



Security Configuration Guide: Cisco IOS Intrusion Prevention System, Cisco IOS Release 12.4T

Americas Headquarters

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



CONTENTS

Configuring Cisco IOS Intrusion Prevention System 1

Finding Feature Information 1

Prerequisites for Configuring Cisco IOS IPS 1

Restrictions for Configuring Cisco IOS IPS 2

Information About Cisco IOS IPS 3

Cisco IOS IPS Overview 3

SDEE 3

Storing SDEE Events in the Buffer 4

Transparent Cisco IOS IPS Overview 4

Transparent Bridging Overview 4

Transparent and Non-Transparent IPS Devices Configured on the Same Router 4

Signature Definition File 5

Signature Microengines Overview and Lists of Supported Engines 5

Lists of Supported Signature Engines 6

Supported Cisco IOS IPS Signatures in the attack-drop.sdf File 8

How to Configure Cisco IOS IPS on a Device 24

Configuring a Bridge Group for Transparent Cisco IOS IPS 25

Troubleshooting Tips 27

What to Do Next 28

Installing Cisco IOS IPS on a New Router 28

Upgrading to the Latest Cisco IOS IPS Signature Definition File (SDF) 30

Merging Built-In Signatures with the attack-drop.sdf File 32

Monitoring Cisco IOS IPS Signatures through Syslog Messages or SDEE 36

Troubleshooting Tips 37

Troubleshooting Cisco IOS IPS 37

Interpreting Cisco IOS IPS System Messages 37

Conditions of an SME Build Failure 39

Configuration Examples 40

Loading the Default Signatures Example 40

Loading the attack-drop.sdf Example	40
Merging the attack-drop.sdf File with the Default Built-in Signatures Example	41
Additional References	41
Feature Information for Configuring Cisco IOS IPS	42
Cisco IOS IPS 5.x Signature Format Support and Usability Enhancements	47
Finding Feature Information	47
Prerequisites for Cisco IOS IPS 5.x Signature Format Support and Usability Enhancements	48
Restrictions for Cisco IOS IPS 5.x Signature Format Support and Usability Enhancements	49
Information About Cisco IOS IPS 5.x Signature Format Support and Usability Enhancements	50
Cisco IOS IPS Overview	50
Cisco IOS IPS Signature Scanning with Lightweight Signatures	50
Signature Categories	50
Router Configuration Files and Signature Event Action Processor (SEAP)	51
Additional Risk Rating Algorithms	51
Cisco IOS IPS Signature Package	51
Preserving Configured Signature Tunings on the Local Router	51
How to Use Cisco IOS IPS 5.x Signature Format Support and Usability Enhancements	52
Retiring All Signatures and Selecting a Category of Signatures	52
What to Do Next	54
Configuring Cisco IOS IPS on Your Router	54
Loading a Signature File into Cisco IOS IPS	57
Prerequisites	57
Flexible Signatures Ordered and Incremental	57
Enabling IPS Regex Table Chaining	59
Tuning Signature Parameters	60
Tuning Signatures for a Signature ID	60
Tuning Signatures per Category	63
Enabling Signature Tunings Inheritance	65
Setting an IPS Memory Threshold	66
Setting the Target Value Rating	67
Configuring Signature Auto Updates from a Local Server	68
Monitoring Cisco IOS IPS Signatures through Syslog Messages or SDEE	70
SDEE Overview	71
Troubleshooting Tips	72

Configuration Examples for Cisco IOS IPS 5.x Signature Format Support and Usability Enhancements	73
Cisco IOS IPS Configuration Example	73
Configuring and Verifying SDEE on your Router Example	76
Additional References	76
Feature Information for Cisco IOS IPS 5.x Signature Format Support and Usability Enhancements	77
Cisco IOS IPS Support for Microsoft Engines	79
Finding Feature Information	79
Information About Cisco IOS IPS Support for Microsoft Engines	79
Cisco IOS IPS Overview	79
How to Use Cisco IOS IPS	80
Configuration Examples for Cisco IOS IPS	80
show ip ips signature Output to Verify MS Engines Example	80
Additional References	81
Feature Information for Cisco IOS IPS Support for Microsoft Engines	82
VRF Aware Cisco IOS IPS	85
Finding Feature Information	85
Prerequisites for VRF Aware Cisco IOS IPS	85
Restrictions for VRF Aware Cisco IOS IPS	86
Information About VRF Aware Cisco IOS IPS	86
Cisco IOS IPS	86
VRF	86
VRF Lite	87
Applying IPS Directly to a VRF	88
How to VRF Aware Cisco IOS IPS	88
Configuring a VRF and Applying IPS Directly to the VRF	88
Configuration Examples for VRF Aware Cisco IOS IPS	90
Example Cisco IOS IPS Configuration	91
Example VRF Aware Cisco IOS IPS Configuration Without Subinterfaces	93
Example VRF Aware Cisco IOS IPS Configuration with Subinterfaces	94
Example Multi VRF with IPS and Zone Based Policy (ZBP) Firewall	95
Examples VRF Aware Cisco IOS IPS Output and Error Message	97
Examples VRF Aware Cisco IOS IPS Output	97
Examples ErrMSG with VRF Name Output	98
Examples SDEE Messages with VRF Name	98

Examples SDEE show Commands 99

Additional References 99

Feature Information for VRF Aware Cisco IOS IPS 100



Configuring Cisco IOS Intrusion Prevention System

The Cisco IOS Intrusion Prevention System (IPS) uses the following methods to protect a network from internal and external attacks and threats:

- IPS signatures are dynamically updated and posted to Cisco.com on a regular basis so that customers can access signatures that help protect their network from the latest known network attacks.
- A Parallel Signature Scanning Engine is used to scan for multiple patterns within a signature microengine (SME) at any given time. IPS signatures are no longer scanned on a serial basis.
- Cisco IOS IPS supports both named and numbered extended access control lists (ACLs).



Note

Cisco IOS IPS restructures and replaces the existing Cisco IOS Intrusion Detection System (IDS).

- [Finding Feature Information, page 1](#)
- [Prerequisites for Configuring Cisco IOS IPS, page 1](#)
- [Restrictions for Configuring Cisco IOS IPS, page 2](#)
- [Information About Cisco IOS IPS, page 3](#)
- [How to Configure Cisco IOS IPS on a Device, page 24](#)
- [Configuration Examples, page 40](#)
- [Additional References, page 41](#)
- [Feature Information for Configuring Cisco IOS IPS, page 42](#)

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 Configuring Cisco IOS IPS

It is recommended that a new Cisco IOS image be loaded on your router before installing Cisco IOS IPS.

Compatibility with VMS IDS MC 2.3 and Cisco Router SDM

VMS IDS MC provides a web-based interface for configuring, managing, and monitoring multiple IDS sensors. SDM is a web-based device-management tool that allows users to import and edit SDFs from Cisco.com to the router. VMS IDS MC is for network-wide management while SDM is for single-device management. It is strongly recommended that customers download the SDF to an IDS MC 2.3 network management device or an SDM.

Customers can choose to download the SDF to a device other than VMS IDS MC or SDM (such as a router) through command-line interface (CLI); however, this approach is not recommended because it requires that the customer know which signatures come from which signature engines.

Restrictions for Configuring Cisco IOS IPS

Signature Support Deprecation

Effective Cisco IOS Release 12.3(8)T, the following signatures are no longer supported by Cisco IOS IPS:

- 1100 IP Fragment Attack (Attack, Atomic)

Triggers when any IP datagram is received with the “more fragments” flag set to 1 or if there is an offset indicated in the offset field. (To scan for application layer signatures across fragments, you can enable virtual fragment reassembly.)

- 1105 Broadcast Source Address (Compound/Attack)

Triggers when an IP packet with a source address of 255.255.255.255 is detected. This signature may be an indicator of an IP spoof attack or an attempt to subvert a firewall, proxy, or gateway.

- 1106 Multicast IP Source Address (Compound/Attack)

Triggers when an IP packet with a source address of 224.x.x.x is detected. This signature may be an indicator of an IP spoof attack or an attempt to subvert a firewall, proxy, or gateway.

- 8000 FTP Retrieve Password File (Attack, Atomic) SubSig ID: 2101

Triggers on string “passwd” issued during an FTP session. May indicate that someone is attempting to retrieve the password file from a machine to try and gain unauthorized access to system resources.

Memory Impact on Low-End to Midrange Routers

Intrusion detection configuration on certain routers may not support the complete list of signatures because of lack of sufficient memory. Thus, the network administrator may have to select a smaller subset of signatures or choose to use the standard 100 (built-in) signatures with which the routers are shipped.

Action Configuration through CLI No Longer Supported

Cisco IOS IPS actions (such as resetting the TCP connection) can no longer be configured through CLI. If you are using the attack-drop.sdf signature file, the signatures are preset with actions to mitigate the attack by dropping the packet and resetting the connection, if applicable. If you are using VMS or SDM to deploy signatures to the router, you must first tune the signatures to use the desired actions.

Any CLI that is issued to configure IPS actions is silently ignored.

Restrictions for Transparent Cisco IOS IPS

- Mixed-media bridging configurations are not supported. Only Ethernet media is supported.
- Layer 2 signatures are not supported.
- Multicast traffic is not processed.
- If more than two interfaces are assigned to a bridge group, any routers that are acting as first-hop gateways to hosts that are in the bridged network (the bridge group) must allow ICMP time-to-live (TTL) exceeded messages to pass.
- Spanning Tree Bridge Protocol Data Units (BPDU) and packets that are to be routed out of the bridge, if IRB is configured, are not inspected.

Information About Cisco IOS IPS

- [Cisco IOS IPS Overview, page 3](#)
- [Transparent Cisco IOS IPS Overview, page 4](#)
- [Signature Definition File, page 5](#)
- [Signature Microengines Overview and Lists of Supported Engines, page 5](#)
- [Supported Cisco IOS IPS Signatures in the attack-drop.sdf File, page 8](#)

Cisco IOS IPS Overview

The Cisco IOS IPS acts as an in-line intrusion detection sensor, watching packets and sessions as they flow through the router and scanning each packet to match any of the Cisco IOS IPS signatures. When Cisco IOS IPS detects suspicious activity, it responds before network security can be compromised and logs the event through Cisco IOS syslog messages or Security Device Event Exchange (SDEE). The network administrator can configure Cisco IOS IPS to choose the appropriate response to various threats. When packets in a session match a signature, Cisco IOS IPS can take any of the following actions, as appropriate:

- Send an alarm to a syslog server or a centralized management interface
- Drop the packet
- Reset the connection
- Deny traffic from the source IP address of the attacker for a specified amount of time
- Deny traffic on the connection for which the signature was seen for a specified amount of time

Cisco developed its Cisco IOS software-based intrusion-prevention capabilities and Cisco IOS Firewall with flexibility in mind, so that individual signatures can be disabled in case of false positives. Generally, it is preferable to enable both the firewall and Cisco IOS IPS to support network security policies. However, each of these features may be enabled independently and on different router interfaces.

- [SDEE, page 3](#)

SDEE

SDEE is an application-level communication protocol that is used to exchange IPS messages between IPS clients and IPS servers.

SDEE is always running, but it does not receive and process events from IPS unless SDEE notification is enabled. If SDEE notification is not enabled and a client sends a request, SDEE responds with a fault response message, indicating that notification is not enabled.

- [Storing SDEE Events in the Buffer, page 4](#)

Storing SDEE Events in the Buffer

When SDEE notification is enabled (through the **ip ips notify sdee** command), 200 events can automatically be stored in the buffer. When SDEE notification is disabled, all stored events are lost. A new buffer is allocated when the notifications are reenabled.

When specifying the size of an events buffer, note the following functionality:

- It is circular. When the end of the buffer is reached, the buffer starts overwriting the earliest stored events. (If overwritten events have not yet been reported, a buffer overflow notice is received.)
- If a new, smaller buffer is requested, all events that are stored in the previous buffer is lost.
- If a new, larger buffer is requested, all existing events is saved.

Transparent Cisco IOS IPS Overview

If customers want to protect their network through a typical Cisco IOS IPS device, they must manually readdress each of the statically defined devices on the trusted network. A transparent Cisco IOS IPS device allows customers to “drop” a Layer 3 IPS device in front of the devices that need to be protected. Thus, the tedious and costly overhead that is required to renumber devices on the trusted network is eliminated.

A transparent Cisco IOS IPS device acts as a Layer 3 IPS between bridged interfaces. (The current implementation of the transparent IPS does not support Layer 2 IPS functionality; thus, IPS can act only as a Layer 3 device.)

- [Transparent Bridging Overview, page 4](#)
- [Transparent and Non-Transparent IPS Devices Configured on the Same Router, page 4](#)

Transparent Bridging Overview

If configured for bridging, a Cisco IOS device can bridge any number of interfaces. The device can complete basic bridging tasks such as learning MAC addresses on ports to restrict collision domains and running Spanning Tree Protocol (STP) to prevent looping in the topology.

Within bridging, a user can configure Integrated Routed Bridging (IRB), which allows a device to bridge on some interfaces while a Layer 3 Bridged Virtual Interface (BVI) is presented for routing. The bridge can determine whether the packet is to be bridged or routed on the basis of the destination of the Layer 2 or Layer 3 IP address in the packet. Configured with an IP address, the BVI can manage the device even if there is no interface configured for routing.

Transparent and Non-Transparent IPS Devices Configured on the Same Router

A transparent IPS device supports a BVI for routing, so a packet that comes in on a bridged interface can be bridged or routed out of the BVI. This functionality allows a transparent IPS device and a nontransparent IPS device to be configured on the same router. The transparent IPS device operates on the bridged packets while the “normal” IPS device operates on the routed packets. For example, if you have six interfaces on your router and two of them are in a bridge group, you can simultaneously configure and run normal IPS inspection on the remaining four interfaces.

Users can also configure transparent IPS and a transparent firewall on the same device. For more information on the transparent firewall, see the document “Transparent Cisco IOS Firewall.”

Signature Definition File

Cisco IOS IPS allows customers to choose between any of the following options when loading IPS signatures onto a device:

- Loading the default, built-in signatures.
- Download the signature definition file (SDF) on the router by using the Cisco Router and Security Device Manager (SDM) to have the latest available detection of security threats. Go to the following link to download the SDF:

<http://www.cisco.com/cgi-bin/tablebuild.pl/ios-sigup>

**Note**

SDM automatically recommends which SDF that should be used the first time IPS is enabled on the router.

After the SDF is downloaded on to their router, the SDM can immediately begin scanning for new signatures.

An SDF has definitions for each signature it contains. After signatures are loaded and compiled onto a router running Cisco IOS IPS, IPS can begin detecting the new signatures immediately. If the default, built-in signatures that are shipped with the routers are not used, then one of three different types of SDFs can be selected for download, which are pre-configured for routers with memory requirements:

- **attack-drop.sdf** file (which is a static file that has 83 signatures) is used for routers with less than 128MB memory.
- **128MB.sdf** (which has about 300 signatures) is used for routers with 128 MB or more memory.
- **256MB.sdf** (which has about 500 signatures) is used for routers with 256 MB or more memory.

The **attack-drop.sdf**, **128MB.sdf**, and **256MB.sdf** are available in flash on all Cisco access routers that are shipped with Cisco IOS Release 12.4 or later. One of these files can then be loaded directly from flash into the Cisco IOS IPS system. If flash is erased, the SDF file may also be erased. If a Cisco IOS image is copied to flash and there is a prompt to erase the contents of flash before copying the new image, you might risk erasing the SDF file. If the SDF file is erased, the router refers to the built-in signatures within the Cisco IOS image.

The SDF file can also be downloaded onto your router from Cisco.com through SDM. SDF files can be loaded through IDS MC 2.3, which can be launched from CSM 3.0.

**Note**

SDF files can be used only with 12.4(9)Tx or earlier IOS Images and mainline images.

To help detect the latest vulnerabilities, Cisco provides signature updates on Cisco.com on a regular basis. Users can use VMS or SDM to download these signature updates, tune the signature parameters as necessary, and deploy the new SDF to a Cisco IOS IPS router.

Signature Microengines Overview and Lists of Supported Engines

Cisco IOS IPS uses SMEs to load the SDF and scan signatures. Signatures contained within the SDF are handled by a variety of SMEs. The SDF typically contains signature definitions for multiple engines. The SME typically corresponds to the protocol in which the signature occurs and looks for malicious activity in that protocol.

A packet is processed by several SMEs. Each SME scans for various conditions that can lead to a signature pattern match. When an SME scans the packets, it extracts certain values, searching for patterns within the

packet through the regular expression engine. See "Lists of Supported Signature Engines" for a list of supported signature engines.

- [Lists of Supported Signature Engines, page 6](#)

Lists of Supported Signature Engines



Note

If the SDF contains a signature that requires an engine that is not supported, the engine is ignored and an error message is displayed. If a signature within a supported engine contains a parameter that is not supported, the parameter is ignored and an error message is displayed.

Table 1 *Supported Signature Engines for Cisco IOS IPS*

Signature Engine	Initial Supported Cisco IOS Release	Parameter Exceptions ¹
ATOMIC.L3.IP	12.3(8)T	--
ATOMIC.ICMP	12.3(8)T	--
ATOMIC.IPOPTIONS	12.3(8)T	--
ATOMIC.TCP	12.3(8)T	--
ATOMIC.UDP	12.3(8)T	--
SERVICE.DNS	12.3(8)T	--
SERVICE.HTTP	12.3(8)T	ServicePorts (applicable only in Cisco IOS Release 12.3(8)T)
SERVICE.FTP	12.3(8)T	ServicePorts
SERVICE.SMTP	12.3(8)T	ServicePorts
SERVICE.RPC	12.3(8)T	ServicePorts, Unique, and isSweep
STRING.ICMP	12.3(14)T	--
STRING.TCP	12.3(14)T	--
STRING.UDP	12.3(14)T	--

The table below lists support for the 100 signatures that are available in Cisco IOS IDS prior to Cisco IOS Release 12.3(8)T. As of Cisco IOS Release 12.3(8)T, these 100 signatures are a part of the Cisco IOS IPS built-in SDF. By default, signatures are loaded from this built-in SDF. The table above lists support for these 100 signatures under Cisco IOS IPS.

¹ The following parameters, which are defined in all signature engines, are currently not supported: AlarmThrottle=Summarize (all other values are supported), MaxInspectLength, MaxTTL, Protocol, ResetAfterIdle, StorageKey, and SummaryKey.

**Note**

Because Cisco IOS IPS counts signatures on the basis of signature-id and subsignature-id, the 100 signatures under Cisco IOS IDS are counted as 132 signatures under Cisco IOS IPS.

Table 2 *Support for Signatures Available in Cisco IOS IDS (prior to 12.3(8)T)*

Signature ID	Count	Signature Engine
1000-1006	7	ATOMIC.IPOPTIONS
1101, 1102	2	ATOMIC.L3.IP
1004, 1007	2	ATOMIC.L3.IP
2000-2012, 2150	14	ATOMIC.ICMP
2151, 2154	2	ATOMIC.L3.IP
3038-3043	6	ATOMIC.TCP
3100-3107	8	SERVICE.SMTP
3153, 3154	2	SERVICE.FTP
4050-4052, 4600	4	ATOMIC.UDP
6100-6103	4	SERVICE.RPC
6150-6155	6	SERVICE.RPC
6175, 6180, 6190	3	SERVICE.RPC
6050-6057	8	SERVICE.DNS
6062-6063	2	SERVICE.DNS
3215, 3229, 3223	3	SERVICE.HTTP
5034-5035	2	SERVICE.HTTP
5041, 5043-5045	4	SERVICE.HTTP
5050, 5055, 5071	3	SERVICE.HTTP
5081, 5090, 5123	3	SERVICE.HTTP
5114, 5116-5118	4	SERVICE.HTTP
1100	1	Not applicable. Signature is replaced by 12xx series.
1105-1106	2	Cisco IOS IPS deprecates these signatures, which do not appear in the SDF.

Signature ID	Count	Signature Engine
1201-1208	10	OTHER ² (fragment attack signatures)
3050	2	OTHER 1 (SYN attack signatures)
3150-3152	3	STRING.TCP
4100	1	STRING.UDP
8000	1	Cisco IOS IPS deprecates these signatures, which do not appear in the SDF.

Supported Cisco IOS IPS Signatures in the attack-drop.sdf File

Customers can choose to use Cisco IOS IPS in one of the following ways:

- Download new signatures that are posted on Cisco.com. These signatures can be obtained at the Cisco Intrusion Prevention Alert Center web page. (You must have a valid Cisco.com account to access this web page.)
- Download the attack-drop.sdf file, which contains the signatures that are identified in the table below.

Table 3 Cisco IOS IPS Signatures Supported in Cisco IOS Release 12.3(8)T

Signature ID: SubSig ID	Signature Name	Action ³	SME	Signature Description
1006:0	IP options-Strict Source Route	A, D	ATOMIC.IPOPTIONS	Triggers on receipt of an IP datagram in which the IP option list for the datagram includes option 2 (Strict Source Routing).
1102:0	Impossible IP Packet	A, D	ATOMIC.L3.IP	Triggers when an IP packet arrives with source equal to destination address. This signature catches the Land Attack.

² The OTHER engine contains existing, hard-coded signatures. Although the standard SDF contains an entry for these signatures, the engine is not dynamically updated. If the SDF that is loaded onto the engine does not contain the signature, the signature is treated as though it has been disabled.

