for Invalid Security Parameter Index Recovery, page 17](#)



Americas Headquarters:

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

Prerequisites for Invalid Security Parameter Index Recovery

Before configuring the Invalid Security Parameter Index Recovery feature, you must have enabled Internet Key Exchange (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

To use the Invalid Security Parameter Index Recovery feature, you should understand the following concept.

- [How the Invalid Security Parameter Index Recovery Feature Works, page 2](#)

How the Invalid Security Parameter Index Recovery 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.

**Note**

A single security association (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

This section contains the following procedure.

- [Configuring Invalid Security Parameter Index Recovery, page 3](#)

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

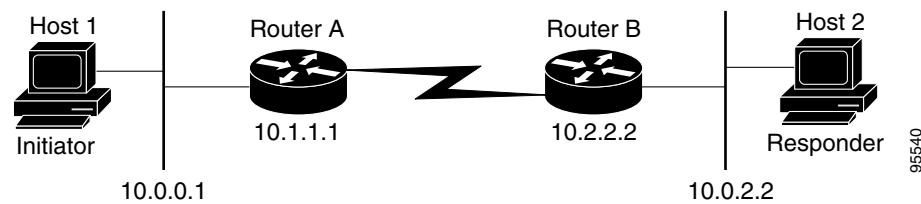
	Command or Action	Purpose
Step 1	enable Example: Router> enable	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	configure terminal Example: Router# configure terminal	Enters global configuration mode.
Step 3	crypto isakmp invalid-spi-recovery Example: Router (config)# crypto isakmp invalid-spi-recovery	Initiates the IKE module process whereby the IKE module notifies the receiving peer that an “Invalid SPI” error has occurred.

Verifying the Invalid Security Parameter Index Recovery 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.

Figure 1 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

To verify the preshared configuration, perform the following 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 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

```
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

```
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

```
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
#send errors 0, #recv errors 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:

outbound esp 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): (4537835/3595)
    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:

```

Router B

```

Router# show crypto ipsec sa interface ethernet1/0

interface: Ethernet1/0
  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 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

#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:

outbound esp sas:
  spi: 0x249C5062(614223970)
    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.

```
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: Ethernet1/0
  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: 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
  #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: 0

inbound esp sas:

inbound ah sas:

inbound pcp sas:

outbound esp sas:

outbound ah sas:

outbound pcp sas:

```

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

```

ping
Protocol [ip]: ip
Target IP address: 10.0.2.2
Repeat count [5]: 30
Datagram size [100]: 100
Timeout in seconds [2]:
Extended commands [n]: no
Sweep range of sizes [n]: n
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

  f_vrf/i_vrf  dst          src          state          conn-id slot
    /          10.1.1.1    10.2.2.2    QM_IDLE          3      0
    /          10.1.1.1    10.2.2.2    MM_NO_STATE          1      0 (deleted)

RouterB# show crypto ipsec sa

interface: Ethernet1/0
  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: 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: D763771F

inbound esp sas:
spi: 0xE7AB4256(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(4179647725)
  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
```

f_vrf/i_vrf	dst	src	state	conn-id	slot	
/	10.2.2.2	10.1.1.1	MM_NO_STATE	1	0	(deleted)


```

/          10.2.2.2      10.1.1.1      QM_IDLE          2          0

```

Step 4 Check for an invalid SPI message on Router B.

```
Router# show logging
```

```

Syslog logging: enabled (10 messages dropped, 13 messages rate-limited, 0 flushes, 0
overruns, xml disabled)
  Console logging: disabled
  Monitor logging: level debugging, 0 messages logged, xml disabled
  Buffer logging: level debugging, 43 messages logged, xml disabled
  Logging Exception size (8192 bytes)
  Count and timestamp logging messages: disabled
  Trap logging: level informational, 72 message lines logged

Log Buffer (8000 bytes):

