IPsec and IKE MIB Support for Cisco VRF-Aware IPsec

The IPsec and IKE MIB Support for the Virtual Private Network routing and forwarding- (VRF-) aware IP security (IPsec) feature allows VRF-aware IPsec to be managed with MIBs, which provide the details of IPsec statistics and performance metrics on a per VRF basis.

- Finding Feature Information, page 1
- Prerequisites for IPsec and IKE MIB Support for Cisco VRF-Aware IPsec, page 1
- Information About IPsec and IKE MIB Support for Cisco VRF-Aware IPsec, page 2
- How to Configure IPsec and IKE MIB Support for Cisco VRF-Aware IPsec, page 3
- Configuration Example for IPsec and IKE MIB Support for Cisco VRF-Aware IPsec, page 4
- Additional References, page 14
- Feature Information for IPsec and IKE MIB Support for Cisco VRF-Aware IPsec, page 16

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 and IKE MIB Support for Cisco VRF-Aware IPsec

- You should be familiar with configuring Simple Network Management Protocol (SNMP).
Information About IPsec and IKE MIB Support for Cisco VRF-Aware IPsec

MIBs Supported by the IPsec and IKE MIB Support for Cisco VRF-Aware IPsec Feature

- CISCO-IPSEC-FLOW-MONITOR-MIB supports IKE and IPSEC per-tunnel history and failure information. The length of this history and failure information can be configured and must be maintained on a per-VRF basis. The table sizes are controlled by using the `crypto mib ipsec flowmib history tunnel size number` and `crypto mib ipsec flowmib history failure size` commands in global configuration mode.
- CISCO-IPSEC-MIB
- CISCO-IPSEC-POLICY-MAP-MIB is supported. However, because this MIB applies to the entire router rather than to a specific VPN VRF instance, it is not VRF aware; therefore, polling of the object identifiers (OIDs) that belong to this MIB is accomplished with respect to the global VRF context.

SNMP Traps Supported by the IPsec and IKE MIB Support for Cisco VRF-Aware IPsec Feature

The following IKE and IPsec tunnel start and stop traps must go with their corresponding VRF:

- IPSEC_TUNNEL_STOP
- IKE_TUNNEL_STOP
- IPSEC_TUNNEL_START
- IKE_TUNNEL_START

The following traps are global traps that have been modified for the Cisco VRF-Aware IPsec feature:

- TOO_MANY_SAS_CREATED
- CRYPTOMAP_ADDED
- CRYPTOMAPSET.Attached
- CRYPTOMAP_DELETED
- CRYPTOMAPSET_DELETED
- ISAKMP_POLICY_ADDED
- ISAKMP_POLICY_DELETED
How to Configure IPsec and IKE MIB Support for Cisco VRF-Aware IPsec

No special configuration is needed for this feature. The SNMP framework can be used to manage VRF-aware IPsec using MIBs. See the Configuration Examples for IPsec and IKE MIB Support for Cisco VRF-Aware IPsec section for more information.

The following section provides information about troubleshooting this feature:

How to Troubleshoot the IPsec and IKE MIB Support for Cisco VRF-Aware IPsec Feature

The following `debug crypto mib` command and keywords may be used to display information about the IPsec and Internet Key Exchange (IKE) MIB as it relates to Cisco VRF-aware IPsec.

SUMMARY STEPS

1. enable
2. debug crypto mib detail
3. debug crypto mib error

DETAILED STEPS

<table>
<thead>
<tr>
<th>Command or Action</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Step 1</strong> enable</td>
<td>Enables privileged EXEC mode.</td>
</tr>
<tr>
<td><strong>Example:</strong></td>
<td></td>
</tr>
<tr>
<td>Router&gt; enable</td>
<td></td>
</tr>
<tr>
<td><strong>Step 2</strong> debug crypto mib detail</td>
<td>Displays different events as they occur in the IPsec MIB subsystem.</td>
</tr>
<tr>
<td><strong>Example:</strong></td>
<td></td>
</tr>
<tr>
<td>Router# debug crypto mib detail</td>
<td></td>
</tr>
<tr>
<td><strong>Step 3</strong> debug crypto mib error</td>
<td>Displays error events in the MIB agent.</td>
</tr>
<tr>
<td><strong>Example:</strong></td>
<td></td>
</tr>
<tr>
<td>Router# debug crypto mib error</td>
<td></td>
</tr>
</tbody>
</table>
Configuration Example for IPsec and IKE MIB Support for Cisco VRF-Aware IPsec