³ A = alarm, D = drop, R = reset

Signature ID: SubSig ID	Signature Name	Action ³	SME	Signature Description
1104:0	IP Localhost Source Spoof	A, D	ATOMIC.L3.IP	Triggers when an IP packet with the address of 127.0.0.1, a local host IP address that should never be seen on the network, is detected. This signature can detect the Blaster attack.
1108:0	IP Packet with Proto 11	A, D	ATOMIC.L3.IP	Alarms upon detecting IP traffic with the protocol set to 11. There have been known “backdoors” running on IP protocol 11.
2154:0	Ping Of Death Attack	A, D	ATOMIC.L3.IP	Triggers when an IP datagram is received with the protocol field in the IP header set to 1 (Internet Control Message Protocol [ICMP]), the Last Fragment bit is set. The IP offset (which represents the starting position of this fragment in the original packet and which is in 8-byte units) plus the rest of the packet is greater than the maximum size for an IP packet.
3038:0	Fragmented NULL TCP Packet	A, D	ATOMIC.TCP	Triggers when a single, fragmented TCP packet with none of the SYN, FIN, ACK, or RST flags set has been sent to a specific host. A reconnaissance sweep of your network may be in progress.

³ A = alarm, D = drop, R = reset

Signature ID: SubSig ID	Signature Name	Action ³	SME	Signature Description
3039:0	Fragmented Orphaned FIN packet	A, D	ATOMIC.TCP	Triggers when a single, fragmented, orphan TCP FIN packet is sent to a privileged port (having a port number less than 1024) on a specific host. A reconnaissance sweep of your network may be in progress.
3040:0	NULL TCP Packet	A, D	ATOMIC.TCP	Triggers when a single TCP packet with none of the SYN, FIN, ACK, or RST flags set has been sent to a specific host. A reconnaissance sweep of your network may be in progress.
3041:0	SYN/FIN Packet	A, D	ATOMIC.TCP	Triggers when a single TCP packet with the SYN and FIN flags set is sent to a specific host. A reconnaissance sweep of your network may be in progress. The use of this type of packet indicates an attempt to conceal the sweep.
3043:0	Fragmented SYN/FIN Packet	A, D	ATOMIC.TCP	Triggers when a single, fragmented TCP packet with the SYN and FIN flags set is sent to a specific host. A reconnaissance sweep of your network may be in progress. The use of this type of packet indicates an attempt to conceal the sweep.

³ A = alarm, D = drop, R = reset

Signature ID: SubSig ID	Signature Name	Action ³	SME	Signature Description
3129:0	Mimail Virus C Variant File Attachment	A, D, R	SERVICE.SMTP	Fires when an e-mail attachment matching the C Variant of the Mimail virus is detected. The virus sends itself to recipients as the e-mail attachment "photos.zip" that contains the file "photos.jpg.exe" and has "our private photos" in the e-mail subject line. If launched, the virus harvests e-mail addresses and possible mail servers from the infected system.
3140:3	Bagle Virus Activity ⁴	A, D, R	SERVICE.HTTP	Fires when HTTP propagation using .jpeg associated with the .Q variant is detected.
3140:4	Bagle Virus Activity ⁵	A, D, R	SERVICE.HTTP	Fires when HTTP propagation using .php associated with the .Q variant is detected.
3300:0	NetBIOS OOB Data	A, D	ATOMIC.TCP	Triggers when an attempt to send Out Of Band data to port 139 is detected.
5045:0	WWW xterm display attack	A, D, R	SERVICE.HTTP	Triggers when any cgi-bin script attempts to execute the command xterm -display. An attempt to illegally log into your system may be in progress.

³ A = alarm, D = drop, R = reset

⁴ This signature requires port to application mapping (PAM) configuration through the command ip port-map http port 81 .

⁵ This signature requires PAM configuration through the command ip port-map http port 81 .

Signature ID: SubSig ID	Signature Name	Action ³	SME	Signature Description
5047:0	WWW Server Side Include POST attack	A, D, R	SERVICE.HTTP	Triggers when an attempt is made to embed a server side include (SSI) in an http POST command. An attempt to illegally access system resources may be in progress.
5055:0	HTTP Basic Authentication Overflow	A, D	SERVICE.HTTP	A buffer overflow can occur on vulnerable web servers if a very large username and password combination is used with basic authentication.
5071:0	WWW msacds.dll Attack	A, D, R	SERVICE.HTTP	An attempt has been made to execute commands or view secured files, with privileged access. Administrators are highly recommended to check the affected systems to ensure that they have not been illicitly modified.
5081:0	WWW WinNT cmd.exe Access	A, D, R	SERVICE.HTTP	Triggers when the use of the Windows NT cmd.exe is detected in a URL. This signature can detect the NIMDA attack.

³ A = alarm, D = drop, R = reset

Signature ID: SubSig ID	Signature Name	Action ³	SME	Signature Description
5114: 0 5114:1 5114:2	WWW IIS Unicode Attack	A, D, R	SERVICE.HTTP	Triggers when an attempt to exploit the Unicode ../ directory traversal vulnerability is detected. Looks for the commonly exploited combinations that are included in publicly available exploit scripts. SubSig 2 is know to detect the NIMDA attack.
5126:0	WWW IIS .ida Indexing Service Overflow	A, D, R	SERVICE.HTTP	Alarms if web traffic is detected with the ISAPI extension .ida? and a data size of greater 200 characters.
5159:0	phpMyAdmin Cmd Exec	A, D, R	SERVICE.HTTP	Triggers when access to sql.php with the arguments goto and btnDrop=No is detected.
5184:0	Apache Authentication Module ByPass	A, D, R	SERVICE.HTTP	Fires upon detecting a select statement on the Authorization line of an HTTP header.
5188:0	HTTP Tunneling ⁶ SubSig 0: GotomyPC	A, D, R	SERVICE.HTTP	Triggers when a computer connects to gotomyPC site.
5188:1	HTTP Tunneling SubSig 1: FireThru	A, D, R	SERVICE.HTTP	Triggers when an attempt to use /cgi-bin/proxy is detected. The /cgi-bin/proxy is used to tunnel connections to other ports using web ports.

³ A = alarm, D = drop, R = reset

⁶ This signature requires PAM configuration through the command ip port-map http port 8200 .

Signature ID: SubSig ID	Signature Name	Action ³	SME	Signature Description
5188:2	HTTP Tunneling SubSig 2: HTTP Port	A, D, R	SERVICE.HTTP	Triggers when a connection is made to exectech-va.com. The site runs a server, which connects to the requested resource and passes the information back to the client on web ports.
5188:3	HTTP Tunneling SubSig 3: httptunnel	A, D, R	SERVICE.HTTP	Triggers when /index/html? is detected on POST request.
5245:0	HTTP 1.1 Chunked Encoding Transfer	A, D, R	SERVICE.HTTP	Fires when HTTP 1.1 chunked encoding transfer activity is detected. This signature is known to detect the Scalper Worm.
5326:0	Root.exe access	A, D, R	SERVICE.HTTP	Alarms upon detecting an HTTP request for root.exe. This signature is known to detect the NIMDA attack.
5329:0	Apache/mod_ssl Worm Probe	A, D, R	SERVICE.HTTP	Fires when a probe by the Apache/mod_ssl worm is detected. If the worm detects a vulnerable web server, a buffer overflow attack is sent to HTTPS port (TCP 443) of the web server. The worm then attempts to propagate itself to the newly infected web server and begins scanning for new hosts to attack.

³ A = alarm, D = drop, R = reset

Signature ID: SubSig ID	Signature Name	Action ³	SME	Signature Description
5364:0	IIS WebDAV Overflow	A, D, R	SERVICE.HTTP	Fires when a long HTTP request (65000+ characters) is detected with an HTTP header option "Translate:". An attack to exploit a weakness in the WebDAV component of the IIS web server may be in progress.
5390:0	Swen Worm HTTP Counter Update Attempt	A, D, R	SERVICE.HTTP	Triggers when an attempt to access the URL "/bin/counter.gif/link=bacillus" is detected. A system may be infected by the Swen worm trying the update a counter on a web page located on the server "ww2.fce.vutbr.cz."
5400:0	Beagle.B (Bagle.B) Web Beacon	A, D, R	SERVICE.HTTP	Fires when a request is made for the script 1.php or 2.php residing on the hosts "www.47df.de" or "www.strato.de," followed by the argument indicating the trojan's listening port number, p=8866.
6055:0	DNS Inverse Query Buffer Overflow	A, D	SERVICE.DNS	Triggers when an IQUERY request arrives with a data section that is greater than 255 characters.
6055:1		R for subsig 1, 2		
6055:2				

³ A = alarm, D = drop, R = reset

Signature ID: SubSig ID	Signature Name	Action ³	SME	Signature Description
6056:0 6056:1 6056:2	DNS NXT Buffer Overflow	A, D R for subsig 1, 2	SERVICE.DNS	Triggers when a Domain Name System (DNS) server response arrives with a long NXT resource where the length of the resource data is greater than 2069 bytes or the length of the TCP stream containing the NXT resource is greater than 3000 bytes.
6057:0 6057:1 6057:2	DNS SIG Buffer Overflow	A, D R for subsig 1, 2	SERVICE.DNS	Triggers when a DNS server response arrives with a long SIG resource where the length of the resource data is greater than 2069 bytes or the length of the TCP stream that contains the SIG resource is greater than 3000 bytes.
6058:0 6058:1	DNS SRV DoS	A, D R for subsig 1	SERVICE.DNS	Alarms when a DNS query type SRV and DNS query class IN is detected with more than ten pointer jumps in the SRV resource record.
6059:0 6059:1 6059:2	DNS TSIG Overflow	A, D R for subsig 2	SERVICE.DNS	Alarms when a DNS query type TSIG is detected and the domain name is greater than 255 characters. This signature is known to detect the Lion work.

³ A = alarm, D = drop, R = reset

Signature ID: SubSig ID	Signature Name	Action ³	SME	Signature Description
6060:0 6060:1 6060:2 6060:3	DNS Complan Overflow	A, D R for subsig 2, 3	SERVICE.DNS	Alarms when a Name Server (NS) record is detected with a domain name greater than 255 characters and the IP address is 0.0.0.0, 255.255.255.255 or a multicast address of the form 224.x.x.x.
6100:0 6100:1	RPC Port Registration	A, D R for subsig 1	SERVICE.RPC	Triggers when attempts are made to register new RPC services on a target host. Port registration is the method used by new services to report their presence to the portmapper and to gain access to a port. Their presence is then advertised by the portmapper.
6101:0 6101:1	RPC Port Unregistration	A, D R for subsig 1	SERVICE.RPC	Triggers when attempts are made to unregister existing Remote Procedure Call (RPC) services on a target host. Port unregistration is the method used by services to report their absence to the portmapper and to remove themselves from the active port map.
6104:0 6104:1	RPC Set Spoof	A, D R for subsig 1	SERVICE.RPC	Triggers when an RPC set request with a source address of 127.x.x.x is detected.
6105:0 6105:1	RPC Unset Spoof	A, D R for subsig 1	SERVICE.RPC	Triggers when an RPC unset request with a source address of 127.x.x.x is detected.

³ A = alarm, D = drop, R = reset

Signature ID: SubSig ID	Signature Name	Action ³	SME	Signature Description
6188:0	statd dot dot	A, D	SERVICE.RPC	Alarms upon detecting a dot dot slash (./) sequence sent to the statd RPC service.
6189:0 6189:1	statd automount attack	A, D R for subsig 1	SERVICE.RPC	Alarms upon detecting a statd bounce attack on the automount process. This attack targets a vulnerability in the automount process that could be exploited only through localhost.
6190:0 6190:1	statd Buffer Overflow	A, D R for subsig 1	SERVICE.RPC	Triggers when a large statd request is sent. This attack could be an attempt to overflow a buffer and gain access to system resources.
6191:0 6191:1	RPC.tooltalk buffer overflow	A, D R for subsig 1	SERVICE.RPC	Fires when an attempt is made to overflow an internal buffer in the tooltalk rpc program.
6192:0 6192:1	RPC mountd Buffer Overflow	A, D R for subsig 1	SERVICE.RPC	Triggers on an attempt to overflow a buffer in the RPC mountd application. This attack may result in unauthorized access to system resources.
6193:0 6193:1	RPC CMSD Buffer Overflow	A, D R for subsig 1	SERVICE.RPC	Fires when an attempt is made to overflow an internal buffer in the Calendar Manager Service Daemon, rpc.cmsd.
6194:0 6194:1	sadmind RPC Buffer Overflow	A, D R for subsig 1	SERVICE.RPC	Fires when a call to RPC program number 100232 procedure 1 with a UDP packet length greater than 1024 bytes is detected.

³ A = alarm, D = drop, R = reset

Signature ID: SubSig ID	Signature Name	Action ³	SME	Signature Description
6195:0 6195:1	RPC amd Buffer Overflow	A, D R for subsig 1	SERVICE.RPC	Detects the exploitation of the RPC AMD Buffer Overflow vulnerability. The trigger for this signature is an RPC call to the berkeley automounter daemons rpc program (300019) procedure 7 that has a UDP length greater than 1024 bytes or a TCP stream length greater than 1024 bytes. The TCP stream length is defined by the contents of the two bytes preceding the RPC header in a TCP packet.
6196:0 6196:1	snmpXdmid Buffer Overflow	A, D R for subsig 1	SERVICE.RPC	Fires when an abnormally long call to the RPC program 100249 (snmpXdmid) and procedure 257 is detected.
6197:0 6197:1	rpc yppaswdd overflow	A, D R for subsig 0	SERVICE.RPC	Fires when an overflow attempt is detected. This alarm looks for an abnormally large argument in the attempt to access yppaswdd.
6276:0 6276:1	TooltalkDB overflow	A, D R for subsig 1	SERVICE.RPC	Alarms upon detecting an RPC connection to rpc program number 100083 using procedure 103 with a buffer greater than 1024.
9200:0	Back Door Response (TCP 12345)	A, D	ATOMIC.TCP	Fires upon detecting a TCP SYN/ACK packet from port 12345, which is a known trojan port for NetBus as others.

³ A = alarm, D = drop, R = reset

Signature ID: SubSig ID	Signature Name	Action ³	SME	Signature Description
9201:0	Back Door Response (TCP 31337)	A, D	ATOMIC.TCP	Fires upon detecting a TCP SYN/ACK packet from port 31337, which is a known trojan port for BackFire.
9202:0	Back Door Response (TCP 1524)	A, D	ATOMIC.TCP	Fires upon detecting a TCP SYN/ACK packet from port 1524, which is a common back door placed on machines by worms and hackers.
9203:0	Back Door Response (TCP 2773)	A, D	ATOMIC.TCP	Fires upon detecting a TCP SYN/ACK packet from port 2773, which is a known trojan port for SubSeven.
9204:0	Back Door Response (TCP 2774)	A, D	ATOMIC.TCP	Fires upon detecting a TCP SYN/ACK packet from port 2774, which is a known trojan port for SubSeven.
9205:0	Back Door Response (TCP 20034)	A, D	ATOMIC.TCP	Fires upon detecting a TCP SYN/ACK packet from port 20034, which is a known trojan port for Netbus Pro.
9206:0	Back Door Response (TCP 27374)	A, D	ATOMIC.TCP	Fires upon detecting a TCP SYN/ACK packet from port 27374, which is a known trojan port for SubSeven.
9207:0	Back Door Response (TCP 1234)	A, D	ATOMIC.TCP	Fires upon detecting a TCP SYN/ACK packet from port 1234, which is a known trojan port for SubSeven.
9208:0	Back Door Response (TCP 1999)	A, D	ATOMIC.TCP	Fires upon detecting a TCP SYN/ACK packet from port 1999, which is a known trojan port for SubSeven.

³ A = alarm, D = drop, R = reset

Signature ID: SubSig ID	Signature Name	Action ³	SME	Signature Description
9209:0	Back Door Response (TCP 6711)	A, D	ATOMIC.TCP	Fires upon detecting a TCP SYN/ACK packet from port 6711, which is a known trojan port for SubSeven.
9210:0	Back Door Response (TCP 6712)	A, D	ATOMIC.TCP	Fires upon detecting a TCP SYN/ACK packet from port 6712, which is a known trojan port for SubSeven.
9211:0	Back Door Response (TCP 6713)	A, D	ATOMIC.TCP	Fires upon detecting a TCP SYN/ACK packet from port 6713, which is a known trojan port for SubSeven.
9212:0	Back Door Response (TCP 6776)	A, D	ATOMIC.TCP	Fires upon detecting a TCP SYN/ACK packet from port 6776, which is a known trojan port for SubSeven.
9213:0	Back Door Response (TCP 16959)	A, D	ATOMIC.TCP	Fires upon detecting a TCP SYN/ACK packet from port 16959, which is a known trojan port for SubSeven.
9214:0	Back Door Response (TCP 27573)	A, D	ATOMIC.TCP	Fires upon detecting a TCP SYN/ACK packet from port 27573, which is a known trojan port for SubSeven.
9215:0	Back Door Response (TCP 23432)	A, D	ATOMIC.TCP	Fires upon detecting a TCP SYN/ACK packet from port 23432, which is a known trojan port for asylum.
9216:0	Back Door Response (TCP 5400)	A, D	ATOMIC.TCP	Fires upon detecting a TCP SYN/ACK packet from port 5400, which is a known trojan port for back-construction.

³ A = alarm, D = drop, R = reset

Signature ID: SubSig ID	Signature Name	Action ³	SME	Signature Description
9217:0	Back Door Response (TCP 5401)	A, D	ATOMIC.TCP	Fires upon detecting a TCP SYN/ACK packet from port 5401, which is a known trojan port for back-construction.
9218:0	Back Door Response (TCP 2115)	A, D	ATOMIC.TCP	Fires upon detecting a TCP SYN/ACK packet from port 2115, which is a known trojan port for bugs.
9223:0	Back Door Response (TCP 36794)	A, D	ATOMIC.TCP	Fires upon detecting a TCP SYN/ACK packet from port 36794, which is a known trojan port for NetBus as well Bugbear.
9224:0	Back Door Response (TCP 10168)	A, D	ATOMIC.TCP	Fires upon detecting a TCP SYN/ACK packet from port 10168, which is a known trojan port for lovegate.
9225:0	Back Door Response (TCP 20168)	A, D	ATOMIC.TCP	Fires upon detecting a TCP SYN/ACK packet from port 20168, which is a known trojan port for lovegate.
9226:0	Back Door Response (TCP 1092)	A, D	ATOMIC.TCP	Fires upon detecting a TCP SYN/ACK packet from port 1092, which is a known trojan port for lovegate.
9227:0	Back Door Response (TCP 2018)	A, D	ATOMIC.TCP	Fires upon detecting a TCP SYN/ACK packet from port 2018, which is a known trojan port for fizzer.
9228:0	Back Door Response (TCP 2019)	A, D	ATOMIC.TCP	Fires upon detecting a TCP SYN/ACK packet from port 2019, which is a known trojan port for fizzer.

³ A = alarm, D = drop, R = reset

Signature ID: SubSig ID	Signature Name	Action ³	SME	Signature Description
9229:0	Back Door Response (TCP 2020)	A, D	ATOMIC.TCP	Fires upon detecting a TCP SYN/ACK packet from port 2020, which is a known trojan port for fizzer.
9230:0	Back Door Response (TCP 2021)	A, D	ATOMIC.TCP	Fires upon detecting a TCP SYN/ACK packet from port 2021, which is a known trojan port for fizzer.
9231:0	Back Door Response (TCP 6777)	A, D	ATOMIC.TCP	Fires upon detecting a TCP SYN/ACK packet from port 6777, which is a known trojan port for Beagle (Bagle).
9232:0	Back Door Response (TCP 5190)	A, D	ATOMIC.TCP	Fires upon detecting a TCP SYN/ACK packet from port 5190, which is a known trojan port for the Anig worm.
9233:0	Back Door Response (TCP 3127)	A, D	ATOMIC.TCP	Fires upon detecting a TCP SYN/ACK packet from port 3127, which is a known trojan port for the MyDoom.A / Novarg.A virus.
9236:0	Back Door Response (TCP 3128)	A, D	ATOMIC.TCP	Fires upon detecting a TCP SYN/ACK packet from port 3128, which is a known trojan port for the MyDoom.B / Novarg.B virus.
9237:0	Back Door Response (TCP 8866)	A, D	ATOMIC.TCP	Fires upon detecting a TCP SYN/ACK packet from port 8866, which is a known trojan port for the Beagle.B (Bagle.B) virus.

³ A = alarm, D = drop, R = reset

Signature ID: SubSig ID	Signature Name	Action ³	SME	Signature Description
9238:0	Back Door Response (TCP 2766)	A, D	ATOMIC.TCP	Fires upon detecting a TCP SYN/ACK packet from port 2766, which is a known trojan port for the DeadHat worm.
9239:0	Back Door Response (TCP 2745)	A, D	ATOMIC.TCP	Fires upon detecting a TCP SYN/ACK packet from port 2745, which is a known trojan port for the Bagle.H-J virus.
9240:0	Back Door Response (TCP 2556)	A, D	ATOMIC.TCP	Fires upon detecting a TCP SYN/ACK packet from port 2556, which is a known trojan port for the Bagle (.M.N.O.P) virus.
9241:0	Back Door Response (TCP 4751)	A, D	ATOMIC.TCP	Fires upon detecting a TCP SYN/ACK packet from port 4751, which is a known trojan port for the Bagle.U virus.