*Mar 24 20:55:45.739: %CRYPTO-4-RECV_PKT_INV_SPI: decaps: rec'd IPSEC packet has invalid
spi for
      destaddr=10.2.2.2, prot=51, spi=0x1214F0D(18960141), srcaddr=10.1.1.1
*Mar 24 20:55:47.743: IPSEC(validate_proposal_request): proposal part #1,
(key eng. msg.) INBOUND local= 10.2.2.2, remote= 10.1.1.1,
  local_proxy= 10.0.2.2/255.255.255.255/0/0 (type=1),
  remote_proxy= 10.0.0.1/255.255.255.255/0/0 (type=1),
  protocol= AH, transform= ah-sha-hmac ,
  lifedur= 0s and 0kb,
  spi= 0x0(0), conn_id= 0, keysize= 0, flags= 0x2
*Mar 24 20:55:47.743: IPSEC(validate_proposal_request): proposal part #2,
(key eng. msg.) INBOUND local= 10.2.2.2, remote= 10.1.1.1,
  local_proxy= 10.0.2.2/255.255.255.255/0/0 (type=1),
  remote_proxy= 10.0.0.1/255.255.255.255/0/0 (type=1),
  protocol= ESP, transform= esp-des esp-sha-hmac ,
  lifedur= 0s and 0kb,
  spi= 0x0(0), conn_id= 0, keysize= 0, flags= 0x2
*Mar 24 20:55:47.743: IPSEC(kei_proxy): head = testtag1, map->ivrf = , kei->ivrf =
*Mar 24 20:55:47.743: IPSEC(key_engine): got a queue event with 2 kei messages
*Mar 24 20:55:47.743: IPSEC(spi_response): getting spi 4179647725 for SA
      from 10.2.2.2      to 10.1.1.1      for prot 2
*Mar 24 20:55:47.747: IPSEC(spi_response): getting spi 3886760534 for SA
      from 10.2.2.2      to 10.1.1.1      for prot 3
*Mar 24 20:55:48.071: IPsec: Flow_switching Allocated flow for flow_id 939524099
*Mar 24 20:55:48.071: IPsec: Flow_switching Allocated flow for flow_id 939524100
*Mar 24 20:55:48.135: IPSEC(key_engine): got a queue event with 4 kei messages
*Mar 24 20:55:48.135: IPSEC(initialize_sas): ,
(key eng. msg.) INBOUND local= 10.2.2.2, remote= 10.1.1.1,
  local_proxy= 10.0.2.2/0.0.0.0/0/0 (type=1),
  remote_proxy= 10.0.0.1/0.0.0.0/0/0 (type=1),
  protocol= AH, transform= ah-sha-hmac ,
  lifedur= 3600s and 4608000kb,
  spi= 0xF9205CED(4179647725), conn_id= 939529221, keysize= 0, flags= 0x2
*Mar 24 20:55:48.135: IPSEC(initialize_sas): ,
(key eng. msg.) OUTBOUND local= 10.2.2.2, remote= 10.1.1.1,
  local_proxy= 10.0.2.2/0.0.0.0/0/0 (type=1),
  remote_proxy= 10.0.0.1/0.0.0.0/0/0 (type=1),
  protocol= AH, transform= ah-sha-hmac ,
  lifedur= 3600s and 4608000kb,
  spi= 0xEB95406F(3952427119), conn_id= 939529222, keysize= 0, flags= 0xA
*Mar 24 20:55:48.135: IPSEC(initialize_sas): ,
(key eng. msg.) INBOUND local= 10.2.2.2, remote= 10.1.1.1,
  local_proxy= 10.0.2.2/0.0.0.0/0/0 (type=1),
  remote_proxy= 10.0.0.1/0.0.0.0/0/0 (type=1),
  protocol= ESP, transform= esp-des esp-sha-hmac ,

```

```

    lifedur= 3600s and 4608000kb,
    spi= 0xE7AB4256(3886760534), conn_id= 939529223, keysize= 0, flags= 0x2
*Mar 24 20:55:48.135: IPSEC(initialize_sas): ,
(key eng. msg.) OUTBOUND local= 10.2.2.2, remote= 10.1.1.1,
    local_proxy= 10.0.2.2/0.0.0.0/0/0 (type=1),
    remote_proxy= 10.0.0.1/0.0.0.0/0/0 (type=1),
    protocol= ESP, transform= esp-des esp-sha-hmac ,

    lifedur= 3600s and 4608000kb,
    spi= 0xD763771F(3613619999), conn_id= 939529224, keysize= 0, flags= 0xA
*Mar 24 20:55:48.139: IPSEC(kei_proxy): head = testtag1, map->ivrf = , kei->ivrf =
*Mar 24 20:55:48.139: IPSEC(mtrees_add_ident): src 10.2.2.2, dest 10.1.1.1, dest_port 0

