



Cisco Intrusion Prevention System CLI Configuration Guide for IPS 7.3

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

Text Part Number: OL-30788-01

THE SPECIFICATIONS AND INFORMATION REGARDING THE PRODUCTS IN THIS MANUAL ARE SUBJECT TO CHANGE WITHOUT NOTICE. ALL STATEMENTS, INFORMATION, AND RECOMMENDATIONS IN THIS MANUAL ARE BELIEVED TO BE ACCURATE BUT ARE PRESENTED WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED. USERS MUST TAKE FULL RESPONSIBILITY FOR THEIR APPLICATION OF ANY PRODUCTS.

THE SOFTWARE LICENSE AND LIMITED WARRANTY FOR THE ACCOMPANYING PRODUCT ARE SET FORTH IN THE INFORMATION PACKET THAT SHIPPED WITH THE PRODUCT AND ARE INCORPORATED HEREIN BY THIS REFERENCE. IF YOU ARE UNABLE TO LOCATE THE SOFTWARE LICENSE OR LIMITED WARRANTY, CONTACT YOUR CISCO REPRESENTATIVE FOR A COPY.

The Cisco implementation of TCP header compression is an adaptation of a program developed by the University of California, Berkeley (UCB) as part of UCB's public domain version of the UNIX operating system. All rights reserved. Copyright © 1981, Regents of the University of California.

NOTWITHSTANDING ANY OTHER WARRANTY HEREIN, ALL DOCUMENT FILES AND SOFTWARE OF THESE SUPPLIERS ARE PROVIDED "AS IS" WITH ALL FAULTS. CISCO AND THE ABOVE-NAMED SUPPLIERS DISCLAIM ALL WARRANTIES, EXPRESSED OR IMPLIED, INCLUDING, WITHOUT LIMITATION, THOSE OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT OR ARISING FROM A COURSE OF DEALING, USAGE, OR TRADE PRACTICE.

IN NO EVENT SHALL CISCO OR ITS SUPPLIERS BE LIABLE FOR ANY INDIRECT, SPECIAL, CONSEQUENTIAL, OR INCIDENTAL DAMAGES, INCLUDING, WITHOUT LIMITATION, LOST PROFITS OR LOSS OR DAMAGE TO DATA ARISING OUT OF THE USE OR INABILITY TO USE THIS MANUAL, EVEN IF CISCO OR ITS SUPPLIERS HAVE BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

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 used in this document are not intended to be actual addresses. Any examples, command display output, and figures included in the document are shown for illustrative purposes only. Any use of actual IP addresses in illustrative content is unintentional and coincidental.

Cisco Intrusion Prevention System CLI Configuration Guide for IPS 7.3
© 2014 Cisco Systems, Inc. All rights reserved.



Preface xxiii

Contents xxiii

Audience xxiii

Organization xxiii

Conventions xxv

Related Documentation xxvi

Obtaining Documentation, Using Cisco Bug Search Tool, and Submitting a Service Request xxvi

CHAPTER 1

Introducing the CLI Configuration Guide 1-1

Supported IPS Platforms 1-1

IPS CLI Configuration Guide 1-1

Sensor Configuration Sequence 1-2

User Roles 1-3

CLI Behavior 1-5

Command Line Editing 1-6

IPS Command Modes 1-7

Regular Expression Syntax 1-8

Generic CLI Commands 1-10

CLI Keywords 1-10

CHAPTER 2

Logging In to the Sensor 2-1

Logging In Notes and Caveats 2-1

Supported User Roles 2-1

Logging In to the Appliance 2-2

Connecting an Appliance to a Terminal Server 2-3

Logging In to the ASA 5500-X IPS SSP 2-4

Logging In to the ASA 5585-X IPS SSP 2-5

Logging In to the Sensor 2-6

CHAPTER 3

Initializing the Sensor 3-1

Initializing Notes and Caveats 3-1

- Understanding Initialization 3-1
- Participating in the SensorBase Network 3-2
- Simplified Setup Mode 3-3
- System Configuration Dialog 3-3
- Basic Sensor Setup 3-5
- Advanced Setup 3-8
 - Advanced Setup for the Appliance 3-9
 - Advanced Setup for the ASA 5500-X IPS SSP 3-14
 - Advanced Setup for the ASA 5585-X IPS SSP 3-15
- Verifying Initialization 3-16

CHAPTER 4

Setting Up the Sensor 4-1

- Setup Notes and Caveats 4-1
- Understanding Sensor Setup 4-2
- Changing Network Settings 4-2
 - Changing the Hostname 4-3
 - Changing the IP Address, Netmask, and Gateway 4-4
 - Enabling and Disabling Telnet 4-5
 - Changing the Access List 4-6
 - Changing the FTP Timeout 4-8
 - Adding a Login Banner 4-9
 - Configuring the DNS and Proxy Servers for Global Correlation and Automatic Update 4-10
 - Enabling SSHv1 Fallback 4-13
- Changing the CLI Session Timeout 4-14
- Changing Web Server Settings 4-15
- Configuring Authentication and User Parameters 4-18
 - Adding and Removing Users 4-18
 - Configuring Authentication 4-20
 - Configuring Packet Command Restriction 4-26
 - Creating the Service Account 4-28
 - The Service Account and RADIUS Authentication 4-29
 - RADIUS Authentication Functionality and Limitations 4-30
 - Configuring Passwords 4-30
 - Changing User Privilege Levels 4-31
 - Showing User Status 4-32
 - Configuring the Password Policy 4-32
 - Locking User Accounts 4-34
 - Unlocking User Accounts 4-35