How to Configure Cisco IOS IPS on a Device

If you want to configure transparent Cisco IOS IPS, you must configure a bridge group before loading IPS onto a device. To configure a bridge group, see the section “Configuring a Bridge Group for Transparent Cisco IOS IPS.” If you do not want to configure transparent IPS, skip this task and immediately begin installing IPS onto your device as shown in the tasks below.

Before configuring Cisco IOS IPS on a router, you should determine which one of the following deployment scenarios best addresses your situation and configure the associated task, as appropriate:

- You are loading signatures onto a router through VMS IDS MC or SDM:

To use VMS IDS MC, see the documents on the VMS index.

To use SDM, see the chapter “Intrusion Prevention System” in the *Cisco Router and Security Device Manager 2.5 User Guide*.

- You are installing a new router with the latest version of Cisco IOS IPS.

To perform this task, see the section “Installing Cisco IOS IPS on a New Router.”

- Your network is transitioning to Cisco IOS IPS in Cisco IOS Release 12.3(8)T or later.

To perform this task, see the section “Upgrading to the Latest Cisco IOS IPS Signature Definition File (SDF).”

³ A = alarm, D = drop, R = reset

- You are merging the default (built-in) Cisco IOS IPS signatures with the latest version of the Cisco IOS IPS signature detection file, “attack-drop.sdf.”

To perform this task, see the section “Merging Built-In Signatures with the attack-drop.sdf File”

After you have configured Cisco IOS IPS on your router, see the following optional sections:

- [Configuring a Bridge Group for Transparent Cisco IOS IPS, page 25](#)
- [Installing Cisco IOS IPS on a New Router, page 28](#)
- [Upgrading to the Latest Cisco IOS IPS Signature Definition File \(SDF\), page 30](#)
- [Merging Built-In Signatures with the attack-drop.sdf File, page 32](#)
- [Monitoring Cisco IOS IPS Signatures through Syslog Messages or SDEE, page 36](#)
- [Troubleshooting Cisco IOS IPS, page 37](#)

Configuring a Bridge Group for Transparent Cisco IOS IPS



Note

You should configure a bridge group only if you want to configure transparent IPS.

- If a BVI is not configured, you must disable IP routing (through the **no ip routing** command) for the bridging operation to take effect.
- If configured, a BVI must be configured with an IP address in the same subnet.
- You *must* configure a BVI if more than two interfaces are placed in a bridge group.
- Bridging between VLAN trunks works only for dot1q encapsulation; Inter-Switch Link (ISL) encapsulation does not work.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **bridge** *bridge-group* **protocol** {**dec** | **ibm** | **ieee** | **vlan-bridge**}
4. **interface** *type number*
5. **bridge-group** *bridge-group*
6. **exit**
7. **bridge irb**
8. **bridge** *bridge-group* **route protocol**
9. **interface** *type number*
10. **ip address** *ip-address mask*
11. **no shutdown**

DETAILED STEPS

	Command or Action	Purpose
Step 1	<p>enable</p> <p>Example:</p> <pre>Router> enable</pre>	<p>Enables privileged EXEC mode.</p> <ul style="list-style-type: none"> Enter your password if prompted.
Step 2	<p>configure terminal</p> <p>Example:</p> <pre>Router# configure terminal</pre>	<p>Enters global configuration mode.</p>
Step 3	<p>bridge <i>bridge-group</i> protocol {dec ibm ieee vlan-bridge}</p> <p>Example:</p> <pre>Router(config)# bridge 1 protocol ieee</pre>	<p>Defines the type of Spanning Tree Protocol (STP).</p>
Step 4	<p>interface <i>type number</i></p> <p>Example:</p> <pre>Router(config)# interface Ethernet0</pre>	<p>Configures an interface type and enters interface configuration mode.</p>
Step 5	<p>bridge-group <i>bridge-group</i></p> <p>Example:</p> <pre>Router(config-if)# bridge-group 1</pre>	<p>Assigns each network interface to a bridge group.</p> <p>Note Complete Step 4 and Step 5 for each interface you want to assign to a bridge group.</p> <p>Note You can also assign subinterfaces to a bridge group to control bridging between VLANs.</p>
Step 6	<p>exit</p> <p>Example:</p> <pre>Router(config-if)# exit</pre>	<p>Exits interface configuration mode.</p>
Step 7	<p>bridge irb</p> <p>Example:</p> <pre>Router(config)# bridge irb</pre>	<p>Enables the Cisco IOS software to route a given protocol between routed interfaces and bridge groups or to route a given protocol between bridge groups.</p> <p>Note Step 7 through Step 11 are necessary only if you want to configure a BVI.</p>

	Command or Action	Purpose
Step 8	<p>bridge <i>bridge-group</i> <i>route protocol</i></p> <p>Example:</p> <pre>Router(config)# bridge 1 route ip</pre>	Enables the routing of a specified protocol in a specified bridge group.
Step 9	<p>interface <i>type number</i></p> <p>Example:</p> <pre>Router(config)# interface BVI1</pre>	Configures a BVI and enters interface configuration mode.
Step 10	<p>ip address <i>ip-address mask</i></p> <p>Example:</p> <pre>Router(config-if) ip address 10.1.1.1 255.255.255.0</pre>	Sets a primary IP address for an interface.
Step 11	<p>no shutdown</p> <p>Example:</p> <pre>Router(config-if)# no shutdown</pre>	Restarts a disabled interface.

Examples

The following example shows how to configure interfaces “ethernet0” and “ethernet1” in a bridge group. These interfaces are associated with the BVI interface “BVI1,” which can be reached from any host on either of the interfaces through the IP address 10.1.1.1.

```
Router(config)# bridge 1 protocol ieee
Router(config)# interface ethernet0
Router(config-if)# bridge-group 1
Router(config-if)# interface ethernet1
Router(config-if)# bridge-group 1
Router(config-if)# exit
! Configure the BVI.
Router(config)# bridge irb
Router(config)# bridge 1 route ip
Router(config)# interface BVI1
Router(config-if)# ip address 10.1.1.1 255.255.255.0
Router(config-if)# no shutdown
```

- [Troubleshooting Tips, page 27](#)
- [What to Do Next, page 28](#)

Troubleshooting Tips

To display the status of each bridge group, use the **show bridge-group** command or to display entries in the bridge table, use the **show bridge** command.

What to Do Next

After you have configured the bridge group, you must configure Cisco IOS IPS as shown in one of the following Cisco IOS IPS tasks, as appropriate to your network needs.

Installing Cisco IOS IPS on a New Router

Perform this task to install the latest Cisco IOS IPS signatures on a router for the first time.

Perform this task to install the default, built-in signatures or the SDF called “attack-drop.sdf”--but not both. If you want to merge the two signature files, you must load the default, built-in signatures as described in this task. Then, you can merge the default signatures with the attack-drop.sdf file as described in the task “Merging Built-In Signatures with the attack-drop.sdf File.”



Note

Installing the signatures provided in flash is the recommended method in Cisco IOS Release 12.3(8)T for IPS attack mitigation.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **ip ips sdf location *url***
4. **ip ips name *ips-name* [*list acl*]**
5. **ip ips signature *signature-id* [*:sub-signature-id*] { **delete** | **disable** | **list *acl-list*** }**
6. **ip ips deny-action *ips-interface***
7. **interface *type number***
8. **ip ips *ips-name* { **in** | **out** }**
9. **exit**
10. **show ip ips configuration**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Router> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: Router# configure terminal	Enters global configuration mode.

	Command or Action	Purpose
Step 3	<p>ip ips sdf location <i>url</i></p> <p>Example: Router(config)# ip ips sdf location disk2:attack-drop.sdf</p>	<p>(Optional) Specifies the location in which the router loads the SDF, “attack-drop.sdf.”</p> <p>Note If this command is not issued, the router loads the default, built-in signatures.</p>
Step 4	<p>ip ips name ips-name [list <i>acl</i>]</p> <p>Example: Router(config)# ip ips name MYIPS</p>	<p>Creates an IPS rule.</p> <p>Note Prior to Cisco IOS Release 12.3(8)T, only standard, numbered ACLs were supported.</p>
Step 5	<p>ip ips signature <i>signature-id</i> [:sub-signature-id] {delete disable list <i>acl-list</i>}</p> <p>Example: Router(config)# ip ips signature 1000 disable</p>	<p>(Optional) Attaches a policy to a given signature.</p>
Step 6	<p>ip ips deny-action ips-interface</p> <p>Example: Router(config)# ip ips deny-action ips- interface</p>	<p>(Optional) Creates an ACL filter for the deny actions (denyFlowInline and denyConnectionInline) on the IPS interface rather than the ingress interface.</p> <p>Note You should configure this command only if at least one signature is configured to use the supported deny actions, if the input interface is configured for load balancing, and if IPS is configured on the output interface.</p>
Step 7	<p>interface <i>type number</i></p> <p>Example: Router(config)# interface GigabitEthernet 0/1</p>	<p>Configures an interface type and enters interface configuration mode.</p>
Step 8	<p>ip ips <i>ips-name</i> {in out}</p> <p>Example: Router(config-if)# ip ips MYIPS in</p>	<p>Applies an IPS rule at an interface and automatically loads the signatures and builds the signature engines.</p> <p>Note Whenever signatures are replaced or merged, the router prompt is suspended while the signature engines for the newly added or merged signatures are being built. The router prompt is available again after the engines are built. Depending on your platform and how many signatures are being loaded, building the engine can take up to several seconds. It is recommended that you enable logging messages to monitor the engine building status.</p>

	Command or Action	Purpose
Step 9	exit Example: <pre>Router(config-if)# exit</pre> Example: <pre>Router(config)# exit</pre>	Exits interface and global configuration modes.
Step 10	show ip ips configuration Example: <pre>Router# show ip ips configuration</pre>	(Optional) Verifies that Cisco IOS IPS is properly configured.

Upgrading to the Latest Cisco IOS IPS Signature Definition File (SDF)

Perform this task to replace the existing signatures on your router with the latest IPS signature file, attack-drop.sdf.



Note

The latest IPS image reads and converts all commands that begin with the words “ip audit” to “ip ips.” For example, the **ip audit name** command becomes the **ip ips name** command. Although IPS accepts the **audit** keyword, it generates the **ips** keyword when you show the configuration. Also, if you issue the help character (?), the CLI displays the **ips** keyword instead of the **audit** keyword, and the Tab key used for command completion does not recognize the **audit** keyword.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **ip ips name *ips-name***
4. **ip ips sdf location *url***
5. **no ip ips location in builtin**
6. **ip ips fail closed**
7. **ip ips deny-action ips-interface**
8. **interface *type number***
9. **ip ips *ips-name* {in | out} [*list acl*]**
10. **exit**
11. **show ip ips configuration**
12. **show ip ips signatures [*detailed*]**

DETAILED STEPS

	Command or Action	Purpose
Step 1	<p>enable</p> <p>Example: Device> enable</p>	<p>Enables privileged EXEC mode.</p> <ul style="list-style-type: none"> Enter your password if prompted.
Step 2	<p>configure terminal</p> <p>Example: Router# configure terminal</p>	<p>Enters global configuration mode.</p>
Step 3	<p>ip ips name <i>ips-name</i></p> <p>Example: Router(config)# ip ips name MIPS</p>	<p>Creates an IPS rule.</p>
Step 4	<p>ip ips sdf location <i>url</i></p> <p>Example: Router(config)# ip ips sdf location disk2:attack-drop.sdf</p>	<p>(Optional) Specifies the location where the router loads the SDF. If this command is not issued, the router loads the default SDF.</p>
Step 5	<p>no ip ips location in builtin</p> <p>Example: Router(config)# no ip ips location in builtin</p>	<p>(Optional) Instructs the router not load the built-in signatures if it cannot find the specified signature file. If this command is not issued, the router loads the built-in signatures if the SDF is not found.</p> <p>Caution If this command is issued and IPS fails to load the SDF, an error message is received stating that IPS is completely disabled.</p>
Step 6	<p>ip ips fail closed</p> <p>Example: Router(config)# ip ips fail closed</p>	<p>(Optional) Instructs the router to drop all packets until the signature engine is built and ready to scan traffic. If this command is issued, one of the following scenarios occurs:</p> <ul style="list-style-type: none"> If IPS fails to load the SDF, all packets are dropped--unless the user specifies an ACL for packets to send to IPS. If IPS successfully loads the SDF but fails to build a signature engine, all packets that are destined for that engine is dropped. <p>If this command is not issued, all packets are passed without scanning if the signature engine fails to build.</p>

Command or Action	Purpose
<p>Step 7 <code>ip ips deny-action ips-interface</code></p> <p>Example: <pre>Router(config)# ip ips deny-action ips-interface</pre></p>	<p>(Optional) Creates an ACL filter for the deny actions (denyFlowInline and denyConnectionInline) on the IPS interface rather than the ingress interface.</p> <p>Note You should configure this command only if at least one signature is configured to use the supported deny actions, if the input interface is configured for load balancing, and if IPS is configured on the output interface.</p>
<p>Step 8 <code>interface type number</code></p> <p>Example: <pre>Router(config)# interface GigabitEthernet0/1</pre></p>	<p>Configures an interface type and enters interface configuration mode.</p>
<p>Step 9 <code>ip ips ips-name {in out} [list acl]</code></p> <p>Example: <pre>Router(config-if)# ip ips MYIPS in</pre></p>	<p>Applies an IPS rule at an interface and automatically loads the signatures and builds the signature engines.</p> <ul style="list-style-type: none"> • list acl--Packets that are permitted through a specified ACL are scanned by IPS. <p>Note Whenever signatures are replaced or merged, the router prompt is suspended while the signature engines for the newly added or merged signatures are being built. The router prompt is available again after the engines are built.</p>
<p>Step 10 <code>exit</code></p> <p>Example: <pre>Router(config)# exit</pre></p>	<p>Exits global configuration mode.</p>
<p>Step 11 <code>show ip ips configuration</code></p> <p>Example: <pre>Router# show ip ips configuration</pre></p>	<p>(Optional) Verifies that Cisco IOS IPS is properly configured.</p>
<p>Step 12 <code>show ip ips signatures [detailed]</code></p> <p>Example: <pre>Router# show ip ips signatures</pre></p>	<p>(Optional) Verifies signature configuration, such as signatures that have been disabled.</p>

Merging Built-In Signatures with the attack-drop.sdf File

You may want to merge the built-in signatures with the attack-drop.sdf file if the built-in signatures are not providing your network with adequate protection from security threats. Perform this task to add the SDF and to change default parameters for a specific signature within the SDF or signature engine.

Before you can merge the attack-drop.sdf file with the built-in signatures, you should already have the built-in signatures loaded onto the router as described in "Installing Cisco IOS IPS on a New Router".

SUMMARY STEPS

1. enable
2. configure terminal
3. no ip ips location in builtin
4. ip ips fail closed
5. exit
6. copy [/erase] url ips-sdf
7. copy ips-sdf url
8. configure terminal
9. ip ips signature signature-id [:sub-signature-id] {delete | disable | list acl-list}
 - list acl --Packets that are permitted through a specified ACL is scanned by IPS.
10. ip ips sdf location url
11. ip ips deny-action ips-interface
12. interface type name
13. ip ips ips-name {in | out}
14. end
15. show ip ips signatures [detailed]

DETAILED STEPS

	Command or Action	Purpose
Step 1	<p>enable</p> <p>Example: Router> enable</p>	<p>Enables privileged EXEC mode.</p> <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	<p>configure terminal</p> <p>Example: Router# configure terminal</p>	<p>Enters global configuration mode.</p>
Step 3	<p>no ip ips location in builtin</p> <p>Example: Router(config)# no ip ips location in builtin</p>	<p>(Optional) Instructs the router not to load the built-in signatures if it cannot find the specified signature file.</p> <p>If this command is not issued, the router loads the built-in signatures if the SDF is not found.</p> <p>Caution If this command is issued and IPS fails to load the SDF, an error message is received stating that IPS is completely disabled.</p>

Command or Action	Purpose
<p>Step 4 <code>ip ips fail closed</code></p> <p>Example: Router(config)# ip ips fail closed</p>	<p>(Optional) Instructs the router to drop all packets until the signature engine is built and ready to scan traffic.</p> <p>If this command is issued, one of the following scenarios occurs:</p> <ul style="list-style-type: none"> • If IPS fails to load the SDF, all packets are dropped--unless the user specifies an ACL for packets to send to IPS. • If IPS successfully loads the SDF but fails to build a signature engine, all packets that are destined for that engine is dropped. <p>If this command is not issued, all packets are passed without scanning if the signature engine fails to build.</p>
<p>Step 5 <code>exit</code></p> <p>Example: Router(config)# exit</p>	<p>Exits global configuration mode.</p>
<p>Step 6 <code>copy [/erase] url ips-sdf</code></p> <p>Example: Router# copy disk2:attack-drop.sdf ips-sdf</p>	<p>Loads the SDF in the router.</p> <p>The SDF merges with the signatures that are already loaded in the router, unless the /erase keyword is issued. The /erase keyword replaces the built-in signatures with the SDF.</p> <p>Note The SDF location is not saved in the configuration. The next time the router is reloaded, it refers to a previously specified SDF location in the configuration or it loads the built-in signatures.</p> <p>Note Whenever signatures are replaced or merged, the router prompt is suspended while the signature engines for the newly added or merged signatures are being built. The router prompt is available again after the engines are built. Depending on your platform and how many signatures are being loaded, building the engine can take up to several seconds. It is recommended that you enable logging messages to monitor the engine building status.</p>
<p>Step 7 <code>copy ips-sdf url</code></p> <p>Example: Router# copy ips-sdf disk2:my-signatures.sdf</p>	<p>Saves the SDF that was loaded in the previous step to a specified location. The SDF location is not be saved unless this command is issued.</p>
<p>Step 8 <code>configure terminal</code></p> <p>Example: Router# configure terminal</p>	<p>Enters global configuration mode.</p>

Command or Action	Purpose
<p>Step 9 <code>ip ips signature signature-id [:sub-signature-id] {delete disable list acl-list}</code></p> <ul style="list-style-type: none"> • list acl --Packets that are permitted through a specified ACL is scanned by IPS. <p>Example: <pre>Router(config)# ip ips signature 1107 disable</pre></p>	<p>(Optional) Instructs the router to scan for the specified signature but not take any action if the signature is detected.</p>
<p>Step 10 <code>ip ips sdf location url</code></p> <p>Example: <pre>Router(config)# ip ips sdf location disk2:my-signatures.sdf</pre></p>	<p>Configures the router to initialize the new SDF.</p>
<p>Step 11 <code>ip ips deny-action ips-interface</code></p> <p>Example: <pre>Router(config)# ip ips deny-action ips-interface</pre></p>	<p>(Optional) Creates an ACL filter for the deny actions (denyFlowInline and denyConnectionInline) on the IPS interface rather than the ingress interface.</p> <p>Note You should configure this command only if at least one signature is configured to use the supported deny actions, if the input interface is configured for load balancing, and if IPS is configured on the output interface.</p>
<p>Step 12 <code>interface type name</code></p> <p>Example: <pre>Router(config)# interface GigabitEthernet0/1</pre></p>	<p>Configures an interface type and enters interface configuration mode.</p>
<p>Step 13 <code>ip ips ips-name {in out}</code></p> <p>Example: <pre>Router(config-if)# ip ips MYIPS in</pre></p>	<p>Applies an IPS rule at an interface and reloads the router and reinitializes Cisco IOS IPS.</p> <p>Note The router prompt disappears while the signatures are loading and the signature engines are building. The router prompt reappears after the signatures have been loaded and the signature engines have been built.</p>
<p>Step 14 <code>end</code></p> <p>Example: <pre>Router(config-if)# end</pre></p>	<p>Exits interface configuration mode.</p>

Command or Action	Purpose
Step 15 <code>show ip ips signatures [detailed]</code> Example: <pre>Router# show ip ips signatures</pre>	(Optional) Verifies signature configuration, such as signatures that have been disabled or marked for deletion.

Monitoring Cisco IOS IPS Signatures through Syslog Messages or SDEE

To use SDEE, the HTTP server must be enabled (through the `ip http server` command). If the HTTP server is not enabled, the router cannot respond to the SDEE clients because it cannot not “see” the requests.

SUMMARY STEPS

1. `enable`
2. `configure terminal`
3. `ip ips notify sdee`
4. `ip sdee events events`
5. `ip sdee subscriptions subscriptions`
6. `exit`
7. `show ip sdee [alerts | all | errors | events | configuration | status | subscriptions]`

DETAILED STEPS

Command or Action	Purpose
Step 1 <code>enable</code> Example: <pre>Router> enable</pre>	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2 <code>configure terminal</code> Example: <pre>Router# configure terminal</pre>	Enters global configuration mode.
Step 3 <code>ip ips notify sdee</code> Example: <pre>Router(config)# ip ips notify sdee</pre>	Enables SDEE event notification on a router.