Configuration That Has Two VRFs Examples

The following output example is for a typical hub configuration that has two VRFs. The output is what you would see if you were to poll for the IPsec security association (SA). Router 3745b is the VRF-aware router.

Two VRFs Configured

The following output shows that two VRFs have been configured (vrf1 and vrf2).

Router3745b# show running-config
Building configuration...
Current configuration : 6567 bytes
!
version 12.4
service timestamps debug datetime msec localtime
service timestamps log uptime
no service password-encryption
!
hostname ipsecf-3745b
!
boot-start-marker
boot-end-marker
!
no logging console
enable password lab
!
no aaa new-model
!
resource policy
!
memory-size iomem 5
! clock timezone PST -8
clock summer-time PDT recurring
ip subnet-zero
ip cef
!
!
ip vrf vrf1
rd 1:101
context vrf-vrf1-context
route-target export 1:101
route-target import 1:101
!
ip vrf vrf2
rd 2:101
context vrf-vrf2-context
route-target export 2:101
route-target import 2:101
!
no ip domain lookup
!
crypto keyring vrf1-1 vrf vrf1
pre-shared-key address 10.1.1.1 255.255.255.0 key vrf1-1
crypto keyring vrf2-1 vrf vrf2
pre-shared-key address 10.1.2.1 255.255.255.0 key vrf2-1
!
crypto isakmp policy 1
authentication pre-share
!
crypto isakmp policy 50
  authentication pre-share
crypto isakmp key global1-1 address 10.1.151.1
crypto isakmp key global2-1 address 10.1.152.1
crypto isakmp profile vrf1-1
  keyring vrf1-1
  match identity address 10.1.1.1 255.255.255.255 vrf1
  keyring vrf2-1
  match identity address 10.1.2.1 255.255.255.255 vrf2
!
crypto ipsec security-association lifetime kilobytes 99000
crypto ipsec security-association lifetime seconds 5000
!
crypto ipsec transform-set tset ah-sha-hmac esp-des esp-sha-hmac
!
crypto map global1-1 10 ipsec-isakmp
  set peer 10.1.151.1
  set transform-set tset
  match address 151
!
crypto map global2-1 10 ipsec-isakmp
  set peer 10.1.152.1
  set transform-set tset
  match address 152
!
crypto map vrf1-1 10 ipsec-isakmp
  set peer 10.1.1.1
  set transform-set tset
  set isakmp-profile vrf1-1
  match address 101
!
crypto map vrf2-1 10 ipsec-isakmp
  set peer 10.1.2.1
  set transform-set tset
  set isakmp-profile vrf2-1
  match address 102
!
!
interface FastEthernet0/0
  ip address 10.1.38.25 255.255.255.0
  no ip mroute-cache
duplex auto
speed auto
!
interface Serial0/0
  no ip address
shutdown
clock rate 2000000
!
interface FastEthernet0/1
  no ip address
no ip mroute-cache
shutdown
duplex auto
speed auto
!
interface Serial0/1
  no ip address
shutdown
clock rate 2000000
!
interface Serial1/0
  no ip address
encapsulation frame-relay
ox ip route-cache cef
no ip route-cache
ox ip mroute-cache
no keepalive
serial restart-delay 0
clock rate 128000
no frame-relay inverse-arp
!
interface Serial1/0.1 point-to-point
  ip vrf forwarding vrf1
  ip address 10.3.1.1 255.255.255.0
  no ip route-cache
  frame-relay interface-dlci 21
!
interface Serial1/0.2 point-to-point
  ip vrf forwarding vrf2
  ip address 10.3.2.1 255.255.255.0
  no ip route-cache
  frame-relay interface-dlci 22
!
interface Serial1/0.151 point-to-point
  ip address 10.7.151.1 255.255.255.0
  no ip route-cache
  frame-relay interface-dlci 151
!
interface Serial1/0.152 point-to-point
  ip address 10.7.152.1 255.255.255.0
  no ip route-cache
  frame-relay interface-dlci 152
!
interface Serial1/1
  no ip address
  no ip mroute-cache
  shutdown
  serial restart-delay 0
!
interface Serial1/2
  no ip address
  encapsulation frame-relay
  no ip route-cache cef
  no ip route-cache
  no ip mroute-cache
  no keepalive
  serial restart-delay 0
  no frame-relay inverse-arp
!
interface Serial1/2.1 point-to-point
  ip vrf forwarding vrf1
  ip address 10.1.1.2 255.255.255.0
  no ip route-cache
  frame-relay interface-dlci 21
  crypto map vrf1-1
!
interface Serial1/2.2 point-to-point
  ip vrf forwarding vrf2
  ip address 10.1.2.2 255.255.255.0
  no ip route-cache
  frame-relay interface-dlci 22
  crypto map vrf2-1
!
interface Serial1/2.151 point-to-point
  ip address 10.5.151.2 255.255.255.0
  no ip route-cache
  frame-relay interface-dlci 151
  crypto map global1-1
!
interface Serial1/2.152 point-to-point
  ip address 10.5.152.2 255.255.255.0
  no ip route-cache
  frame-relay interface-dlci 152
  crypto map global2-1
!
interface Serial1/3
  no ip address
  no ip mroute-cache
  shutdown
  serial restart-delay 0
!
ip default-gateway 10.1.38.1
Both VRFs Cleared