*Mar 24 20:55:48.139: IPSEC(create_sa): sa created,
(sa) sa_dest= 10.1.1.1, sa_prot= 51,
    sa_spi= 0xF9205CED(4179647725),
    sa_trans= ah-sha-hmac , sa_conn_id= 939529221
*Mar 24 20:55:48.139: IPSEC(create_sa): sa created,
(sa) sa_dest= 10.2.2.2, sa_prot= 51,
    sa_spi= 0xEB95406F(3952427119),
    sa_trans= ah-sha-hmac , sa_conn_id= 939529222
*Mar 24 20:55:48.139: IPSEC(create_sa): sa created,
(sa) sa_dest= 10.1.1.1, sa_prot= 50,
    sa_spi= 0xE7AB4256(3886760534),
    sa_trans= esp-des esp-sha-hmac , sa_conn_id= 939529223
*Mar 24 20:55:48.139: IPSEC(create_sa): sa created,
(sa) sa_dest= 10.2.2.2, sa_prot= 50,
    sa_spi= 0xD763771F(3613619999),
    sa_trans= esp-des esp-sha-hmac , sa_conn_id= 939529224
ipseca-72a#

```

Configuration Examples for Invalid Security Parameter Index Recovery

This section provides the following configuration example.

- [Invalid Security Parameter Index Recovery: Example, page 10](#)

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. [Figure 1](#) shows the topology used for this example.

Router A

```
Router# show running-config
```

```
Building configuration...
```

```
Current configuration : 2048 bytes
```

```
!
version 12.3
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$4GZB$L2Y0mnenOCNAu0jgFxebT/
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 12.3
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 $1$kKqL$5Th5Qhw1ubDkkK90KWFxi1
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 mroute-cache
 shutdown
 duplex half
!
interface Ethernet1/0
 ip address 10.2.2.2 255.0.0.0
```

```
no ip route-cache cef
duplex half
crypto map testtag1
!
interface Ethernet1/1
ip address 10.0.2.2 255.0.0.0
no ip route-cache cef
duplex half
!
interface Ethernet1/2
no ip address

no ip route-cache
no ip mroute-cache
shutdown
duplex half
!
interface Ethernet1/3
no ip address
no ip route-cache
no ip mroute-cache
shutdown
duplex half
!
interface Ethernet1/4
no ip address
no ip route-cache
no ip mroute-cache
shutdown
duplex half
!
interface Ethernet1/5
no ip address
no ip route-cache
no ip mroute-cache
shutdown
duplex half
!
interface Ethernet1/6
no ip address
no ip route-cache
no ip mroute-cache
shutdown
duplex half
!
interface Ethernet1/7
no ip address
no ip route-cache
no ip mroute-cache
shutdown
duplex half
!
interface Serial3/0
no ip address
no ip route-cache
no ip mroute-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 Serial13/2
    no ip address
    no ip route-cache
    no ip mroute-cache
    shutdown
    serial restart_delay 0
    !
interface Serial13/3
    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

Related Documents

Related Topic	Document Title
Configuring IKE	<i>Configuring Internet Key Exchange for IPsec VPNs</i>
Interface commands	<i>Cisco IOS Master Command List</i>

MIBs

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

Technical Assistance

Description	Link
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.	http://www.cisco.com/cisco/web/support/index.html

Feature Information for Invalid Security Parameter Index Recovery

Table 1 lists the release history for this feature.

Use Cisco Feature Navigator to find information about platform support and software image support. Cisco Feature Navigator enables you to determine which software images support a specific software release, feature set, or platform. To access Cisco Feature Navigator, go to <http://www.cisco.com/go/cfn>. An account on Cisco.com is not required.



Note

Table 1 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.

Table 1 Feature Information for Invalid Security Parameter Index Recovery

Feature Name	Releases	Feature Information
Invalid Security Parameter Index Recovery	12.3(2)T 12.2(18)SXE	<p>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.</p> <p>This feature was introduced in Cisco IOS Release 12.3(2)T.</p> <p>This feature was integrated into Cisco IOS Release 12.2(18)SXE.</p> <p>The following command was introduced or modified: crypto isakmp invalid-spi-recovery</p>

Cisco and the Cisco Logo are trademarks of Cisco Systems, Inc. and/or its affiliates in the U.S. and other countries. A listing of Cisco's trademarks can be found at www.cisco.com/go/trademarks. Third party trademarks mentioned are the property of their respective owners. The use of the word partner does not imply a partnership relationship between Cisco and any other company. (1005R)

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

© 2007–2011 Cisco Systems, Inc. All rights reserved.