Command or Action	Purpose
<p>Step 4 <code>ip sdee events <i>events</i></code></p> <p>Example: <pre>Router(config)# ip sdee events 500</pre></p>	<p>(Optional) Sets the maximum number of SDEE events that can be stored in the event buffer. Maximum value: 1000 events.</p> <p>Note By default, 200 events can be stored in the buffer when SDEE is enabled. When SDEE is disabled, all stored events are lost; a new buffer is allocated when the notifications are reenabled.</p>
<p>Step 5 <code>ip sdee subscriptions <i>subscriptions</i></code></p> <p>Example: <pre>Router(config)# ip sdee subscriptions 1</pre></p>	<p>(Optional) Sets the maximum number of SDEE subscriptions that can be open simultaneously. Valid value ranges from 1 to 3.</p>
<p>Step 6 <code>exit</code></p> <p>Example: <pre>Router(config)# exit</pre></p>	<p>Exits global configuration mode.</p>
<p>Step 7 <code>show ip sdee [alerts all errors events configuration status subscriptions]</code></p> <p>Example: <pre>Router# show ip sdee configuration</pre></p>	<p>(Optional) Verifies SDEE configuration information and notification functionality.</p>

- [Troubleshooting Tips, page 37](#)

Troubleshooting Tips

To print out new SDEE alerts on the router console, issue the **debug ip sdee** command.

To clear the event buffer or SDEE subscriptions from the router (which helps with error recovery), issue the **clear ip sdee** command.

Troubleshooting Cisco IOS IPS

- [Interpreting Cisco IOS IPS System Messages, page 37](#)
- [Conditions of an SME Build Failure, page 39](#)

Interpreting Cisco IOS IPS System Messages

Table 4 Cisco IOS IPS System Alarm, Status and Error Messages

System Message	Description
Alarm Messages	
<code>%IPS-4-SIGNATURE:Sig:1107 Subsig:0 Sev:2 RFC1918 address [192.168.121.1:137 -> 192.168.121.255:137]</code>	An IPS signature has been triggered.
<code>%IPS-5-SIGNATURE:Sig:1107 Subsig:0 Global Summary:50 alarms in this interval</code>	A flood of the specified IPS signature has been seen and summarized. (For example, signature 1107 has been seen 50 times.)
Status Messages	
<code>%IPS-6-ENGINE_READY:SERVICE.HTTP - 183136 ms - packets for this engine will be scanned</code>	An IPS signature engine has been built and is ready to scan packets.
<code>%IPS-6-ENGINE_BUILD_SKIPPED:STRING.UDP - there are no new signature definitions for this engine</code>	There are not any signature definitions or changes to the existing signature definitions of an IPS signature engine, and the engine does not have to be rebuilt.
<code>%IPS-5-PACKET_DROP:SERVICE.DNS - packets dropped while engine is building</code>	Packets are being dropped because the specified IPS module is not functioning and the ip ips fail closed command is configured. The message is rate limited to 1 message per 60 seconds.
<code>%IPS-5-PACKET_UNSCANNED:SERVICE.DNS - packets passed unscanned while engine is building</code>	Packets are passing through the network but are not being scanned because the specified IPS module is not functioning and the ip ips fail closed command is not configured. The message is rate limited to 1 message per 60 seconds.
<code>%IPS-6-SDF_LOAD_SUCCESS:SDF loaded successfully from flash:sdf_8http.xml</code>	An SDF is successfully loaded from a given location.
Error Messages	
<code>%IPS-3-BUILTIN_SIGS:Configured to load builtin signatures %IPS-3-BUILTIN_SIGS:Not Configured to load builtin signatures %IPS-3-BUILTIN_SIGS:Failed to load builtin signatures</code>	One of these three messages can be displayed when IPS loads the built-in signatures.

System Message	Description
<code>%IPS-5-ENGINE_UNKNOWN: SERVICE.GENERIC - unknown engine encountered while parsing SDF</code>	<p>The router has encountered an unknown and unsupported signature engine while parsing the SDF.</p> <p>To prevent this message from being generated again, ensure that the SDF being loaded on the router does not contain any engines that are not supported by IPS.</p>
<code>%IPS-5-UNSUPPORTED_PARAM: SERVICE.RPC 6275:1 isSweep=False - bad parameter - removing parameter</code>	<p>The router has encountered an unsupported parameter while parsing the SDF.</p> <p>The signature is deleted if the unsupported parameter is required for the signature. The parameter is removed from the signature if it is not required.</p> <p>To prevent this message from being generated again, ensure that the SDF being loaded on the router does not contain any parameters that are not supported by IPS.</p>
<code>%IPS-3-ENGINE_BUILD_FAILED: SERVICE.HTTP - 158560 ms - engine build</code>	<p>One of the signature engines fails to build after an SDF is loaded. A message is sent for each engine that fails.</p> <p>An engine typically fails to build because of low memory, so increasing router memory can alleviate the problem. Also, try to load the SDF immediately after a route reboots, which is when system resources are available.</p>
<code>%IPS-4-SDF_PARSE_FAILED: not well-formed (invalid token) at Line 1 Col 0 Byte 0 Len 1006</code>	<p>An SDF has not parsed correctly. The SDF might have been corrupt.</p>
<code>%IPS-4-SDF_LOAD_FAILED: failed to parse SDF from tftp://tftp-server/sdf.xml</code>	<p>An SDF fails to load. The SDF may fail for any of the following reasons:</p> <ul style="list-style-type: none"> • Fails to load if it resides on a network server that cannot be reached • Does not have the correct read permissions
<code>%IPS-2-DISABLED: IPS removed from all interfaces - IPS disabled</code>	<p>IPS has been disabled. This message indicates why IPS has been disabled.</p>

Conditions of an SME Build Failure

Sometimes an SME that is being built fails. The SME can fail because it is attempting to load a corrupted SDF file or it exceeds memory limitations of the router. If a failure occurs, Cisco IOS IPS is designed to handle it. Possible failures are as follows:

- By default, IPS is designed to “fail open,” which means that if an SME does not build, all packets that are destined for that particular engine passes traffic without scanning.
- If IPS cannot load the attack-drop.sdf file onto a router, the router reverts to the previously loaded available signatures. (In most cases, the previously loaded signatures are the Cisco IOS built-in signatures.)
- If an engine build fails when you are merging the attack-drop.sdf file with the built-in signatures, IPS reverts, by default, to the previously available engine (or engines).

The default behavior for engine failure allows for packets to be passed unscanned. To prevent traffic from being passed unscanned, issue the **ip ips fail closed** command, which forces the router to drop all packets if an SME build fails.

**Note**

If a signature or a signature parameter is not supported, Cisco IOS prints a syslog message, indicating that the signature or parameter is not supported.

Configuration Examples

- [Loading the Default Signatures Example, page 40](#)
- [Loading the attack-drop.sdf Example, page 40](#)
- [Merging the attack-drop.sdf File with the Default Built-in Signatures Example, page 41](#)

Loading the Default Signatures Example

The following example shows the Cisco IOS IPS commands required to load the default, built-in signatures. Note that a configuration option for specifying an SDF location is not necessary; built-in signatures reside statically in Cisco IOS.

```
!
ip ips po max-events 100
ip ips name MYIPS
!
interface GigabitEthernet0/1
 ip address 10.1.1.16 255.255.255.0
 ip ips MYIPS in
 duplex full
 speed 100
 media-type rj45
 no negotiation auto
!
```

Loading the attack-drop.sdf Example

The following example shows the basic configuration necessary to load the attack-drop.sdf file onto a router running Cisco IOS IPS. Note that the configuration is almost the same as loading the default signatures onto a router, except for the **ip ips sdf location** command, which specifies the attack-drop.sdf file.

```
!
ip ips sdf location disk2:attack-drop.sdf
ip ips name MYIPS
!
```

```

interface GigabitEthernet0/1
 ip address 10.1.1.16 255.255.255.0
 ip ips MYIPS in
 duplex full
 speed 100
 media-type rj45
 no negotiation auto
!
```

Merging the attack-drop.sdf File with the Default Built-in Signatures Example

The following example shows how to configure the router to load and merge the attack-drop.sdf file with the default signatures. After you have merged the two files, it is recommended that you copy the newly merged signatures to a separate file. The router can then be reloaded (through the **reload** command) or reinitialized to recognize the newly merged file (as shown the following example).

```

!
ip ips name MYIPS
!
interface GigabitEthernet0/1
 ip address 10.1.1.16 255.255.255.0
 ip ips MYIPS in
 duplex full
 speed 100
 media-type rj45
 no negotiation auto
!
!
! Merge the flash-based SDF (attack-drop.sdf) with the built-in signatures.
Router# copy disk2:attack-drop.sdf ips-sdf
! Save the newly merged signatures to a separate file.
Router# copy ips-sdf disk2:my-signatures.sdf
!
! Configure the router to use the new file, my-signatures.sdf
Router# configure terminal
Router(config)# ip ips sdf location disk2:my-signatures.sdf
! Reinitialize the IPS by removing the IPS rule set and reapplying the rule set.
Router(config-if)# interface gig 0/1
Router(config-if)# no ip ips MYIPS in
!
*Apr 8 14:05:38.243:%IPS-2-DISABLED:IPS removed from all interfaces - IPS disabled
!
Router(config-if)# ip ips MYIPS in
!
Router(config-if)# exit
```

Additional References

Related Documents

Related Topic	Document Title
Cisco IOS commands	Cisco IOS Master Commands List, All Releases

Related Topic	Document Title
Security commands	<ul style="list-style-type: none"> • Cisco IOS Security Command Reference: Commands A to C • Cisco IOS Security Command Reference: Commands D to L • Cisco IOS Security Command Reference: Commands M to R • Cisco IOS Security Command Reference: Commands S to Z

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 Configuring Cisco IOS IPS

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 Configuring Cisco IOS IPS**

Feature Name	Software Releases	Feature Configuration Information
Cisco IOS Intrusion Prevention System (IPS)	12.3(8)T	<p>The Cisco IOS Intrusion Prevention System (IPS) uses the following methods to protect a network from internal and external attacks and threats:</p> <ul style="list-style-type: none"> • IPS signatures are dynamically updated and posted to Cisco.com on a regular basis so that customers can access signatures that help protect their network from the latest known network attacks. • A Parallel Signature Scanning Engine is used to scan for multiple patterns within a signature microengine (SME) at any given time. IPS signatures are no longer scanned on a serial basis. • Cisco IOS IPS supports both named and numbered extended access control lists (ACLs). <p>The following commands were introduced by this feature: clear ip sdee, copy ips-sdf, debug ip ips, debug ip sdee, ip ips fail closed, ip ips sdf location, ip sdee events, ip sdee subscriptions, no ip ips sdf builtin, show ip sdee</p>

Feature Name	Software Releases	Feature Configuration Information
Cisco IOS Intrusion Prevention System (IPS)	12.3(14)T	<ul style="list-style-type: none"> • Supports access to more recent virus and attack signatures with the addition of three more SMEs--STRING.TCP, STRING.ICMP, and STRING.UDP. • Intelligent and local shunning is supported, which allows Cisco IOS IPS to shun offending traffic on the same router that Cisco IOS IPS is configured. • The ip ips deny-action ips-interface command was added, which allows users to choose between two available ACL filter settings for detecting offending packets. <p>Support for the Post Office Protocol was deprecated and the following commands were removed from the Cisco IOS software: ip ips po local, ip ips po max-events, ip ips po protected, and ip ips po remote.</p>
Transparent Cisco IOS IPS	12.4(2)T	<p>Support was added for Layer 2 transparent bridging for Cisco IOS IPS. Transparent Cisco IOS IPS eases certain network and management deployment by allowing users to “drop” a device running Cisco IOS IPS in front of their existing network without changing the statically defined IP addresses of their network-connected devices. Thus, users can allow selected devices from a subnet to traverse the IPS while access to other devices on the same subnet is denied.</p> <p>No commands were introduced or modified for this feature.</p>

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

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



Cisco IOS IPS 5.x Signature Format Support and Usability Enhancements

The Cisco IOS Intrusion Prevention System (IPS) acts as an in-line intrusion prevention sensor that scans packets and sessions as they flow through the router to match any Cisco IOS IPS 5.x signature. These signatures are defined in Extensible Markup Language (XML) format and provide the following functionality:

- Automatic signature updates from local servers. Network administrators can either preserve the user's current configuration of signature actions or override the user's current configuration of signature actions with the current IPS configuration.
- Top-level signature categories to classify signatures for easy grouping and tuning. Group-wide parameters, such as signature event action, can be applied to a group through CLI, so the user does not have to modify each individual signature.
- Encrypted (NDA) signature support.
- Direct Download from CCO capability in IOS IPS feature allows an administrator to use the CLI to specify, download and upgrade to new signatures posted for the IOS directly from Cisco.com. An administrator can also configure the router through the CLI to receive future periodic signature downloads automatically to eliminate the manual maintenance efforts and costs of changing or tuning IPS signatures whenever a new update is posted.
- [Finding Feature Information, page 47](#)
- [Prerequisites for Cisco IOS IPS 5.x Signature Format Support and Usability Enhancements, page 48](#)
- [Restrictions for Cisco IOS IPS 5.x Signature Format Support and Usability Enhancements, page 49](#)
- [Information About Cisco IOS IPS 5.x Signature Format Support and Usability Enhancements, page 50](#)
- [How to Use Cisco IOS IPS 5.x Signature Format Support and Usability Enhancements, page 52](#)
- [Configuration Examples for Cisco IOS IPS 5.x Signature Format Support and Usability Enhancements, page 73](#)
- [Additional References, page 76](#)
- [Feature Information for Cisco IOS IPS 5.x Signature Format Support and Usability Enhancements, page 77](#)

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 Cisco IOS IPS 5.x Signature Format Support and Usability Enhancements

System and Image Requirements for Cisco IOS IPS 5.x

- Cisco IOS IPS signatures. See the “Cisco IOS Signature Package” section for more information.
- Cisco IOS IPS system requirements depend on the type of deployment, the bandwidth requirements, and security requirements. The larger the number of signatures, the larger the amount of memory consumed.
- You must generate a RSA crypto key and load the public signature on your router for signature decryption.

This following cisco public key configuration can be cut and pasted directly into your router configuration:

```
crypto key pubkey-chain rsa
named-key realm-cisco.pub signature
key-string
30820122 300D0609 2A864886 F70D0101 01050003 82010F00 3082010A 02820101
00C19E93 A8AF124A D6CC7A24 5097A975 206BE3A2 06FBA13F 6F12CB5B 4E441F16
17E630D5 C02AC252 912BE27F 37FDD9C8 11FC7AF7 DCDD81D9 43CDABC3 6007D128
B199ABCB D34ED0F9 085FADC1 359C189E F30AF10A C0EFB624 7E0764BF 3E53053E
5B2146A9 D7A5EDE3 0298AF03 DED7A5B8 9479039D 20F30663 9AC64B93 C0112A35
FE3F0C87 89BCB7BB 994AE74C FA9E481D F65875D6 85EAF974 6D9CC8E3 F0B08B85
50437722 FFBEE85B9 5E4189FF CC189CB9 69C46F9C A84DFBA5 7A0AF99E AD768C36
006CF498 079F88F8 A3B3FB1F 9FB7B3CB 5539E1D1 9693CCBB 551F78D2 892356AE
2F56D826 8918EF3C 80CA4F4D 87BFCA3B BFF668E9 689782A5 CF31CB6E B4B094D3
F3020301 0001
quit
```



Note

You can also access the public key (realm-cisco.pub.key.txt) configuration at the following URL: <http://www.cisco.com/cgi-bin/tablebuild.pl/ios-v5sigup>. Ensure that you have your Cisco userid, and password to access this URL.

- You must load one of the following images on your router to install Cisco IOS IPS 5.x: `adventerprisek9`, `advsecurityk9`, and `advipservicesk9`.



Note

To check the current system version, use the `show subsys name ips` command. IPS 4.x uses a version format of 2.xxx.xxx; IPS 5.x uses a version format of 3.xxx.xxx.

Upgrading from Cisco IOS IPS 4.x to Cisco IOS IPS 5.x Signatures

Cisco IOS IPS 5.x format signatures are not backward compatible with Cisco IOS IPS 4.x. You must reconfigure your Cisco IOS IPS features for use with the IPS 5.x signature format command-line interface (CLI) and features.

When reconfiguring Cisco IOS IPS on a router to convert to the 5.x signature format, you must have the following Cisco IOS IPS 4.x information:

- Cisco IOS IPS rule name (which was specified through the **ip ips name** *ips-name* command)
- Interfaces for which the Cisco IOS IPS rule has been applied

To gather this information, issue the **show ip ips configuration** command, which displays a copy of the existing output.

```
Router# show ip ips configuration

Configured SDF Locations:
disk2:my-signatures.sdf
Builtin signatures are enabled but not loaded
Last successful SDF load time: 05:31:54 MST Sep 20 2003
IPS fail closed is disabled
Fastpath ips is enabled
Quick run mode is enabled
Event notification through syslog is enabled
Event notification through SDEE is enabled
Total Active Signatures: 13
Total Inactive Signatures: 0
Signature 50000:0 disable
Signature 50000:1 disable
Signature 50000:2 disable
IPS Rule Configuration
IPS name MYIPS
Interface Configuration
Interface GigabitEthernet0/1
Inbound IPS rule is MYIPS
Outgoing IPS rule is not set
```

Direct Download from CCO capability in IOS IPS Support

The router must have access to Cisco.com in order for IPS signatures to be upgraded directly from Cisco.com. If the router does not have access to Cisco.com, then signature file updates can still be able to be retrieved from a local server.

Restrictions for Cisco IOS IPS 5.x Signature Format Support and Usability Enhancements



Danger

Do not enable all IPS signatures at once. The router may not be able to compile all of the signatures, resulting in high CPU and memory usage, degraded performance, or a system reload.

Backward Compatibility

Cisco IOS IPS 5.x format signatures are not backward compatible with Cisco IOS IPS 4.x SDFs.

Cisco 870 Series Platform Support

The 870 series platform with Cisco IOS IPS in Cisco IOS Release 12.4(11)T may experience lower performance relative to previous releases (CSCsg57228). The Cisco IOS IPS performance on the 870 series platform is enhanced in a later 12.4(11)T image rebuild.

On the 870 series platform, Cisco IOS IPS is supported only on the adv-ipservices and the adv-enterprise images. Cisco IOS IPS is the same on both images.

Information About Cisco IOS IPS 5.x Signature Format Support and Usability Enhancements

- [Cisco IOS IPS Overview, page 50](#)
- [Cisco IOS IPS Signature Scanning with Lightweight Signatures, page 50](#)
- [Signature Categories, page 50](#)
- [Cisco IOS IPS Signature Package, page 51](#)
- [Preserving Configured Signature Tunings on the Local Router, page 51](#)

Cisco IOS IPS Overview

The Cisco IOS IPS acts as an in-line intrusion prevention sensor, watching packets and sessions as they flow through the router and scanning each packet to match any of the Cisco IOS IPS signatures. When it detects suspicious activity, it responds before network security can be compromised and logs the event through Cisco IOS syslog messages or Security Device Event Exchange (SDEE). The network administrator can configure Cisco IOS IPS to choose the appropriate response to various threats. The Signature Event Action Processor (SEAP) can dynamically control actions that are to be taken by a signature event on the basis of parameters such as fidelity, severity, or target value rating. These parameters have default values but can also be configured through CLI. When packets in a session match a signature, Cisco IOS IPS can take any of the following actions, as appropriate:

- Send an alarm to a syslog server or a centralized management interface
- Drop the packet
- Reset the connection
- Deny traffic from the source IP address of the attacker for a specified amount of time
- Deny traffic on the connection for which the signature was seen for a specified amount of time

Cisco developed its Cisco IOS software-based intrusion-prevention capabilities and Cisco IOS Firewall with flexibility in mind, so that individual signatures could be disabled in case of false positives. Generally, it is preferable to enable both the firewall and Cisco IOS IPS to support network security policies. However, each of these features may be enabled independently and on different router interfaces.

Cisco IOS IPS Signature Scanning with Lightweight Signatures

The addition of Cisco IOS IPS signature scanning with lightweight signatures in Cisco IOS Release 15.0(1)M is an enhancement to Cisco IOS IPS that allows loading of larger signatures sets, without consuming significant additional memory or reducing the memory consumed by an existing signature set, by loading equivalent lighter-weight signatures. These signatures are referred to as lightweight signatures.

For the signatures made obsolete by new lightweight signatures from new signature scanning engines, Cisco IOS provides the **ip ips inherit-obsolete-tunings** command to apply common parameters the user may have changed (customized) in the old signature file (delta.xml), to the equivalent new signature file. For more information, see "Enabling Signature Tunings Inheritance".

Signature Categories

Cisco IPS appliances and Cisco IOS IPS with Cisco 5.x format signatures operate with signature categories. All signatures are pregrouped into categories; the categories are hierarchical. An individual signature can belong to more than one category. Top-level categories help to define general types of signatures.

Subcategories exist beneath each top-level signature category. (For a list of supported top-level categories, use your router CLI help (?).)

- [Router Configuration Files and Signature Event Action Processor \(SEAP\), page 51](#)

Router Configuration Files and Signature Event Action Processor (SEAP)

As of Cisco IOS Release 12.4(11)T, SDFs are no longer used by Cisco IOS IPS. Instead, routers access signature definition information through a directory that contains three configuration files--the default configuration, the delta configuration, and the SEAP configuration. Cisco IOS accesses this directory through the **ip ips config location** command.



Note

You must issue the **ip ips config location** command; otherwise, the configuration files are not saved to any location.