The following output, for abc1 and abc2, shows that both VRFs have been "cleared" to ensure that all the counters are initialized to a known value.

Both VRFs Cleared

The following output, for abc1 and abc2, shows that both VRFs have been "cleared" to ensure that all the counters are initialized to a known value.
The following output shows that VRF abc1 has been cleared:

orcas:2> setenv SR_MGR_CONF /users/green1
orcas:3> setenv SR_UTIL_SNMP_VERSION v2c
orcas:5> setenv SR_UTIL_COMMUNITY abc1
orcas:6> setenv SR_MGR_CONF_DIR /users/green1
orcas:7> /auto/sw/packages/snmpr/10.14.2.0/solaris2bin/getmany -v2c 10.1.38.25
cipSecMIBObjects
cipSecMibLevel.0 = 1
cikeGlobalActiveTunnels.0 = 0
cikeGlobalPreviousTunnels.0 = 0
cikeGlobalInOctets.0 = 0
cikeGlobalInPkts.0 = 0
cikeGlobalInDropPkts.0 = 0
cikeGlobalInNotifySys.0 = 0
cikeGlobalInP2Exchs.0 = 0
cikeGlobalInP2ExchgInvalids.0 = 0
cikeGlobalInP2ExchgRejects.0 = 0
cikeGlobalInP2SaRequests.0 = 0
cikeGlobalOutOctets.0 = 0
cikeGlobalOutPkts.0 = 0
cikeGlobalOutDropPkts.0 = 0
cikeGlobalOutNotifySys.0 = 0
cikeGlobalOutP2Exchs.0 = 0
cikeGlobalOutP2ExchgInvalids.0 = 0
cikeGlobalOutP2ExchgRejects.0 = 0
cikeGlobalOutP2SaRequests.0 = 0
cikeGlobalInitTunnels.0 = 0
cikeGlobalInitTunnelFails.0 = 0
cikeGlobalRespTunnelFails.0 = 0
cikeGlobalSysCapFails.0 = 0
cikeGlobalAuthFails.0 = 0
cikeGlobalDecryptFails.0 = 0
cikeGlobalHashValidFails.0 = 0
cikeGlobalNoSaFails.0 = 0
cipSecGlobalActiveTunnels.0 = 0
cipSecGlobalPreviousTunnels.0 = 0
cipSecGlobalInOctets.0 = 0
cipSecGlobalHcInOctets.0 = 0x00
cipSecGlobalInOctWraps.0 = 0
cipSecGlobalInDecompOctets.0 = 0
cipSecGlobalHcInDecompOctets.0 = 0x00
cipSecGlobalInDecompOctWraps.0 = 0
cipSecGlobalInPkts.0 = 0
cipSecGlobalInDrops.0 = 0
cipSecGlobalInReplayDrops.0 = 0
cipSecGlobalInAuths.0 = 0
cipSecGlobalInAuthFails.0 = 0
cipSecGlobalInDecrypts.0 = 0