SEAP is the control unit responsible for coordinating the data flow of a signature event. It allows for advanced filtering and signature overrides on the basis of the Event Risk Rating (ERR) feedback. ERR is used to control the level in which a user chooses to take actions in an effort to minimize false positives. Signatures once stored in NVRAM, are now stored in the delta configuration file; thus, support for access control lists (ACLs) is no longer necessary.

- [Additional Risk Rating Algorithms, page 51](#)

Additional Risk Rating Algorithms

The ERR characterizes the risk of an attack and allows users to make decisions on the basis of the risk control signature event actions. To help further control signature event actions, the following additional rating categories are now supported:

Cisco IOS IPS Signature Package

The latest Cisco IOS IPS signature package can be accessed from Cisco.com using the following URL:
<http://www.cisco.com/cisco/software/release.html?mdfid=281442967&release=S636&reind=AVAILABLE&flowid=4836&softwareid=280775022&rellifecycle=&reltype=latest>



Note

Ensure that you have your Cisco userid, and password to access this URL.

- The IOS-Sxxx-CLI.pkg file is listed on this web page. See the “Configuring Cisco IOS IPS on Your Router” for more information on creating a directory on the router flash directory where the required signature files and configurations are stored. Alternatively, you can use a Cisco USB flash drive connected to the router's USB port to store the signature files and configurations. The USB flash drive needs to remain connected to the router USB port if it is used as the IOS IPS configuration directory location. IOS IPS also supports any IOS File System as its configuration location with proper write access.

Preserving Configured Signature Tunings on the Local Router

Most IPS devices and applications provide either a single default configuration or multiple default configurations. Using one of these default configurations is an ideal starting point for deploying IPS. When

IOS IPS is deployed, parameters such as severity, active status or event actions of certain signatures need to be tuned to meet the requirements of an enterprise network traffic profile.

Once the **ip ips enable-clidelta** command is enabled, a local cli-delta.xmz file is generated containing the local tuning signatures configured through the CLI. The settings in the clidelta.xmz file take precedence when a globally administered delta signature update, contained in the iosips-sig-delta.xmz file, is sent from a central repository and applied to the configuration of the local router. See "Tuning Signatures per Signature ID" for more information about the configuration of this feature.

How to Use Cisco IOS IPS 5.x Signature Format Support and Usability Enhancements

- [Retiring All Signatures and Selecting a Category of Signatures, page 52](#)
- [Configuring Cisco IOS IPS on Your Router, page 54](#)
- [Loading a Signature File into Cisco IOS IPS, page 57](#)
- [Enabling IPS Regex Table Chaining, page 59](#)
- [Tuning Signature Parameters, page 60](#)
- [Enabling Signature Tunings Inheritance, page 65](#)
- [Setting an IPS Memory Threshold, page 66](#)
- [Setting the Target Value Rating, page 67](#)
- [Configuring Signature Auto Updates from a Local Server, page 68](#)
- [Monitoring Cisco IOS IPS Signatures through Syslog Messages or SDEE, page 70](#)

Retiring All Signatures and Selecting a Category of Signatures

Router memory and resource constraints prevent a router from loading all Cisco IOS IPS signatures. Thus, it is recommended that you load only a selected set of signatures that are defined by the categories. Because the categories are applied in a “top-down” order, you should first retire all signatures, followed by “unretiring” specific categories. Retiring signatures enables the router to load information for all signatures, but the router does not build the parallel scanning data structure.

Retired signatures are not scanned by Cisco IOS IPS, so they do not fire alarms. If a signature is irrelevant to your network or if you want to save router memory, you should retire signatures, as appropriate.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **ip ips signature-category**
4. **category** *category* [*subcategory*]
5. **retired** {**true** | **false**}
6. **exit**
7. **category** *category* [*subcategory*]
8. **retired** {**true** | **false**}
9. **exit**

DETAILED STEPS

Command or Action	Purpose
<p>Step 1 <code>enable</code></p> <p>Example: Router> enable</p>	<p>Enables privileged EXEC mode.</p> <ul style="list-style-type: none"> • Enter your password if prompted.
<p>Step 2 <code>configure terminal</code></p> <p>Example: Router# configure terminal</p>	<p>Enters global configuration mode.</p>
<p>Step 3 <code>ip ips signature-category</code></p> <p>Example: Router(config)# ip ips signature-category</p>	<p>Enters enters IPS category configuration mode.</p>
<p>Step 4 <code>category category [subcategory]</code></p> <p>Example: Router(config-ips-category)# category all</p>	<p>Specifies that all categories (and all signatures) are retired in the following step and enters IPS category action configuration mode.</p>
<p>Step 5 <code>retired {true false}</code></p> <p>Example: Router(config-ips-category-action)# retired true</p>	<p>Specifies that the router should retire all categories (and all signatures).</p> <ul style="list-style-type: none"> • true --Retires all signatures within a given category. • false --“Unretires” all signatures within a given category.
<p>Step 6 <code>exit</code></p> <p>Example: Router(config-ips-category-action)# exit</p>	<p>Exits IPS category action configuration mode.</p>
<p>Step 7 <code>category category [subcategory]</code></p> <p>Example: Router(config-ips-category)# category ios_ips basic</p>	<p>Specifies the basic category (and a set of signatures) that are to be “unretired” in the following step.</p>

Command or Action	Purpose
Step 8 <code>retired {true false}</code> Example: <pre>Router(config-ips-category-action)# retired false</pre>	Specifies that all signatures within the basic category are to be unretired; that is, signatures are enabled for the basic category.
Step 9 <code>exit</code> Example: <pre>Router(config-ips-category-action)# exit</pre>	Exits IPS category action and IPS category configuration modes.

- [What to Do Next, page 54](#)

What to Do Next

After you have configured the basic category, you should enable IPS on your router. See "Configuring Cisco IOS IPS on Your Router" for more information.

You can customize (or tune) either the entire category or individual signatures within a category to address the needs of your network. See "Tuning Signature Parameters", for more information.

Configuring Cisco IOS IPS on Your Router

After you have set up a "load definition" for the signatures to be copied to the idconf, you must configure an IPS rule name. Use this task to configure an IPS rule name and start the IPS configuration.

You can also use this task to configure a Cisco IOS IPS signature location, which tells Cisco IOS IPS where to save signature information.

The configuration location is used to restore the IPS configuration in case the router reboots or IPS is disabled or reenabled. Files, such as signature definition, signature-type definitions, and signature category information, are written in XML format, compressed, and saved to the specified IPS signature location.

SUMMARY STEPS

1. `enable`
2. `mkdir flash:/ips5`
3. `configure terminal`
4. `ip ips name ips-name`
5. `ip ips config location url`
6. `interface type name`
7. `ip ips ips-name {in | out}`
8. `exit`
9. `show ip ips configuration`
10. `show ip ips signature count`

DETAILED STEPS

	Command or Action	Purpose
Step 1	<p>enable</p> <p>Example:</p> <pre>Router> enable</pre>	<p>Enables privileged EXEC mode.</p> <ul style="list-style-type: none"> Enter your password if prompted.
Step 2	<p>mkdir flash:/ips5</p> <p>Example:</p> <pre>Router# mkdir flash:/ips5</pre>	<p>Create a directory for which Cisco IOS IPS saves signature information.</p> <p>Note The directory location is specified through the ip ips config location command.</p>
Step 3	<p>configure terminal</p> <p>Example:</p> <pre>Router# configure terminal</pre>	<p>Enters global configuration mode.</p>
Step 4	<p>ip ips name <i>ips-name</i></p> <p>Example:</p> <pre>Router(config)# ip ips name myips</pre>	<p>Creates an IPS rule.</p>
Step 5	<p>ip ips config location <i>url</i></p> <p>Example:</p> <pre>Router(config)# ip ips config location flash:/ips5</pre>	<p>Specifies the location where Cisco IOS IPS saves the signature information, and, if necessary, access the signature configuration information.</p> <p>Note You must specify a location; otherwise, the signatures are not saved.</p> <p>Note If the specified location is a URL, such as an FTP server, the user must have writer privileges.</p>
Step 6	<p>interface type name</p> <p>Example:</p> <pre>Router(config)# interface gigbitEthernet 0/0</pre>	<p>Identifies the interface in which to enable Cisco IOS IPS and enters interface configuration mode.</p>

Command or Action	Purpose
<p>Step 7 <code>ip ips <i>ips-name</i> {in out}</code></p> <p>Example:</p> <pre>Router(config-if)# ip ips MYIPS in</pre>	<p>Applies an IPS rule at an interface and automatically loads the signatures and builds the signature engines.</p> <p>Note Whenever signatures are replaced or merged, the router prompt is suspended while the signature engines for the newly added or merged signatures are being built. The router prompt is available again after the engines are built. Depending on your platform and how many signatures are being loaded, building the engine can take up to several minutes. It is recommended that you enable logging messages to monitor the engine building status.</p>
<p>Step 8 <code>exit</code></p> <p>Example:</p> <pre>Router(config-if)# exit</pre> <p>Example:</p> <pre>Router(config)# exit</pre>	<p>Exits interface and global configuration modes.</p>
<p>Step 9 <code>show ip ips configuration</code></p> <p>Example:</p> <pre>Router# show ip ips configuration</pre>	<p>(Optional) Verifies that Cisco IOS IPS is properly configured.</p>
<p>Step 10 <code>show ip ips signature count</code></p> <p>Example:</p> <pre>Router# show ip ips signature</pre>	<p>(Optional) Verifies the number of signatures that are loaded into each signature micro engine (SME).</p>

Examples

The following sample output displays the number of signatures that have been loaded into each SME:

```
Router# show ip ips signature count
Cisco SDF release version S247.0
Trend SDF release version V1.2
Signature Micro-Engine: multi-string
Total Signatures: 7
Enabled: 7
Retired: 2
Compiled: 5
Signature Micro-Engine: service-http
Total Signatures: 541
Enabled: 284
Retired: 336
Compiled: 205
Signature Micro-Engine: string-tcp
```

```

Total Signatures: 487
Enabled: 332
Retired: 352
Compiled: 135
Signature Micro-Engine: string-udp
Total Signatures: 50
Enabled: 3
Retired: 23
Compiled: 27
Signature Micro-Engine: state
Total Signatures: 26
Enabled: 15
Retired: 23
Compiled: 3
Signature Micro-Engine: atomic-ip
Total Signatures: 140
Enabled: 87
Retired: 93
Compiled: 46
Inactive - invalid params: 1
Signature Micro-Engine: string-icmp
Total Signatures: 2
Enabled: 0
Retired: 1
Compiled: 1
Signature Micro-Engine: service-ftp
Total Signatures: 3
Enabled: 3
Compiled: 3
Signature Micro-Engine: service-rpc (INACTIVE)
Signature Micro-Engine: service-dns
Total Signatures: 1
Enabled: 1
Retired: 1
Signature Micro-Engine: normalizer
Total Signatures: 9
Enabled: 9
Compiled: 9
Total Signatures: 1266
Total Enabled Signatures: 741
Total Retired Signatures: 831
Total Compiled Signatures: 434
Total Signatures with invalid parameters: 1

```

Loading a Signature File into Cisco IOS IPS

Use this task to load signatures into Cisco IOS IPS. You may wish to load new signatures into Cisco IOS IPS if a signature (or signatures) with the current signatures are not providing your network with adequate protection from security threats.

- [Prerequisites, page 57](#)
- [Flexible Signatures Ordered and Incremental, page 57](#)

Prerequisites

You must enable Cisco IOS IPS. See "Configuring Cisco IOS IPS on Your Router" before loading new signatures.

Flexible Signatures Ordered and Incremental

Each signature is compiled incrementally into the scanning tables at the same time. Thus, Cisco IOS IPS can deactivate signatures that fail to compile. (Prior to Cisco IOS Release 12.4(11)T, Cisco IOS IPS deactivated the entire signature microengine (SME) if a single signature failed to compile.)

Signatures are loaded into the scanning table on the basis of importance. Parameters such as signature severity, signature fidelity rating, and time lapsed since signatures were last released allow Cisco IOS IPS

to compile the most important signatures first, followed by less important signatures, thereby, creating a load order and prioritizing which signatures are loaded first.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **ip ips config location *url***
4. **interface *type name***
5. **ip ips *ips-name* {in | out}**
6. **end**
7. **copy *url idconf***

DETAILED STEPS

Command or Action	Purpose
Step 1 enable Example: Router> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2 configure terminal Example: Router# configure terminal	Enters global configuration mode.
Step 3 ip ips config location <i>url</i> Example: Router(config)# ip ips config location flash:/ ips5	Specifies the location where Cisco IOS IPS saves the signature information, and, if necessary, access the signature configuration information.
Step 4 interface <i>type name</i> Example: Router(config)# interface gigbitEthernet 0/0	Identifies the interface in which to enable Cisco IOS IPS.
Step 5 ip ips <i>ips-name</i> {in out} Example: Router(config-if)# ip ips MYIPS in	Applies an IPS rule at an interface and automatically loads the signatures and builds the signature engines.

Command or Action	Purpose
Step 6 <code>end</code> Example: <pre>Router(config-if)# end</pre>	Exits interface configuration mode.
Step 7 <code>copy url idconf</code> Example: <pre>Router# copy tftp://tftp_server/sig.xml idconf</pre>	Loads signatures into Cisco IOS IPS. After the signatures are loaded, all signature information is saved to the location specified through the ip ips config location command.

Enabling IPS Regex Table Chaining

Regex Table Chaining is an enhancement to Cisco IOS IPS in Cisco IOS Release 15.0(1)M, that allows the IPS to chain multiple regular (regex) tables together when signatures are being loaded. This functionality is enabled by the **ip ips memory regex chaining** command. Enabling regex table chaining allows three regex tables to be chained. There is a slight performance degradation in IPS scanning time due to the scanning of multiple tables.

When a user tries to load a specific set of signatures that does not fit using a single table, compilation errors result.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **ip ips memory regex chaining**

DETAILED STEPS

Command or Action	Purpose
Step 1 <code>enable</code> Example: <pre>Router> enable</pre>	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2 <code>configure terminal</code> Example: <pre>Router# configure terminal</pre>	Enters global configuration mode.

Command or Action	Purpose
Step 3 ip ips memory regex chaining Example: Router(config)# ip ips memory regex chaining	Enables IPS regex table chaining when signatures are being loaded.

Tuning Signature Parameters

You can tune signature parameters on the basis of a signature ID (for an individual signature), or you can tune signature parameters on the basis of a category (that is, all signatures that are within a specified category). To tune signature parameters, use the following tasks, as appropriate:



Note

Some changes to the signature definitions are not shown in the run time config because the changes are recorded in the sigdef-delta.xml file, which can be located through the **ip ips config location** command.

- [Tuning Signatures for a Signature ID, page 60](#)
- [Tuning Signatures per Category, page 63](#)

Tuning Signatures for a Signature ID

Use this task to change default signature parameters for a specified signature ID.

SUMMARY STEPS

1. enable
2. configure terminal
3. ip ips enable-clidelta
4. ip ips signature-definition
5. signature *signature-id* [*signature-id*]
6. engine
7. event-action *action*
8. exit
9. alert-severity {high | medium | low | informational}
10. fidelity-rating *rating*
11. status
12. enabled {true | false }
13. exit
14. show ip ips signature
15. show ip ips sig-clidelta

DETAILED STEPS

	Command or Action	Purpose
Step 1	<p>enable</p> <p>Example:</p> <pre>Router> enable</pre>	<p>Enables privileged EXEC mode.</p> <ul style="list-style-type: none"> Enter your password if prompted.
Step 2	<p>configure terminal</p> <p>Example:</p> <pre>Router# configure terminal</pre>	<p>Enters global configuration mode.</p>
Step 3	<p>ip ips enable-clidelta</p> <p>Example:</p> <pre>Router(config)# ip ips enable-clidelta</pre>	<p>(Optional) Enables the signature tuning settings in the clidelta.xml file on the router to take precedence over the signature settings in the IPS iosips-sig-delta.xml file.</p>
Step 4	<p>ip ips signature-definition</p> <p>Example:</p> <pre>Router(config)# ip ips signature-definition</pre>	<p>Enters signature-definition-signature configuration mode.</p>
Step 5	<p>signature <i>signature-id</i> [<i>signature-id</i>]</p> <p>Example:</p> <pre>Router(config-sigdef-sig)# signature 9000:0</pre>	<p>Specifies a signature for which the CLI user tunings are changed and enters signature-definition-action configuration mode.</p>
Step 6	<p>engine</p> <p>Example:</p> <pre>Router(config-sigdef-action)# engine</pre>	<p>(Optional) Enters signature-definition-action-engine configuration mode, which allows you to change router actions for a specified signature.</p>

Command or Action	Purpose
<p>Step 7 <code>event-action <i>action</i></code></p> <p>Example:</p> <pre>Router(config-sigdef-action-engine)# event- action deny-attacker-inline</pre>	<p>Changes router actions for a specified signature.</p> <p>The <i>action</i> argument can be any of the following options:</p> <ul style="list-style-type: none"> • deny-attacker-inline • deny-connection-inline • deny-packet-inline • produce-alert • reset-tcp-connection <p>Note Signature event actions must be entered on a single line.</p> <p>Note You must enter the engine command before issuing this command.</p>
<p>Step 8 <code>exit</code></p> <p>Example:</p> <pre>Router(config-sigdef-action-engine)# exit</pre>	<p>Exits the signature-definition-action-engine configuration mode.</p> <p>This step is required only if the engine and event-action commands are issued.</p>
<p>Step 9 <code>alert-severity {<i>high</i> <i>medium</i> <i>low</i> <i>informational</i>}</code></p> <p>Example:</p> <pre>Router(config-sigdef-action)# alert- severity medium</pre>	<p>(Optional) Changes the alert severity rating for a given signature.</p>
<p>Step 10 <code>fidelity-rating <i>rating</i></code></p> <p>Example:</p> <pre>Router(config-sigdef-action)# fidelity- rating 2</pre>	<p>(Optional) Changes the signature fidelity rating for a given signature.</p>
<p>Step 11 <code>status</code></p> <p>Example:</p> <pre>Router(config-sigdef-action)# status</pre>	<p>(Optional) Enters the signature-definition-status configuration mode, which allows you to change the enabled status of a signature.</p>
<p>Step 12 <code>enabled {<i>true</i> <i>false</i>}</code></p> <p>Example:</p> <pre>Router(config-sigdef-status)# enabled true</pre>	<p>(Optional) Changes the enabled status of a given signature or signature category.</p>

	Command or Action	Purpose
Step 13	exit Example: Router(config-sigdef-status)# exit	Returns to EXEC mode.
Step 14	show ip ips signature Example: Router# show ip ips signature	(Optional) Verifies the signature changes that have been made.
Step 15	show ip ips sig-clidelta Example: Router# show ip ips sig-clidelta	(Optional) Displays the signature parameter tunings configured using the CLI, which are stored in the iosips-sig-clidelta.xmz signature file.

Tuning Signatures per Category

Use this task to change default signature parameters for a category of signatures. Categories such as operating systems; Layer 2, Layer 3, or Layer 4 protocols; or service-based categories can be configured to provide wider changes to a group of signatures.



Tip

Category configuration information is processed in the order that it is entered. Thus, it is recommended that the process of retiring all signatures. See "Retiring All Signatures and Selecting a Category of Signatures" before all other category tuning. If a category is configured more than once, the parameters entered in the second configuration are added to or replace the previous configuration.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **ip ips signature-category**
4. **category** *category* [*subcategory*]
5. **event-action** *action*
6. **alert-severity** {**high** | **medium** | **low** | **informational**}
7. **fidelity-rating** *rating*
8. **enabled** {**true** | **false**}
9. **retired** {**true** | **false**}
10. **end**
11. **show ip ips signature**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Router> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: Router# configure terminal	Enters global configuration mode.
Step 3	ip ips signature-category Example: Router(config)# ip ips signature-category	Enters IPS category configuration mode.
Step 4	category <i>category</i> [<i>subcategory</i>] Example: Router(config-ips-category)# category attack adware/spyware	Specifies a category that is to be used for multiple signature actions or conditions and enters IPS category action configuration mode.
Step 5	event-action <i>action</i> Example: Router(config-ips-category-action)# event- action produce-alert	Changes router actions for a specified signature category. The <i>action</i> argument can be any of the following options: <ul style="list-style-type: none"> • deny-attacker-inline • deny-connection-inline • deny-packet-inline • produce-alert • reset-tcp-connection Note Event actions associated with a category can be entered separately or on a single line.
Step 6	alert-severity {high medium low informational} Example: Router(config-ips-category-action)# alert- severity medium	(Optional) Changes the alert severity rating for a given signature category.

	Command or Action	Purpose
Step 7	fidelity-rating <i>rating</i> Example: Router(config-ips-category-action)# fidelity-rating	(Optional) Changes the signature fidelity rating for a signature given category.
Step 8	enabled {true false} Example: Router(config-ips-category-action)# enabled true	(Optional) Changes the enabled status of a given signature or signature category.
Step 9	retired {true false } Example: Router(config-ips-category-action)# retired true	(Optional) Specifies whether or not the router should retire a signature category.
Step 10	end Example: Router(config-ips-category-action)# end	Returns to EXEC mode.
Step 11	show ip ips signature Example: Router# show ip ips signature	(Optional) Verifies the signature category changes that have been made.

Enabling Signature Tunings Inheritance

When new signatures are replacing older signatures, Cisco IOS IPS provides the **ip ips inherit-obsolete-tunings** command to enable new signatures to obsolete older signatures and inherit the event-action and enabled parameters of the obsolete tuning values, without the need to manually tune the new signatures. This functionality is called signature tuning inheritance. All other parameter changes including the “Retire” parameter is ignored.

After you enter the command, the screen displays a warning message asking you to clarify the intended usage and then asks whether you accept the configuration or not. By default, old signatures are not inherited by new signatures.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **ip ips inherit-obsolete-tunings**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: <pre>Router> enable</pre>	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: <pre>Router# configure terminal</pre>	Enters global configuration mode.
Step 3	ip ips inherit-obsolete-tunings Example: <pre>Router(config)#ip ips inherit-obsolete-tunings</pre>	Enables the inheritance of the tunings of the enabled and event-action parameters from obsolete signatures to new signatures in an IPS,

Setting an IPS Memory Threshold

When a router is powered up, 90 percent of the available memory is allocated to IPS-related activities. The remaining 10 percent is referred to as the IPS Memory Threshold--the amount of free memory unavailable to the IPS. Cisco IOS IPS allows the IPS memory threshold to be set to a different value using the **ip ips memory threshold** command.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **ip ips memory threshold megabytes**

DETAILED STEPS

Command or Action	Purpose
<p>Step 1 <code>enable</code></p> <p>Example:</p> <pre>Router> enable</pre>	<p>Enables privileged EXEC mode.</p> <ul style="list-style-type: none"> Enter your password if prompted.
<p>Step 2 <code>configure terminal</code></p> <p>Example:</p> <pre>Router# configure terminal</pre>	<p>Enters global configuration mode.</p>
<p>Step 3 <code>ip ips memory threshold megabytes</code></p> <p>Example:</p> <p>Example:</p> <p>Example:</p> <p>Example:</p> <pre>Router(config)# ip ips memory threshold 50</pre>	<p>Specifies an IPS memory threshold, or the amount of free memory unavailable to the IPS.</p> <ul style="list-style-type: none"> The units are in megabytes (MB). This example specifies that Cisco IOS IPS cannot consume more memory if the remaining (free) memory becomes less than 50 MB.

Setting the Target Value Rating

Use this task to set the target value rating, which allows users to develop security policies that can be more strict for some resources than others. The security policy is applied to a table of hosts that are protected by Cisco IOS IPS. A host can be a single IP address or a range of IP addresses with an associated target value rating.



Note

Changes to the target value rating is not shown in the run time config because the changes are recorded in the seap-delta.xml file, which can be located through the **ip ips config location** command.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **ip ips event-action-rules**
4. **target-value {mission-critical | high | medium | low} target-address ip-address [/nn | to ip-address]**
5. **exit**

DETAILED STEPS

Command or Action	Purpose
Step 1 enable Example: Router> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2 configure terminal Example: Router# configure terminal	Enters global configuration mode.
Step 3 ip ips event-action-rules Example: Router(config)# ip ips event-action-rules	Enters the config-rule configuration mode, which allows users to change the target value rating.
Step 4 target-value {mission-critical high medium low} target-address ip-address [/nn to ip-address] Example: Router(config-rul)# target-value medium target-address 10.12.100.53	Sets the target value rating for a host to rate how important the system is to the network. <ul style="list-style-type: none"> • The target-address keyword and arguments specify a host, which can consist of a single IP address or range of IP addresses.
Step 5 exit Example: Router(config-rul)# exit	Exits config-rule configuration mode.

Configuring Signature Auto Updates from a Local Server**Note**

This functionality was introduced through the Cisco IOS IPS 5.x Signature Format and Usability Enhancements feature in Cisco IOS Release 12.4(11)T.

SUMMARY STEPS

1. enable
2. configure terminal
3. key config-key password-encrypt
4. password encryption aes
5. ip ips auto-update
6. occur-at {[monthly | weekly] days minutes hours}
7. username name password password
8. url url
9. exit
10. show ip ips auto-update

DETAILED STEPS

	Command or Action	Purpose
Step 1	<p>enable</p> <p>Example:</p> <pre>Router> enable</pre>	<p>Enables privileged EXEC mode.</p> <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	<p>configure terminal</p> <p>Example:</p> <pre>Router# configure terminal</pre>	<p>Enters global configuration mode.</p>
Step 3	<p>key config-key password-encrypt</p> <p>Example:</p> <pre>Router(config-ips-auto-update)# key config-key password-encrypt</pre>	<p>Stores a type 6 encryption key in private NVRAM.</p>
Step 4	<p>password encryption aes</p> <p>Example:</p> <pre>Router(config-ips-auto-update)# password encryption aes</pre>	<p>Enables a type 6 encrypted preshared key.</p> <p>Note Once the key config-key password-encrypt and password encryption aes commands are configured, they enable the password (symmetric cipher Advanced Encryption Standard (AES) encrypts the keys).</p>

Command or Action	Purpose
Step 5 <code>ip ips auto-update</code> Example: <pre>Router(config)# ip ips auto-update</pre>	Enables automatic signature updates for Cisco IOS IPS and enters IPS auto-update configuration mode.
Step 6 <code>occur-at {[monthly weekly] days minutes hours}</code> Example: <pre>Router(config-ips-auto-update)# occur-at weekly 4 23 23</pre>	(Optional) Defines a preset time for which the Cisco IOS IPS automatically obtains updated signature information.
Step 7 <code>username name password password</code> Example: <pre>Router(config-ips-auto-update)# username myips password secret</pre>	(Optional) Defines a username and password for the signature update function.
Step 8 <code>url url</code> Example: <pre>Router(config-ips-auto-update)# url tftp://10.10.10.5/username1/ips-auto-update/</pre>	URL from the local server on which the router retrieves the Cisco IOS IPS signature configuration files.
Step 9 <code>exit</code> Example: <pre>Router(config-ips-auto-update)# exit Router(config)# exit</pre>	Exits IPS auto-update and global configuration modes.
Step 10 <code>show ip ips auto-update</code> Example: <pre>Router# show ip ips auto-update</pre>	Verifies the signature update configuration.

Monitoring Cisco IOS IPS Signatures through Syslog Messages or SDEE

Cisco IOS IPS provides two methods to report IPS intrusion alerts--Cisco IOS logging (syslog) and SDEE. Perform this task to enable SDEE to report IPS intrusion alerts. See the “Troubleshooting and Fault Management” feature module for more information on configuring syslog messages.

- [SDEE Overview, page 71](#)

- [Troubleshooting Tips, page 72](#)

SDEE Overview

SDEE is an application-level communication protocol that is used to exchange IPS messages between IPS clients and IPS servers. SDEE is always running, but it does not receive and process events from IPS unless SDEE notification is enabled. If SDEE notification is not enabled and a client sends a request, SDEE responds with a fault response message, indicating that notification is not enabled.

Storing SDEE Events in the Buffer

When SDEE notification is enabled (through the **ip ips notify sdee** command), 200 events can automatically be stored in the buffer. When SDEE notification is disabled, all stored events are lost. A new buffer is allocated when the notifications are reenabled.

When specifying the size of an events buffer, note the following functionality:

- It is circular. When the end of the buffer is reached, the buffer starts overwriting the earliest stored events. (If overwritten events have not yet been reported, a buffer overflow notice is received.)
- If a new, smaller buffer is requested, all events that are stored in the previous buffer is lost.
- If a new, larger buffer is requested, all existing events are saved.

To use SDEE, the HTTP server must be enabled (through the **ip http server** command). If the HTTP server is not enabled, the router cannot respond to the SDEE clients because it cannot not “see” the requests.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **ip ips notify sdee**
4. **ip sdee events *events***
5. **ip sdee subscriptions *subscriptions***
6. **ip sdee messages *messages***
7. **ip sdee alerts *alerts***
8. **exit**
9. **show ip sdee [alerts | all | errors | events | configuration | status | subscriptions]**

DETAILED STEPS

Command or Action	Purpose
Step 1 enable Example: Router> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2 configure terminal Example: Router# configure terminal	Enters global configuration mode.

Command or Action	Purpose
Step 3 <code>ip ips notify sdee</code> Example: <pre>Router(config)# ip ips notify sdee</pre>	Enables SDEE event notification on a router.
Step 4 <code>ip sdee events <i>events</i></code> Example: <pre>Router(config)# ip sdee events 500</pre>	(Optional) Sets the maximum number of SDEE events that can be stored in the event buffer. <ul style="list-style-type: none"> Maximum value: 1000 events. Note By default, 200 events can be stored in the buffer when SDEE is enabled. When SDEE is disabled, all stored events are lost; a new buffer is allocated when the notifications are reenabled.
Step 5 <code>ip sdee subscriptions <i>subscriptions</i></code> Example: <pre>Router(config)# ip sdee subscriptions 1</pre>	(Optional) Sets the maximum number of SDEE subscriptions that can be open simultaneously. <ul style="list-style-type: none"> Valid value ranges from 1 to 3.
Step 6 <code>ip sdee messages <i>messages</i></code> Example: <pre>Router(config)# ip sdee messages 500</pre>	(Optional) Sets the maximum number of SDEE messages that can be stored in the buffer at one time.
Step 7 <code>ip sdee alerts <i>alerts</i></code> Example: <pre>Router(config)# ip sdee alerts 2000</pre>	(Optional) Sets the maximum number of SDEE alerts that can be stored in the buffer at one time.
Step 8 <code>exit</code> Example: <pre>Router(config)# exit</pre>	Exits global configuration mode.
Step 9 <code>show ip sdee [alerts all errors events configuration status subscriptions]</code> Example: <pre>Router# show ip sdee configuration</pre>	(Optional) Verifies SDEE configuration information and notification functionality.

Troubleshooting Tips

To print out new SDEE alerts on the router console, issue the **debug ip sdee** command.

To clear the event buffer or SDEE subscriptions from the router (which helps with error recovery), issue the **clear ip sdee** command.

Configuration Examples for Cisco IOS IPS 5.x Signature Format Support and Usability Enhancements

- [Cisco IOS IPS Configuration Example, page 73](#)
- [Configuring and Verifying SDEE on your Router Example, page 76](#)

Cisco IOS IPS Configuration Example

The following example shows how to enable and verify Cisco IOS IPS on your router:

```

Router# mkdir
flash:/ips5
Create directory filename [ips5]?
Created dir flash:/ips5
Router#
Router#
Router#
Router# configure terminal
Enter configuration commands, one per line. End with CNTL/Z.
Router(config)# ip ips name MYIPS
Router(config)# ip ips config location
flash:/ips5
Router(config)# ip ips signature-category
Router(config-ips-category)# category all
Router(config-ips-category-action)# retired true
Router(config-ips-category-action)# exit
Router(config-ips-category)# category ios_ips advanced
Router(config-ips-category-action)# retired false
Router(config-ips-category-action)# exit

Router(config-ips-category)# exit
Do you want to accept these changes? [confirm]
Router(config)# d
*Nov 14 2006 17:16:42 MST: Applying Category configuration to signatures ..
Router(config)#
Router(config)# do show ip interface brief
Interface                IP-Address      OK?    Method  Status        Protocol
GigabitEthernet0/0       10.0.20.120     YES    NVRAM   up            up
GigabitEthernet0/1       10.12.100.120  YES    NVRAM   administratively down  down
NVI0                     unassigned      NO     unset   up            up
Router(config)#
Router(config)# interface gigabits 0/0
Router(config-if)# ip ips MYIPS in
Router(config-if)#
*Nov 14 2006 17:17:07 MST: %IPS-6-ENGINE_BUILDS_STARTED: 17:17:07 MST Nov 14 2006
*Nov 14 2006 17:17:07 MST: %IPS-6-ENGINE_BUILDING: atomic-ip - 3 signatures - 1 of 13
engines
*Nov 14 2006 17:17:07 MST: %IPS-6-ENGINE_READY: atomic-ip - build time 0 ms - packets for
this engine will be scanned
*Nov 14 2006 17:17:07 MST: %IPS-6-ALL_ENGINE_BUILDS_COMPLETE: elapsed time 0 ms
Router(config-if)#
Router(config-if)# ip ips MYIPS out
Router(config-if)#
Router(config-if)#
Router(config-if)#^Z
Router#
*Nov 14 2006 17:17:23 MST: %SYS-5-CONFIG_I: Configured from console by cisco on console
Router# wr
Building configuration...
[OK]

```

```

Router#
Router# show ip ips signature count
Cisco SDF release version S0.0
Signature Micro-Engine: multi-string (INACTIVE)
Signature Micro-Engine: service-http (INACTIVE)
Signature Micro-Engine: string-tcp (INACTIVE)
Signature Micro-Engine: string-udp (INACTIVE)
Signature Micro-Engine: state (INACTIVE)
Signature Micro-Engine: atomic-ip
    Total Signatures: 3
        Enabled: 0
        Compiled: 3
Signature Micro-Engine: string-icmp (INACTIVE)
Signature Micro-Engine: service-ftp (INACTIVE)
Signature Micro-Engine: service-rpc (INACTIVE)
Signature Micro-Engine: service-dns (INACTIVE)
Signature Micro-Engine: normalizer (INACTIVE)
Signature Micro-Engine: service-smb-advanced (INACTIVE)
Signature Micro-Engine: service-msrpc (INACTIVE)
    Total Signatures: 3
    Total Enabled Signatures: 0
    Total Retired Signatures: 0
    Total Compiled Signatures: 3
Router#
Router# copy flash:IOS-S258-CLI-kd.pkg idconf
*Nov 14 2006 17:19:47 MST: %IPS-6-ENGINE_BUILDS_STARTED: 17:19:47 MST Nov 14 2006
*Nov 14 2006 17:19:47 MST: %IPS-6-ENGINE_BUILDING: multi-string - 3 signatures - 1 of 13 engines
*Nov 14 2006 17:19:47 MST: %IPS-6-ENGINE_READY: multi-string - build time 4 ms - packets for this engine will be scanned
*Nov 14 2006 17:19:47 MST: %IPS-6-ENGINE_BUILDING: service-http - 611 signatures - 2 of 13 engines
*Nov 14 2006 17:20:00 MST: %IPS-6-ENGINE_READY: service-http - build time 12932 ms - packets for this engine will be scanned
*Nov 14 2006 17:20:00 MST: %IPS-6-ENGINE_BUILDING: string-tcp - 864 signatures - 3 of 13 engines
*Nov 14 2006 17:20:02 MST: %IPS-6-ENGINE_READY: string-tcp - build time 2692 ms - packets for this engine will be scanned
*Nov 14 2006 17:20:02 MST: %IPS-6-ENGINE_BUILDING: string-udp - 74 signatures - 4 of 13 engines
*Nov 14 2006 17:20:03 MST: %IPS-6-ENGINE_READY: string-udp - build time 316 ms - packets for this engine will be scanned
*Nov 14 2006 17:20:03 MST: %IPS-6-ENGINE_BUILDING: state - 28 signatures - 5 of 13 engines
*Nov 14 2006 17:20:03 MST: %IPS-6-ENGINE_READY: state - build time 24 ms - packets for this engine will be scanned
*Nov 14 2006 17:20:03 MST: %IPS-6-ENGINE_BUILDING: atomic-ip - 252 signatures - 6 of 13 engines
*Nov 14 2006 17:20:03 MST: %IPS-4-META_ENGINE_UNSUPPORTED: atomic-ip 2154:0 - this signature is a component of the unsupported META engine
*Nov 14 2006 17:20:03 MST: %IPS-6-ENGINE_READY: atomic-ip - build time 232 ms - packets for this engine will be scanned
*Nov 14 2006 17:20:03 MST: %IPS-6-ENGINE_BUILDING: string-icmp - 3 signatures - 7 of 13 engines
Router# engines
*Nov 14 2006 17:20:03 MST: %IPS-6-ENGINE_READY: string-icmp - build time 12 ms - packets for this engine will be scanned
*Nov 14 2006 17:20:03 MST: %IPS-6-ENGINE_BUILDING: service-ftp - 3 signatures - 8 of 13 engines
*Nov 14 2006 17:20:03 MST: %IPS-6-ENGINE_READY: service-ftp - build time 8 ms - packets for this engine will be scanned
*Nov 14 2006 17:20:03 MST: %IPS-6-ENGINE_BUILDING: service-rpc - 75 signatures - 9 of 13 engines
*Nov 14 2006 17:20:03 MST: %IPS-6-ENGINE_READY: service-rpc - build time 80 ms - packets for this engine will be scanned
*Nov 14 2006 17:20:03 MST: %IPS-6-ENGINE_BUILDING: service-dns - 38 signatures - 10 of 13 engines
*Nov 14 2006 17:20:03 MST: %IPS-6-ENGINE_READY: service-dns - build time 20 ms - packets for this engine will be scanned
*Nov 14 2006 17:20:03 MST: %IPS-6-ENGINE_BUILDING: normalizer - 9 signatures - 11 of 13 engines
*Nov 14 2006 17:20:03 MST: %IPS-6-ENGINE_READY: normalizer - build time 0 ms - packets for this engine will be scanned
*Nov 14 2006 17:20:03 MST: %IPS-6-ENGINE_BUILDING: service-msrpc - 22 signatures - 12 of 13 engines

```

```
*Nov 14 2006 17:20:03 MST: %IPS-6-ENGINE_READY: service-msrpc - build time 8 ms - packets
for this engine will be scanned
*Nov 14 2006 17:20:03 MST: %IPS-6-ALL_ENGINE_BUILDS_COMPLETE: elapsed time 16344 ms
Router#
Router#
Router# show ip ips signature count
Cisco SDF release version S258.0
Signature Micro-Engine: multi-string
  Total Signatures: 3
    Enabled: 3
    Retired: 3
Signature Micro-Engine: service-http
  Total Signatures: 611
    Enabled: 159
    Retired: 428
    Compiled: 183
Signature Micro-Engine: string-tcp
  Total Signatures: 864
    Enabled: 414
    Retired: 753
    Compiled: 111
Signature Micro-Engine: string-udp
  Total Signatures: 74
    Enabled: 1
    Retired: 44
    Compiled: 30
Signature Micro-Engine: state
  Total Signatures: 28
    Enabled: 16
    Retired: 25
    Compiled: 3
Signature Micro-Engine: atomic-ip
  Total Signatures: 252
    Enabled: 56
    Retired: 148
    Compiled: 103
    Inactive - invalid params: 1
Signature Micro-Engine: string-icmp
  Total Signatures: 3
    Enabled: 0
    Retired: 2
    Compiled: 1
Signature Micro-Engine: service-ftp
  Total Signatures: 3
    Enabled: 1
    Compiled: 3
Signature Micro-Engine: service-rpc
  Total Signatures: 75
    Enabled: 44
    Retired: 44
    Compiled: 31
Signature Micro-Engine: service-dns
  Total Signatures: 38
    Enabled: 30
    Retired: 5
    Compiled: 33
Signature Micro-Engine: normalizer
  Total Signatures: 9
    Enabled: 8
    Retired: 5
    Compiled: 4
Signature Micro-Engine: service-smb-advanced (INACTIVE)
Signature Micro-Engine: service-msrpc
  Total Signatures: 22
    Enabled: 22
    Retired: 22
```

Configuring and Verifying SDEE on your Router Example

The following example shows how to configure and verify SDEE on your router:

```
Router(config)# ip ips notify SDEE

Router(config)# ip sdee event 500
Router(config)# ip sdee subscriptions 1
Router(config)# ip sdee messages 500
Router(config)# ip sdee alerts 2000
router(config)# exit
*Nov 9 21:41:33.171: %SYS-5-CONFIG_I: Configured from console by cisco on console
Router# show ip sdee all
Configured concurrent subscriptions: 1
No currently open subscriptions.
Alert storage: 2000 alerts using 560000 bytes of memory
Message storage: 500 messages using 212000 bytes of memory
SDEE Events
Time Type Description
Router#
```

Additional References

Related Documents

Related Topic	Document Title
Cisco IOS commands	<i>Cisco IOS Master Commands List, All Releases</i>
Security commands	<ul style="list-style-type: none"> • Cisco IOS Security Command Reference: Commands A to C • Cisco IOS Security Command Reference: Commands D to L • Cisco IOS Security Command Reference: Commands M to R • Cisco IOS Security Command Reference: Commands S to Z

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 Cisco IOS IPS 5.x Signature Format Support and Usability Enhancements

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 Cisco IOS 5.x Signature Format Support and Usability Enhancements

Feature Name	Releases	Feature Information
Cisco IOS IPS 5.x Signature Format and Usability Enhancements	12.4(11)T	<p>This feature introduces support for Cisco IOS Intrusion Prevention System (IPS) version 5.0, which is a version-based signature definition XML format. Cisco IOS IPS 4.x format signatures are replaced by the 5.x format signatures that are used by all other Cisco IPS devices.</p> <p>The following commands were introduced or modified by this feature: alert-severity, category, copy idconf enabled (IPS), engine (IPS), event-action, fidelity-rating, ip ips auto-update, ip ips config location, ip ips event-action-rules, ip ips signature-category, ip ips signature-definition, occur-at (ips-auto-update), retired (IPS), show ip ips auto-update, signature, status, target-value url (ips-auto-update), username (ips-autoupdate).</p>
Direct Download from CCO capability in IOS IPS	15.1(1)T	<p>The following commands were introduced or modified by this feature: cisco, ida-client server url, ip ips auto-update, ip signature update cisco, occur-at.</p>

Feature Name	Releases	Feature Information
Capability to save local delta changes on IOS routers	15.1(2)T	<p>This feature was introduced to generate a local cli-delta.xml file on the router containing the signature tuning settings configured through the CLI. This local file takes precedence when a globally administered delta signature update, contained in the IPS iosips-sig-delta.xml file, is sent from a central repository and applied to the configuration of the local router.</p> <p>The following commands were introduced or modified: ip ips enable-clidelta, show ip ips sig-clidelta.</p>
Cisco IOS IPS with Lightweight Signatures	15.2(1)T and 15.2(2)T	<p>This feature was deprecated in Cisco IOS Release 15.2(1)T and 15.2(2)T.</p> <p>The following commands were deprecated: ip ips inherit-obsolete tunings, ip ips memory regex chaining, ip ips memory threshold.</p>

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

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



Cisco IOS IPS Support for Microsoft Engines

The Cisco IOS IPS Support for Microsoft Engines feature extends Cisco IOS Intrusion Prevention Systems (IPS) to support Microsoft RPC (Remote Procedure Call) and Microsoft SMB (Server Message Block) protocols. IPS signatures can now scan for, detect, and take proper action against vulnerabilities in MSRPC and SMB protocols.