cipSecGlobalInDecryptFails.0 = 0
cipSecGlobalOutOctets.0 = 0
cipSecGlobalHcOutOctets.0 = 0x00
cipSecGlobalOutOctWraps.0 = 0
cipSecGlobalOutUncompOctets.0 = 0
cipSecGlobalHcOutUncompOctets.0 = 0x00
cipSecGlobalOutUncompOctWraps.0 = 0
cipSecGlobalOutPkts.0 = 0
cipSecGlobalOutDrops.0 = 0
cipSecGlobalOutAuths.0 = 0
cipSecGlobalOutAuthFails.0 = 0
cipSecGlobalOutEncrypts.0 = 0
cipSecGlobalOutEncryptFails.0 = 0
cipSecGlobalProtocolUseFails.0 = 0
cipSecGlobalNoSaFails.0 = 0
cipSecGlobalSysCapFails.0 = 0
cipSecHistTableSize.0 = 200
cipSecHistCheckPoint.0 = ready(1)
cipSecFailTableSize.0 = ready(1)
cipSecTrapCntlIkeTunnelStart.0 = enabled(1)
cipSecTrapCntlIkeTunnelStop.0 = enabled(1)
cipSecTrapCntlIkeSysFailure.0 = disabled(2)
cipSecTrapCntlIkeCertCrlFailure.0 = disabled(2)
The following output shows that VRF abc2 has been cleared:

```
orcas:8> setenv SR_UTIL_COMMUNITY abc2
orcas:9> /auto/sw/packages/snmpr/14.2.0.0/solaris2bin/getmany-v2c 10.1.38.25 cipSecMIBObjects
cipSecMibLevel.0 = 1
cikeGlobalActiveTunnels.0 = 0
cikeGlobalPreviousTunnels.0 = 0
cikeGlobalInOctets.0 = 0
cikeGlobalInPkts.0 = 0
cikeGlobalInDropPkts.0 = 0
cikeGlobalInNotify.0 = 0
cikeGlobalInP2Exchgs.0 = 0
cikeGlobalInP2ExchgInvalids.0 = 0
cikeGlobalInP2ExchgRejects.0 = 0
cikeGlobalInP2SaDeleteRequests.0 = 0
cikeGlobalOutOctets.0 = 0
cikeGlobalOutPkts.0 = 0
cikeGlobalOutDropPkts.0 = 0
cikeGlobalOutNotify.0 = 0
cikeGlobalOutP2Exchgs.0 = 0
cikeGlobalOutP2ExchgInvalids.0 = 0
cikeGlobalOutP2ExchgRejects.0 = 0
cikeGlobalOutP2SaDeleteRequests.0 = 0
cikeGlobalInitTunnels.0 = 0
cikeGlobalInitTunnelFailures.0 = 0
cikeGlobalRespTunnelFailures.0 = 0
cikeGlobalSysCapFailures.0 = 0
cikeGlobalAuthFailures.0 = 0
cikeGlobalDecryptFailures.0 = 0
cikeGlobalHashValidationFailures.0 = 0
cikeGlobalNoSaFailures.0 = 0
cipSecGlobalActiveTunnels.0 = 0
cipSecGlobalPreviousTunnels.0 = 0
```