- [Finding Feature Information, page 79](#)
- [Information About Cisco IOS IPS Support for Microsoft Engines, page 79](#)
- [How to Use Cisco IOS IPS, page 80](#)
- [Configuration Examples for Cisco IOS IPS, page 80](#)
- [Additional References, page 81](#)
- [Feature Information for Cisco IOS IPS Support for Microsoft Engines, page 82](#)

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.

Information About Cisco IOS IPS Support for Microsoft Engines

- [Cisco IOS IPS Overview, page 79](#)

Cisco IOS IPS Overview

The Cisco IOS IPS acts as an in-line intrusion prevention sensor, watching packets and sessions as they flow through the router and scanning each packet to match any of the Cisco IOS IPS signatures. When it detects suspicious activity, it responds before network security can be compromised and logs the event through Cisco IOS syslog messages or Security Device Event Exchange (SDEE). The network administrator can configure Cisco IOS IPS to choose the appropriate response to various threats. The Signature Event Action Processor (SEAP) can dynamically control actions that are to be taken by a signature event on the basis of parameters such as fidelity, severity, or target value rating. These parameters

have default values but can also be configured via CLI. When packets in a session match a signature, Cisco IOS IPS can take any of the following actions, as appropriate:

- Send an alarm to a syslog server or a centralized management interface
- Drop the packet
- Reset the connection
- Deny traffic from the source IP address of the attacker for a specified amount of time
- Deny traffic on the connection for which the signature was seen for a specified amount of time

Cisco developed its Cisco IOS software-based intrusion-prevention capabilities and Cisco IOS Firewall with flexibility in mind, so that individual signatures could be disabled in case of false positives. Generally, it is preferable to enable both the firewall and Cisco IOS IPS to support network security policies. However, each of these features may be enabled independently and on different router interfaces.

How to Use Cisco IOS IPS

The addition of the MSRPC and MSB protocol support does not change the way in which Cisco IOS IPS is defined and enabled in your network. For information on how to enable IPS on your network via command-line interface (CLI), see the section “How to Use Cisco IOS 5.x Format Signatures with Cisco IOS IPS” within the document *Cisco IOS IPS 5.x Signature Format Support and Usability Enhancement*.

Configuration Examples for Cisco IOS IPS

- [show ip ips signature Output to Verify MS Engines Example, page 80](#)

show ip ips signature Output to Verify MS Engines Example

The following sample output from the **show ip ips signature** command displays output for the service-msrpc and service-smb-advanced signatures:

```
Signature Micro-Engine: service-msrpc: Total Signatures 21
service-msrpc enabled signatures: 21
service-msrpc compiled signatures: 21
SigID:SubID En Cmp Action Sev Trait EC AI GST SI SM SW SFR Rel
-----
3330:0 Y Y A HIGH 0 1 0 0 0 FA N 100 S148
3332:0 Y Y A HIGH 0 35 0 0 0 FA N 100 S148
3337:0 Y Y A HIGH 0 8 2 0 0 FA N 100 S85
3331:2 Y Y A HIGH 0 1 0 0 0 FA N 90 S215
3327:12 Y Y A HIGH 0 1 0 0 0 FA N 85 S214
3328:3 Y Y A MED 0 1 0 0 0 FA N 85 S170
3328:1 Y Y A MED 0 1 0 0 0 FA N 85 S148
3327:8 Y Y A INFO 0 1 0 0 0 FA N 85 S214
3334:6 Y Y A HIGH 0 1 0 0 0 FA N 80 S215
3327:0 Y Y A HIGH 0 1 0 0 0 FA N 80 S165
6232:0 Y Y A HIGH 0 1 0 0 0 FA N 75 S209
3327:4 Y Y A HIGH 0 1 0 0 0 FA N 75 S188
3334:5 Y Y A HIGH 0 2 2 0 0 FA N 75 S179
3338:2 Y Y A HIGH 0 40 3 0 0 FA N 75 S175
3338:3 Y Y A HIGH 0 1 0 0 0 FA N 75 S175
6130:0 Y Y A HIGH 0 1 0 0 0 FA N 75 S167
6130:6 Y Y A INFO 0 1 0 0 0 FA N 75 S192
5567:1 Y Y A INFO 0 1 0 0 0 FA N 55 S187
5567:2 Y Y A INFO 0 1 0 0 0 FA N 55 S187
5567:3 Y Y A INFO 0 1 0 0 0 FA N 55 S187
5567:4 Y Y A INFO 0 1 0 0 0 FA N 55 S187
```

```
Signature Micro-Engine: service-smb-advanced: Total Signatures 31
service-smb-advanced enabled signatures: 31
service-smb-advanced compiled signatures: 31
SigID:SubID En Cmp Action Sev Trait EC AI GST SI SM SW SFR Rel
```

```
-----
5593:0 Y Y A HIGH 0 1 0 0 0 FA N 100 S262
5592:0 Y Y A HIGH 0 1 0 0 0 FA N 100 S262
5582:0 Y Y A HIGH 0 1 0 0 0 FA N 100 S262
5599:0 Y Y A HIGH 0 1 0 0 0 FA N 100 S262
5595:0 Y Y A MED 0 1 0 0 0 FA N 100 S262
5579:0 Y Y A INFO 0 1 0 0 0 FA N 100 S264
5581:0 Y Y A INFO 0 1 0 0 0 FA N 100 S264
5580:0 Y Y A INFO 0 1 0 0 0 FA N 100 S264
5584:0 Y Y A INFO 0 1 0 0 0 FA N 100 S262
5576:0 Y Y A INFO 0 1 0 0 0 FA N 100 S262
5577:0 Y Y A INFO 0 1 0 0 0 FA N 100 S262
5583:0 Y Y A INFO 0 1 0 0 0 FA N 100 S262
5591:0 Y Y A INFO 0 1 0 0 0 FA N 100 S262
5590:0 Y Y A INFO 0 1 0 0 0 FA N 100 S262
5598:0 Y Y A HIGH 0 1 0 0 0 FA N 85 S264
5588:0 Y Y A HIGH 0 1 0 0 0 FA N 85 S262
5586:0 Y Y A HIGH 0 1 0 0 0 FA N 85 S262
5585:0 Y Y A MED 0 1 0 0 0 FA N 85 S264
5579:1 Y Y A MED 0 1 0 0 0 FA N 85 S264
5602:0 Y Y A MED 0 1 0 0 0 FA N 85 S262
5589:0 Y Y A LOW 0 1 0 0 0 FA N 85 S262
5578:0 Y Y A INFO 0 1 0 0 0 FA N 85 S264
5605:0 Y Y A INFO 0 1 0 0 0 FA N 85 S262
5600:0 Y Y A HIGH 0 1 0 0 0 FA N 75 S262
5597:0 Y Y A HIGH 0 50 0 0 0 FA N 75 S262
5594:0 Y Y A HIGH 0 1 0 0 0 FA N 75 S262
5587:0 Y Y A HIGH 0 1 0 0 0 FA N 75 S262
5603:0 Y Y A MED 0 1 0 0 0 FA N 75 S262
5591:1 Y Y A INFO 0 1 0 0 0 FA N 75 S262
5575:0 Y Y A INFO 0 1 0 0 0 FA N 75 S262
5590:1 Y Y A INFO 0 1 0 0 0 FA N 75 S262
```

Additional References

Related Documents

Related Topic	Document Title
Cisco IOS commands	<i>Cisco IOS Master Commands List, All Releases</i>
Security commands	<ul style="list-style-type: none"> • Cisco IOS Security Command Reference: Commands A to C • Cisco IOS Security Command Reference: Commands D to L • Cisco IOS Security Command Reference: Commands M to R • Cisco IOS Security Command Reference: Commands S to Z

MIBs

MIB	MIBs Link
None	To locate and download MIBs for selected platforms, Cisco IOS 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 Cisco IOS IPS Support for Microsoft Engines

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

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

Table 7 Feature Information for Cisco IOS IPS Support for Microsoft Engines

Feature Name	Releases	Feature Information
Cisco IOS IPS Support for Microsoft Engines	12.4(15)T	The Cisco IOS IPS Support for Microsoft Engines feature extends Cisco IPS to support MSRPC and SMB protocols. The following commands were introduced or modified: debug ip ips

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

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



VRF Aware Cisco IOS IPS

Virtual Route Forwarding or Virtual Private Network (VPN) Route Forwarding (VRF), is a mechanism that allows multiple instances of a routing table to exist on a router and work simultaneously. This mechanism allows for network paths to be segregated without using multiple devices, thereby increasing network security and eliminating the need for encryption and authentication. VRFs are generally used to create separate VPNs. Allowing Intrusion Prevention System (IPS) to be configured on a per-VRF basis means global parameters will be shared by multiple VPNs, providing VRF related information on the Security Device Event Exchange (SDEE) and syslog alerts.

- [Finding Feature Information, page 85](#)
- [Prerequisites for VRF Aware Cisco IOS IPS, page 85](#)
- [Restrictions for VRF Aware Cisco IOS IPS, page 86](#)
- [Information About VRF Aware Cisco IOS IPS, page 86](#)
- [How to VRF Aware Cisco IOS IPS, page 88](#)
- [Configuration Examples for VRF Aware Cisco IOS IPS, page 90](#)
- [Examples VRF Aware Cisco IOS IPS Output and Error Message, page 97](#)
- [Additional References, page 99](#)
- [Feature Information for VRF Aware Cisco IOS IPS, page 100](#)

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 VRF Aware Cisco IOS IPS

- Understand Cisco IOS IPS.
- Configure VRFs.
- Verify that the VRFs are operational.
- Verify IPS is supported.
- Capability to send SDEE alarms and syslog with VRF information.

- If two VPN networks have overlapping addresses, VRF-aware network address translation (NAT) is required for them to support VRF Aware Cisco IOS IPS.
- Every VRF instance requires a new IPS rule.

Restrictions for VRF Aware Cisco IOS IPS

- VRF Aware Cisco IOS IPS is not supported on Multiprotocol Label Switching (MPLS) interfaces.

Information About VRF Aware Cisco IOS IPS

- [Cisco IOS IPS, page 86](#)
- [VRF, page 86](#)
- [VRF Lite, page 87](#)
- [Applying IPS Directly to a VRF, page 88](#)

Cisco IOS IPS

The Cisco IOS IPS acts as an in-line intrusion detection sensor, watching packets and sessions as they flow through the router and scanning each packet to match any of the Cisco IOS IPS signatures. When Cisco IOS IPS detects suspicious activity, it responds before network security can be compromised and logs the event through Cisco IOS syslog messages or SDEE. The network administrator can configure Cisco IOS IPS to choose the appropriate response to various threats. When packets in a session match a signature, Cisco IOS IPS can take any of the following actions, as appropriate:

- Send an alarm to a syslog server or a centralized management interface
- Drop the packet
- Reset the connection
- Deny traffic from the source IP address of the attacker for a specified amount of time
- Deny traffic on the connection for which the signature was seen for a specified amount of time

Cisco IOS software-based intrusion-prevention capabilities and Cisco IOS firewall are developed with flexibility in mind, so that individual signatures could be disabled in case of false positives. Generally, it is preferable to enable both the firewall and Cisco IOS IPS to support network security policies. However, each of these features may be enabled independently and on different router interfaces.

VRF

Each VPN has its own routing and forwarding table in the router, so any customer or site that belongs to a VPN is provided access only to the set of routes contained within that table. Any Provider Edge (PE) router in the MPLS VPN network therefore contains a number of per-VPN routing tables and a global routing table that is used to reach other routers in the service provider network. Effectively, virtual routers are created in a single physical router.

VRF is a Cisco IOS route table instance for connecting a set of sites to a VPN service. A VRF contains a template of a VPN Routing/Forwarding table in a PE router.

The overlapping addresses, usually resulting from the use of private IP addresses in customer networks, are one of the major obstacles to successful deployment of peer-to-peer VPN implementation. The MPLS VPN technology provides a solution to this dilemma.

VRF Lite

VRF-lite is a feature that enables a service provider to support two or more VPNs, where IP addresses can be overlapped among the VPNs. VRF Lite uses input interfaces to distinguish routes for different VPNs and forms virtual packet-forwarding tables by associating one or more Layer 3 interfaces with each VRF. Interfaces in a VRF can be physical, such as Ethernet ports, or logical, such as VLAN switched virtual interfaces (SVIs). However, a Layer 3 interface cannot belong to more than one VRF at a time.



Note

VRF Lite interfaces must be Layer 3 interfaces.

VRF-lite includes these devices:

- Customer edge (CE) devices provide customer access to the service provider network over a data link to one or more PE routers. The CE device advertises the site's local routes to the PE router and learns the remote VPN routes from it. A Catalyst 4500 switch can be a CE device.
- PE routers exchange routing information with CE devices by using static routing or a routing protocol such as BGP, RIPv1, or RIPv2.
- Provider routers (or core routers) are any routers in the service provider network that do not attach to CE devices.
- The PE is only required to maintain VPN routes for those VPNs to which it is directly attached, eliminating the need for the PE to maintain all of the service provider VPN routes. Each PE router maintains a VRF for each of its directly connected sites. Multiple interfaces on a PE router can be associated with a single VRF if all of these sites participate in the same VPN. Each VPN is mapped to a specified VRF. After learning local VPN routes from CEs, a PE router exchanges VPN routing information with other PE routers by using Internal BGP (IBGP).

With VRF Lite, multiple customers can share one CE, and only one physical link is used between the CE and the PE. The shared CE maintains separate VRF tables for each customer and switches or routes packets for each customer based on its own routing table. VRF Lite extends limited PE functionality to a CE device, giving the CE device the ability to maintain separate VRF tables to extend the privacy and security of a VPN to the branch office.

If VRF is configured on an interface and then IPS is attached, an IPS control block is created as shown in the figure below, with an appropriate name for the VRF instead of the existing default value. VRF support for IPS allows customization of IPS parameters, settings and statistics per VRF interface and customization of IPS signature sets for each VRF user. The IPS VRF code takes care of the VRF support including overlapping addresses.

Figure 1 *IPS in a VRF-to-VRF Scenario*



Applying IPS Directly to a VRF

Virtual Route Forwarding (VRF) is a mechanism that allows multiple instances of a routing table to exist on a router and work simultaneously. This mechanism allows for network paths to be segregated without using multiple devices which increases network security and eliminates the need for encryption and authentication. VRFs are generally used to create separate Virtual Private Networks (VPNs). If VRF is configured on an interface and IPS is attached, an IPS control block is created with the appropriate name for the VRF instead of the existing default value. VRF support for IPS allows customization of IPS parameters, settings and statistics per VRF interface and customization of IPS signature sets for each VRF user. All interfaces share the same global parameters for IPS, but alarms and event log information on the SDEE and syslog alerts carry respective VRF information.

How to VRF Aware Cisco IOS IPS

- [Configuring a VRF and Applying IPS Directly to the VRF, page 88](#)

Configuring a VRF and Applying IPS Directly to the VRF

The following steps are used to configure VRF routing and forwarding tables, configuring VRF on an interface and attach IPS to this interface:

**Note**

If a global VRF is removed, the IPS configuration on the interfaces belonging to that VRF is cleaned and removed, including any sessions and statistics created on the VRF. The user must reconfigure IPS on the affected interfaces if they are to be used again.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **ip vrf vrfname**
4. **rd route-distinguisher**
5. **route-target export target VPN extended community**
6. **route-target import target VPN extended community**
7. **exit**
8. **interface FastEthernet port**
9. **ip vrf forwarding vrfname**
10. **ip address range**
11. **ip ips *ips-name* in**
12. **exit**
13. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	<p>enable</p> <p>Example:</p> <pre>Router> enable</pre>	<p>Enables privileged EXEC mode.</p> <ul style="list-style-type: none"> Enter your password if prompted.
Step 2	<p>configure terminal</p> <p>Example:</p> <pre>Router# configure terminal</pre>	<p>Enters global configuration mode.</p>
Step 3	<p>ip vrf vrfname</p> <p>Example:</p> <pre>Router# ip vrf VRF600</pre>	<p>Configures a VRF table and enters VRF configuration mode.</p>
Step 4	<p>rd route-distinguisher</p> <p>Example:</p> <pre>Router# rd 100:600</pre>	<p>Creates routing and forwarding tables for the VRF instance.</p> <p>Note There are two formats for configuring the route distinguisher argument. It can be configured in the as-number:network number (ASN:nn) format, as shown in the example, or it can be configured in the IP address:network number format (IP-address:nn).</p>
Step 5	<p>route-target export target VPN extended community</p> <p>Example:</p> <pre>Router(config-vrf)# route-target export 100:600</pre>	<p>Creates lists of export route-target extended communities for the specified VRF.</p>
Step 6	<p>route-target import target VPN extended community</p> <p>Example:</p> <pre>Router(config-vrf)#route-target import 100:600</pre>	<p>Creates lists of import route-target extended communities for the specified VRF.</p>
Step 7	<p>exit</p> <p>Example:</p> <pre>Router(config-vrf)# exit</pre>	<p>Exits VRF configuration mode.</p>

Command or Action	Purpose
<p>Step 8 <code>interface FastEthernet port</code></p> <p>Example:</p> <pre>Router(config)# interface FastEthernet0/1.600</pre>	<p>Enters subinterface configuration mode and specifies a subinterface that is associated with a VRF.</p>
<p>Step 9 <code>ip vrf forwarding vrfname</code></p> <p>Example:</p> <pre>Router(config-subif)# ip vrf forwarding VRF600</pre>	<p>Configures the forwarding details for the respective interfaces.</p>
<p>Step 10 <code>ip address range</code></p> <p>Example:</p> <pre>Router(config-subif)# ip address 192.168.25.3 255.255.255.0</pre>	<p>Sets a primary or secondary IP address for an interface.</p>
<p>Step 11 <code>ip ips ips-name in</code></p> <p>Example:</p> <pre>Router(config-subif)# ip ips ips_policy600 in</pre>	<p>Applies an IPS rule at an interface and automatically loads the signatures and builds the signature engines.</p>
<p>Step 12 <code>exit</code></p> <p>Example:</p> <pre>Router(config-subif)# exit</pre> <p>Example:</p> <pre>Router(config-if)# end</pre>	<p>Exits subinterface mode, and enters interface configuration mode.</p>
<p>Step 13 <code>end</code></p> <p>Example:</p> <pre>Router(config-if)# end</pre>	<p>Exits interface configuration mode.</p>

Configuration Examples for VRF Aware Cisco IOS IPS

- [Example Cisco IOS IPS Configuration, page 91](#)
- [Example VRF Aware Cisco IOS IPS Configuration Without Subinterfaces, page 93](#)
- [Example VRF Aware Cisco IOS IPS Configuration with Subinterfaces, page 94](#)
- [Example Multi VRF with IPS and Zone Based Policy \(ZBP\) Firewall, page 95](#)

Example Cisco IOS IPS Configuration

The following example shows how to enable and verify Cisco IOS IPS on your router:

```

Router# mkdir
flash:/ips5
Create directory filename [ips5]?
Created dir flash:/ips5
Router#
Router#
Router#
Router# configure terminal
Enter configuration commands, one per line. End with CNTL/Z.
Router(config)# ip ips name MYIPS
Router(config)# ip ips config location
flash:/ips5
Router(config)# ip ips signature-category
Router(config-ips-category)# category all
Router(config-ips-category-action)# retired true
Router(config-ips-category-action)# exit
Router(config-ips-category)# category ios_ips advanced
Router(config-ips-category-action)# retired false
Router(config-ips-category-action)# exit

Router(config-ips-category)# exit
Do you want to accept these changes? [confirm]
Router(config)# d
*Nov 14 2006 17:16:42 MST: Applying Category configuration to signatures ..
Router(config)#
Router(config)# do show ip interface brief
Interface                IP-Address      OK?    Method  Status        Protocol
GigabitEthernet0/0       10.0.20.120     YES    NVRAM   up            up
GigabitEthernet0/1       10.12.100.120  YES    NVRAM   administratively down  down
NVI0                     unassigned      NO     unset   up            up
Router(config)#
Router(config)# interface gigabits 0/0
Router(config-if)# ip ips MYIPS in
Router(config-if)#
*Nov 14 2006 17:17:07 MST: %IPS-6-ENGINE_BUILDS_STARTED: 17:17:07 MST Nov 14 2006
*Nov 14 2006 17:17:07 MST: %IPS-6-ENGINE_BUILDING: atomic-ip - 3 signatures - 1 of 13
engines
*Nov 14 2006 17:17:07 MST: %IPS-6-ENGINE_READY: atomic-ip - build time 0 ms - packets for
this engine will be scanned
*Nov 14 2006 17:17:07 MST: %IPS-6-ALL_ENGINE_BUILDS_COMPLETE: elapsed time 0 ms
Router(config-if)#
Router(config-if)# ip ips MYIPS out
Router(config-if)#
Router(config-if)#
Router(config-if)#^Z
Router#
*Nov 14 2006 17:17:23 MST: %SYS-5-CONFIG_I: Configured from console by cisco on console
Router# wr
Building configuration...
[OK]
Router#
Router# show ip ips signature count
Cisco SDF release version S0.0
Signature Micro-Engine: multi-string (INACTIVE)
Signature Micro-Engine: service-http (INACTIVE)
Signature Micro-Engine: string-tcp (INACTIVE)
Signature Micro-Engine: string-udp (INACTIVE)
Signature Micro-Engine: state (INACTIVE)
Signature Micro-Engine: atomic-ip
Total Signatures: 3

```

```

        Enabled: 0
        Compiled: 3
Signature Micro-Engine: string-icmp (INACTIVE)
Signature Micro-Engine: service-ftp (INACTIVE)
Signature Micro-Engine: service-rpc (INACTIVE)
Signature Micro-Engine: service-dns (INACTIVE)
Signature Micro-Engine: normalizer (INACTIVE)
Signature Micro-Engine: service-smb-advanced (INACTIVE)
Signature Micro-Engine: service-msrpc (INACTIVE)
        Total Signatures: 3
        Total Enabled Signatures: 0
        Total Retired Signatures: 0
        Total Compiled Signatures: 3
Router#
Router# copy flash:IOS-S258-CLI-kd.pkg idconf
*Nov 14 2006 17:19:47 MST: %IPS-6-ENGINE_BUILDS_STARTED: 17:19:47 MST Nov 14 2006
*Nov 14 2006 17:19:47 MST: %IPS-6-ENGINE_BUILDING: multi-string - 3 signatures - 1 of 13
engines
*Nov 14 2006 17:19:47 MST: %IPS-6-ENGINE_READY: multi-string - build time 4 ms - packets
for this engine will be scanned
*Nov 14 2006 17:19:47 MST: %IPS-6-ENGINE_BUILDING: service-http - 611 signatures - 2 of
13 engines
*Nov 14 2006 17:20:00 MST: %IPS-6-ENGINE_READY: service-http - build time 12932 ms -
packets for this engine will be scanned
*Nov 14 2006 17:20:00 MST: %IPS-6-ENGINE_BUILDING: string-tcp - 864 signatures - 3 of 13
engines
*Nov 14 2006 17:20:02 MST: %IPS-6-ENGINE_READY: string-tcp - build time 2692 ms - packets
for this engine will be scanned
*Nov 14 2006 17:20:02 MST: %IPS-6-ENGINE_BUILDING: string-udp - 74 signatures - 4 of 13
engines
*Nov 14 2006 17:20:03 MST: %IPS-6-ENGINE_READY: string-udp - build time 316 ms - packets
for this engine will be scanned
*Nov 14 2006 17:20:03 MST: %IPS-6-ENGINE_BUILDING: state - 28 signatures - 5 of 13 engines
*Nov 14 2006 17:20:03 MST: %IPS-6-ENGINE_READY: state - build time 24 ms - packets for
this engine will be scanned
*Nov 14 2006 17:20:03 MST: %IPS-6-ENGINE_BUILDING: atomic-ip - 252 signatures - 6 of 13
engines
*Nov 14 2006 17:20:03 MST: %IPS-4-META_ENGINE_UNSUPPORTED: atomic-ip 2154:0 - this
signature is a component of the unsupported META engine
*Nov 14 2006 17:20:03 MST: %IPS-6-ENGINE_READY: atomic-ip - build time 232 ms - packets
for this engine will be scanned
*Nov 14 2006 17:20:03 MST: %IPS-6-ENGINE_BUILDING: string-icmp - 3 signatures - 7 of 13 e
Router# engines
*Nov 14 2006 17:20:03 MST: %IPS-6-ENGINE_READY: string-icmp - build time 12 ms - packets
for this engine will be scanned
*Nov 14 2006 17:20:03 MST: %IPS-6-ENGINE_BUILDING: service-ftp - 3 signatures - 8 of 13
engines
*Nov 14 2006 17:20:03 MST: %IPS-6-ENGINE_READY: service-ftp - build time 8 ms - packets
for this engine will be scanned
*Nov 14 2006 17:20:03 MST: %IPS-6-ENGINE_BUILDING: service-rpc - 75 signatures - 9 of 13
engines
*Nov 14 2006 17:20:03 MST: %IPS-6-ENGINE_READY: service-rpc - build time 80 ms - packets
for this engine will be scanned
*Nov 14 2006 17:20:03 MST: %IPS-6-ENGINE_BUILDING: service-dns - 38 signatures - 10 of 13
engines
*Nov 14 2006 17:20:03 MST: %IPS-6-ENGINE_READY: service-dns - build time 20 ms - packets
for this engine will be scanned
*Nov 14 2006 17:20:03 MST: %IPS-6-ENGINE_BUILDING: normalizer - 9 signatures - 11 of 13
engines
*Nov 14 2006 17:20:03 MST: %IPS-6-ENGINE_READY: normalizer - build time 0 ms - packets
for this engine will be scanned
*Nov 14 2006 17:20:03 MST: %IPS-6-ENGINE_BUILDING: service-msrpc - 22 signatures - 12 of
13 engines
*Nov 14 2006 17:20:03 MST: %IPS-6-ENGINE_READY: service-msrpc - build time 8 ms - packets
for this engine will be scanned
*Nov 14 2006 17:20:03 MST: %IPS-6-ALL_ENGINE_BUILDS_COMPLETE: elapsed time 16344 ms
Router#
Router#
Router# show ip ips signature count
Cisco SDF release version S258.0
Signature Micro-Engine: multi-string
        Total Signatures: 3
        Enabled: 3

```

```

    Retired: 3
Signature Micro-Engine: service-http
  Total Signatures: 611
    Enabled: 159
    Retired: 428
    Compiled: 183
Signature Micro-Engine: string-tcp
  Total Signatures: 864
    Enabled: 414
    Retired: 753
    Compiled: 111
Signature Micro-Engine: string-udp
  Total Signatures: 74
    Enabled: 1
    Retired: 44
    Compiled: 30
Signature Micro-Engine: state
  Total Signatures: 28
    Enabled: 16
    Retired: 25
    Compiled: 3
Signature Micro-Engine: atomic-ip
  Total Signatures: 252
    Enabled: 56
    Retired: 148
    Compiled: 103
    Inactive - invalid params: 1
Signature Micro-Engine: string-icmp
  Total Signatures: 3
    Enabled: 0
    Retired: 2
    Compiled: 1
Signature Micro-Engine: service-ftp
  Total Signatures: 3
    Enabled: 1
    Compiled: 3
Signature Micro-Engine: service-rpc
  Total Signatures: 75
    Enabled: 44
    Retired: 44
    Compiled: 31
Signature Micro-Engine: service-dns
  Total Signatures: 38
    Enabled: 30
    Retired: 5
    Compiled: 33
Signature Micro-Engine: normalizer
  Total Signatures: 9
    Enabled: 8
    Retired: 5
    Compiled: 4
Signature Micro-Engine: service-smb-advanced (INACTIVE)
Signature Micro-Engine: service-msrpc
  Total Signatures: 22
    Enabled: 22
    Retired: 22

```

Example VRF Aware Cisco IOS IPS Configuration Without Subinterfaces

The following example shows a physical interface supporting a VRF forwarding table with IPS enabled on VRF600:

```

Router(config)# ip vrf
  VRF600
Router(config-vrf)# rd
  100:600
Router(config-vrf)# route-target export
  100:600
Router(config-vrf)# route-target import
  100:600
Router(config-vrf)# exit

```

```

Router(config)# interface FastEthernet
0/1
Router(config-subif)# ip vrf forwarding
VRF600
Router(config-subif)# ip address
192.168.00.3 192.168.255.225
Router(config-subif)# ip ips
ips_policy600 in

router(config-subif)# exit

router(config-if)# end

```

Example VRF Aware Cisco IOS IPS Configuration with Subinterfaces

The following example shows two physical interfaces supporting two VRF forwarding tables, VRF600 and VRF601, with IPS enabled on only VRF600:

```

config terminal
Router1(config)# ip vrf
VRF600
Router1(config-vrf)# rd
100:600
Router1(config-vrf)# route-target export
100:600
Router1(config-vrf)# route-target import
100:600
Router1(config-vrf)# exit
Router1(config)# ip vrf
VRF601
Router1(config-vrf)# rd
100:601
Router1(config-vrf)# route-target export
100:601
Router1(config-vrf)# route-target import
100:601
Router1(config-vrf)# exit
Router1(config)# interface FastEthernet
0/0.600
Router1(config-subif)# encapsulation dot1Q
600
Router1(config-subif)# ip vrf forwarding
VRF600
Router1(config-subif)# ip address
192.168.00.0 192.168.255.0
Router1(config-subif)# exit
Router1(config)# interface FastEthernet
0/0.601
Router1(config-subif)# encapsulation dot1Q
601
Router1(config-subif)# ip vrf forwarding
VRF601
Router1(config-subif)# ip address
192.168.00.0 192.168.255.0
Router1(config-subif)# end

```

```

config terminal
Router2(config)# ip ips name
ips_policy600

Router2(config)# ip vrf
VRF600
Router2(config-vrf)# rd
100:600
Router2(config-vrf)# route-target export
100:600
Router2(config-vrf)# route-target import

```

```

100:600
Router2(config-vrf)# exit

Router2(config)# ip vrf
VRF601
Router2(config-vrf)# rd
100:601
Router2(config-vrf)# route-target export
100:601
Router2(config-vrf)# route-target import
100:601
Router2(config-vrf)# exit

Router2(config)# interface FastEthernet
0/1.600
Router2(config-subif)# encapsulation dot1Q
600
Router2(config-subif)# ip vrf forwarding
VRF600
Router2(config-subif)# ip address
192.168.00.0 192.168.255.0
Router2(config-subif)# ip ips
ips_policy600 in
Router2(config-subif)# exit

Router2(config)# interface FastEthernet
0/1.601
Router2(config-subif)# encapsulation dot1Q
601
Router2(config-subif)# ip vrf forwarding
VRF601
Router2(config-subif)# ip address
192.168.00.0 192.168.255.0
Router2(config-subif)# end

```

Example Multi VRF with IPS and Zone Based Policy (ZBP) Firewall

The following example shows Multiple VRFs configured with IPS and ZBP firewalls:

```

ip cef
!
ip vrf VRF_600
rd 100:110
route-target export 100:1000
route-target import 100:1000
!
ip vrf VRF_601
rd 100:120
route-target export 100:2000
route-target import 100:2000
!
ip ips config location flash:ips5/ retries 1
ip ips name IPS_POLICY_201
ip ips name IPS_POLICY_VRF_600
ip ips name IPS_POLICY_VRF_601
!
ip ips signature-category
category all
retired true
!
crypto key pubkey-chain rsa
named-key realm-cisco.pub signature
key-string
30820122 300D0609 2A864886 F70D0101 01050003 82010F00 3082010A 02820101
00C19E93 A8AF124A D6CC7A24 5097A975 206BE3A2 06FBA13F 6F12CB5B 4E441F16
17E630D5 C02AC252 912BE27F 37FDD9C8 11FC7AF7 DCDD81D9 43CDABC3 6007D128
B199ABCB D34ED0F9 085FADC1 359C189E F30AF10A C0EFB624 7E0764BF 3E53053E
5B2146A9 D7A5EDE3 0298AF03 DED7A5B8 9479039D 20F30663 9AC64B93 C0112A35
FE3F0C87 89BCB7BB 994AE74C FA9E481D F65875D6 85EAF974 6D9CC8E3 F0B08B85
50437722 FFBEB85B9 5E4189FF CC189CB9 69C46F9C A84DFBA5 7A0AF99E AD768C36
006CF498 079F88F8 A3B3FB1F 9FB7B3CB 5539E1D1 9693CCBB 551F78D2 892356AE

```

```

    2F56D826 8918EF3C 80CA4F4D 87BFCA3B BFF668E9 689782A5 CF31CB6E B4B094D3
    F3020301 0001
quit
!
class-map type inspect match-any L4-cmti
match protocol tcp
match protocol udp
match protocol icmp
!
policy-map type inspect inside201-outside-pmti
class type inspect L4-cmti
inspect
!
policy-map type inspect inside600-outside-pmti
class type inspect L4-cmti
inspect
!
policy-map type inspect inside602-outside-pmti
class type inspect L4-cmti
inspect
!
zone security inside201
zone security inside600
zone security inside601
zone security outside
!
zone-pair security inside201-outside source inside201 destination outside
service-policy type inspect inside201-outside-pmti
!
zone-pair security inside600-outside source inside600 destination outside
service-policy type inspect inside600-outside-pmti
!
zone-pair security inside601-outside source inside602 destination outside
service-policy type inspect inside602-outside-pmti
!
interface Loopback0
ip address 10.10.10.4 10.255.255.255
ip router isis
!
interface GigabitEthernet0/0
no ip address
duplex auto
speed auto
media-type rj45
no keepalive
!
interface GigabitEthernet0/0.201
encapsulation dot1Q 201
ip address 192.168.00.0 192.168.255.0
zone-member security inside201
ip ips myips201 in
ip ips myips201 out
!
interface GigabitEthernet0/0.600
encapsulation dot1Q 600
ip vrf forwarding VRF_600
ip address 10.0.0.0 10.255.255.255
zone-member security inside600
ip ips IPS_POLICY_VRF_600 in
ip ips IPS_POLICY_VRF_600 out
!
interface GigabitEthernet0/0.601
encapsulation dot1Q 601
ip vrf forwarding IPS_POLICY_VRF_601
ip address 10.0.0.0 10.255.255.255
zone-member security inside602
ip ips IPS_POLICY_VRF_601 in
ip ips IPS_POLICY_VRF_601 out
!
!
interface FastEthernet2/0
ip address 10.1.1.14 10.255.255.255
ip router isis

```

```

duplex auto
speed auto
mpls ip
!
router isis
net 10.225.225.225
is-type level-1
!
router bgp 100
no synchronization
bgp log-neighbor-changes
neighbor 10.10.10.6 remote-as 100
neighbor 10.10.10.6 update-source Loopback0
no auto-summary
!
address-family vpnv4
neighbor 10.10.10.6 activate
neighbor 10.10.10.6 send-community both
exit-address-family
!
address-family ipv4 vrf VRF_600
redistribute connected
no synchronization
exit-address-family
!
address-family ipv4 vrf VRF_601
redistribute connected
no synchronization
exit-address-family

```

Examples VRF Aware Cisco IOS IPS Output and Error Message



Note

All VRFs will share the same global IPS configurations, therefore, some show commands which show the information of the global items are shown for every VRF irrespective of whether the event happened on that particular VRF.

- [Examples VRF Aware Cisco IOS IPS Output, page 97](#)
- [Examples ErrMSG with VRF Name Output, page 98](#)
- [Examples SDEE Messages with VRF Name, page 98](#)
- [Examples SDEE show Commands, page 99](#)

Examples VRF Aware Cisco IOS IPS Output

The following is sample output from the **show ip ips statistics** command. The output provides statistics that may not necessarily be the ones that fired on VRF 600.

```

Router# show ip ips statistics vrf VRF_600
Signature statistics [process switch:fast switch]
signature 5170:1 packets checked: [0:4]
Interfaces configured for ips 4
Session creations since subsystem startup or last reset 0
Current session counts (estab/half-open/terminating) [0:0:0]
Maxever session counts (estab/half-open/terminating) [0:0:0]
Last session created never
Last statistic reset never
TCP reassembly statistics
received 0 packets out-of-order; dropped 0

```

```
peak memory usage 0 KB; current usage: 0 KB
peak queue length 0
```

The following is sample output from the **show ip ips interfaces** command.

```
Router# show ip ips interface
Interface Configuration
Interface GigabitEthernet0/0
VRF name: vrf1
Inbound IPS rule is tst
Outgoing IPS rule is not set
```

The following is sample output from the **show ip ips session** command.

```
Router# show ip ips session vrf vrf1
Established Sessions
Session 485EBEE8 (172.16.0.0:10001)=>(172.31.255.255:80) tcp SIS_OPEN
```

The following is sample output from the **clear ip ips statistics** command.

```
Router# clear ip ips statistics vrf vrf1
Router# show ip ips stat vrf vrf1

Interfaces configured for ips 1
Session creations since subsystem startup or last reset 0
Current session counts (estab/half-open/terminating) [1:0:0]
Maxever session counts (estab/half-open/terminating) [1:0:0]
Last session created 00:00:26
Last statistic reset 00:00:01
```

Examples ErrMSG with VRF Name Output

The following is sample error message output with the VRF name.

```
%IPS-4-SIGNATURE: Sig:5405 Subsig:0 Sev:100 [192.168.103.1:51129 -> 192.168.3.4:80]
VRF:vrf600 RiskRating:100
```

Examples SDEE Messages with VRF Name

The SDEE messages have been enhanced to show the VRF name. An example of output from the browser is as follows:

```
<?xml version="1.0" encoding="UTF-8" standalone="yes" ?>
  <env:Envelope xmlns:env="http://www.w3.org/2003/05/soap-envelope">
    <env:Body>
      <sd:events xmlns:cid="http://www.cisco.com/cids/2003/08/cidee" xmlns:sd="http://
example.org/2003/08/sdee" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="http://example.org/2003/08/sdee sdee.xsd http://www.cisco.com/cids/
2003/08/cidee cidee.xsd">
        <sd:evIdsAlert eventId="11630069224" vendor="Cisco" severity="unknown">
          <sd:originator>
            <sd:hostId>iosfw-28a</sd:hostId>
          </sd:originator>
          <sd:time offset="0" timeZone="UTC">116300692225223338</sd:time>
          <sd:signature description="" id="5123" version="S4">
            <cid:subsigId>0</cid:subsigId>
            <cid:sigDetails>Host:\x3c250+ chars></cid:sigDetails>
          </sd:signature>
          <cid:protocol>tcp</cid:protocol>
          <cid:riskRatingValue>85</cid:riskRatingValue>
          <sd:participants>
            <sd:attacker>
              <sd:addr>10.1.0.1</sd:addr>
              <sd:port>10001</sd:port>
            </sd:attacker>
          </sd:participants>
        </sd:events>
      </env:Body>
    </env:Envelope>
```

```

        <sd:target>
            <sd:addr>10.2.0.1</sd:addr>
            <sd:port>80</sd:port>
        </sd:target>
    </sd:participants>
    <sd:actions></sd:actions>
    <cid:interface>Gi0/1</cid:interface>
    <cid:vrf_name>vrf1</cid:vrf_name>
</sd:evIdsAlert>
</sd:events>
</env:Body>
</env:Envelope>

```

Examples SDEE show Commands

The SDEE show commands have been enhanced to include VRF specific information, the following is sample output from the **show ip sdee alerts** command:

```
Router# show ip sdee alerts
```

```
Alert storage: 200 alerts using 56000 bytes of memory
SDEE Alerts
  SigID   Sig Name                SrcIP:SrcPort          DstIP:DstPort          VRF
                or Summary Info
1:  5170:1                192.162.4.0:3692      192.162.6.0:80        NONE
2:  5170:1                192.162.4.0:3692      192.162.6.0:80      VRF_600
```

The following is sample output from the **show ip sdee events** command.

```
Router# show ip sdee events
```

```
Alert storage: 200 alerts using 56000 bytes of memory
Message storage: 200 messages using 84800 bytes of memory
SDEE Events
  Time                Type      Description
1: 10:25:55 MST Jan 22 2007 ALERT    Sig ID 5170:1
2: 10:25:57 MST Jan 22 2007 ALERT    Sig ID 5170:1
```

Additional References

Related Documents

Related Topic	Document Title
Cisco IOS commands	Cisco IOS Master Commands List, All Releases
Security commands	<ul style="list-style-type: none"> Cisco IOS Security Command Reference: Commands A to C Cisco IOS Security Command Reference: Commands D to L Cisco IOS Security Command Reference: Commands M to R Cisco IOS Security Command Reference: Commands S to Z

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 VRF Aware Cisco IOS IPS

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 VRF Aware Cisco IOS IPS**

Feature Name	Releases	Feature Information
VRF aware Cisco IOS IPS	12.4(20)T	<p data-bbox="1154 348 1516 1073">Virtual Route Forwarding or Virtual Private Network (VPN) Route Forwarding (VRF), is a mechanism that allows multiple instances of a routing table to exist on a router and work simultaneously. This mechanism allows for network paths to be segregated without using multiple devices, thereby increasing network security and eliminating the need for encryption and authentication. VRFs are generally used to create separate VPNs. Allowing Intrusion Prevention System (IPS) to be configured on a per-VRF basis means global parameters will be shared by multiple VPNs, providing VRF related information on the Security Device Event Exchange (SDEE) and syslog alerts.</p> <p data-bbox="1154 1094 1516 1182">The following commands were introduced or modified: clear ip ips statistics, show ip ips</p>

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

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