IPsec Management Configuration Guide, Cisco IOS XE Fuji 16.8.x
VRF abc1 Pinged
The following output shows that VRF abc1 has been pinged:

```
Router3745a# ping
Protocol [ip]:
Target IP address: 10.22.1.1
Repeat count [5]:
Datagram size [100]:
Timeout in seconds [2]:
Extended commands [y]:
Source address or interface: 10.20.1.1
Type of service [0]:
Set DF bit in IP header? [no]:
Validate reply data? [no]:
Data pattern [0xABCD]:
Loose, Strict, Record, Timestamp, Verbose[none]:
Sweep range of sizes [n]:
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 10.22.1.1, timeout is 2 seconds:
Packet sent with a source address of 10.20.1.1
```

VRF abc1 Polled
Polling VRF abc1 results in the following output:

```
orcas:10> setenv SR_UTIL_COMMUNITY abc1
orcas:12> /auto/sw/packages/snmp/10.14.2.0/solaris2bin/getmany -v2c 10.1.38.25
cipSecMIBObjects
cipSecMibLevel.0 = 1
cikeGlobalActiveTunnels.0 = 1
cikeGlobalPreviousTunnels.0 = 0
cikeGlobalInOctets.0 = 336
cikeGlobalInPkt.0 = 2
cikeGlobalInDropPkts.0 = 0
cikeGlobalInNotifys.0 = 1
cikeGlobalInP2Exchgs.0 = 2
cikeGlobalInP2ExchgInvalids.0 = 0
cikeGlobalInP2ExchgRejects.0 = 0
cikeGlobalInP2SaDelRequests.0 = 0
cikeGlobalOutOctets.0 = 344
cikeGlobalOutPkts.0 = 2
cikeGlobalOutDropPkts.0 = 0
cikeGlobalOutNotifys.0 = 0
cikeGlobalOutP2Exchgs.0 = 1
```

Note: After the ping, the counters should show some nonzero values.
cipSecGlobalOutUncompOctets.0 = 704
chipSecGlobalHcOutUncompOctets.0 = 0x02c0
cipSecGlobalOutUncompOctWraps.0 = 0
cipSecGlobalOutPkts.0 = 4
cipSecGlobalOutDrops.0 = 0
cipSecGlobalOutAuths.0 = 4
cipSecGlobalOutAuthFails.0 = 0
cipSecGlobalOutEncrypts.0 = 4
cipSecGlobalOutEncryptFails.0 = 0
cipSecGlobalProtocolUsedFails.0 = 0
cipSecGlobalNoSaFails.0 = 0
cipSecGlobalSysCapFails.0 = 0
cipSecTunIKE TunnelIndex.1 = 1
cipSecTunIKE TunnelAlive.1 = true(1)
cipSecTunLocalAddr.1 = 0a 01 01 02
cipSecTunRemoteAddr.1 = 0a 01 01 01
cipSecTunKeyType.1 = ike(1)
cipSecTunEncapMode.1 = tunnel(1)
cipSecTunLifeSize.1 = 99000
cipSecTunLifeTime.1 = 5000
cipSecTunActiveTime.1 = 13749
cipSecTunSaLifeSizeThreshold.1 = 64
cipSecTunSaLifeTimeThreshold.1 = 10
chipSecTunTotalRefreshes.1 = 0
chipSecTunExpiredSainstances.1 = 0
chipSecTunCurrentSainstances.1 = 4
chipSecTunInSaDiff HellmanGrp.1 = dhGroup1(2)
cipSecTunInSaEncryptAlgo.1 = des(2)
cipSecTunInSaAhAuthAlgo.1 = hmacSha(3)
cipSecTunInSaEspAuthAlgo.1 = hmacSha(3)
cipSecTunInSaDecompAlgo.1 = none(1)
cipSecTunOutSaDiff HellmanGrp.1 = dhGroup1(2)
cipSecTunOutSaEncryptAlgo.1 = des(2)
cipSecTunOutSaAhAuthAlgo.1 = hmacSha(3)
cipSecTunOutSaEspAuthAlgo.1 = hmacSha(3)
cipSecTunOutSaCompAlgo.1 = none(1)
cipSecTunInOctets.1 = 400
chipSecTunInOctWraps.1 = 0
chipSecTunInDecompOctets.1 = 400
chipSecTunInDecompOctWraps.1 = 0
chipSecTunInPkts.1 = 4
chipSecTunInDropPkts.1 = 0
chipSecTunInReplayDropPkts.1 = 0
chipSecTunInAuths.1 = 4
chipSecTunInAuthFails.1 = 0
chipSecTunInDecrypts.1 = 4
chipSecTunInDecryptFails.1 = 0
chipSecTunOutOctets.1 = 704
chipSecTunOutOctWraps.1 = 0
chipSecTunOutUncompOctets.1 = 0x02c0
chipSecTunOutUncompOctWraps.1 = 0
chipSecTunOutPkts.1 = 4
chipSecTunOutDropPkts.1 = 0
chipSecTunOutAuths.1 = 4
chipSecTunOutAuthFails.1 = 0
chipSecTunOutEncrypts.1 = 4
chipSecTunOutEncryptFails.1 = 0
chipSecTunStatus.1 = active(1)
chipSecEndPtLocalName.1.1 = chpSecEndPtLocalType.1.1 = singleIpAddr(1)
chipSecEndPtLocalAddr1.1.1 = 16 01 01 01
chipSecEndPtLocalAddr2.1.1 = 16 01 01 01
chipSecEndPtLocalProtocol.1.1 = 0
chipSecEndPtLocalPort.1.1 = 0
chipSecEndPtRemoteName.1.1 = chipSecEndPtRemoteType.1.1 = singleIpAddr(1)
chipSecEndPtRemoteAddr1.1.1 = 14 01 01 01
chipSecEndPtRemoteAddr2.1.1 = 14 01 01 01
chipSecEndPtRemoteProtocol.1.1 = 0
VRF abc2 Polled
Polling VRF abc2 results in the following output:

The ping was completed for VRF abc1 only. Therefore, the counters of VRF abc2 should remain in the initialized state.
Additional References

### Related Documents

<table>
<thead>
<tr>
<th>Related Topic</th>
<th>Document Title</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cisco IOS commands by technology</td>
<td>Cisco IOS Release Command References</td>
</tr>
<tr>
<td>Cisco IOS master commands list</td>
<td>Master Command List</td>
</tr>
</tbody>
</table>

---

orcas:16>
### Related Topic

<table>
<thead>
<tr>
<th>Topic</th>
<th>Document Title</th>
</tr>
</thead>
<tbody>
<tr>
<td>Configuring SNMP</td>
<td>The chapter &quot;Configuring SNMP Support&quot; in the Cisco IOS Network Management Configuration Guide.</td>
</tr>
<tr>
<td>Configuring VRF-Aware IPsec</td>
<td>VRF-Aware IPsec</td>
</tr>
</tbody>
</table>

### Standards

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

### MIBs

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

### RFCs

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

### Technical Assistance

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

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

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

Table 1: Feature Information for IPsec and IKE MIB Support for Cisco VRF-Aware IPsec

<table>
<thead>
<tr>
<th>Feature Name</th>
<th>Releases</th>
<th>Feature Information</th>
</tr>
</thead>
<tbody>
<tr>
<td>IPsec and IKE MIB Support for Cisco VRF-Aware IPsec</td>
<td>IOS XE 3.1S</td>
<td>The IPsec and IKE MIB Support for the Virtual Private Network routing and forwarding- (VRF-) aware IP security (IPsec) feature allows VRF-aware IPsec to be managed with MIBs, which provide the details of IPsec statistics and performance metrics on a per VRF basis. This feature was introduced in Cisco IOS Release 12.4(4)T. This feature was integrated into Cisco IOS Release XE 3.1S. The following commands were introduced or modified: <code>debug crypto mib</code>.</td>
</tr>
</tbody>
</table>