Configuring Time	4-36
Time Sources and the Sensor	4-36
Synchronizing IPS Module System Clocks with the Parent Device System Clock	4-37
Correcting Time on the Sensor	4-37
Configuring Time on the Sensor	4-37
Displaying the System Clock	4-37
Manually Setting the System Clock	4-38
Configuring Recurring Summertime Settings	4-39
Configuring Nonrecurring Summertime Settings	4-41
Configuring Time Zones Settings	4-43
Configuring NTP	4-43
Configuring a Cisco Router to be an NTP Server	4-43
Configuring the Sensor to Use an NTP Time Source	4-45
Configuring SSH	4-46
Understanding SSH	4-47
Adding Hosts to the SSH Known Hosts List	4-47
Adding Authorized Keys	4-49
Generating the RSA Server Host Key	4-50
Configuring TLS	4-51
Understanding TLS	4-52
Adding TLS Trusted Hosts	4-53
Enabling Strict TLS Certificate Checks	4-54
Adding and Updating TLS Trusted Root Certificates	4-55
Displaying TLS Trusted Root Certificates	4-57
Displaying and Generating the Server Certificate	4-58
Installing the License Key	4-58
Understanding the License Key	4-59
Service Programs for IPS Products	4-59
Obtaining and Installing the License Key	4-60
Licensing the ASA 5500-X IPS SSP	4-62
Uninstalling the License Key	4-62

CHAPTER 5**Configuring Interfaces 5-1**

Interface Notes and Caveats	5-1
Understanding Interfaces	5-2
IPS Interfaces	5-2
Command and Control Interface	5-3
Sensing Interfaces	5-4
TCP Reset Interfaces	5-4

- Understanding Alternate TCP Reset Interfaces 5-4
 - Designating the Alternate TCP Reset Interface 5-5
 - Interface Support 5-6
 - Interface Configuration Restrictions 5-8
 - Interface Configuration Sequence 5-10
- Configuring Physical Interfaces 5-11
- Configuring Promiscuous Mode 5-14
 - Understanding Promiscuous Mode 5-14
 - Configuring Promiscuous Mode 5-15
 - IPv6, Switches, and Lack of VACL Capture 5-15
- Configuring Inline Interface Mode 5-16
 - Understanding Inline Interface Mode 5-16
 - Configuring Inline Interface Pairs 5-17
- Configuring Inline VLAN Pair Mode 5-21
 - Understanding Inline VLAN Pair Mode 5-21
 - Configuring Inline VLAN Pairs 5-22
- Configuring VLAN Group Mode 5-26
 - Understanding VLAN Group Mode 5-26
 - Deploying VLAN Groups 5-27
 - Configuring VLAN Groups 5-28
- Configuring LACP for the 4500 Series Sensors 5-33
 - Understanding ECLB Using LACP 5-33
 - LACP Restrictions 5-35
 - Understanding Failover/Fallback 5-36
 - LACP Link States 5-36
 - Configuring LACP 5-36
 - Displaying LACP 5-38
- Configuring Inline Bypass Mode 5-42
 - Understanding Inline Bypass Mode 5-42
 - Configuring Inline Bypass Mode 5-43
- Configuring Interface Notifications 5-44
- Configuring CDP Mode 5-45
- Displaying Interface Statistics 5-46
- Displaying Interface Traffic History 5-48

CHAPTER 6

Configuring Virtual Sensors 6-1

- Virtual Sensor Notes and Caveats 6-1
- Understanding the Analysis Engine 6-2

Understanding Virtual Sensors	6-2
Advantages and Restrictions of Virtualization	6-2
Inline TCP Session Tracking Mode	6-3
Normalization and Inline TCP Evasion Protection Mode	6-4
HTTP Advanced Decoding	6-4
Adding, Editing, and Deleting Virtual Sensors	6-5
Adding Virtual Sensors	6-5
Editing and Deleting Virtual Sensors	6-9
Configuring Global Variables	6-12

CHAPTER 7**Defining Signatures 7-1**

Signature Definition Notes and Caveats	7-1
Understanding Policies	7-2
Working With Signature Definition Policies	7-2
Understanding Signatures	7-3
Configuring Threat Profiles	7-4
Understanding Threat Profiles	7-4
Displaying, Applying, and Removing Threat Profiles	7-5
Displaying a Signature Tuned with a Threat Profile	7-8
Displaying Threat Profile Versions	7-8
Configuring Signature Variables	7-9
Understanding Signature Variables	7-9
Creating Signature Variables	7-10
Configuring Signatures	7-11
Signature Definition Options	7-11
Configuring Alert Frequency	7-12
Configuring Alert Severity	7-14
Configuring the Event Counter	7-15
Configuring Signature Fidelity Rating	7-17
Configuring the Status of Signatures	7-18
Configuring the Vulnerable OSeS for a Signature	7-19
Assigning Actions to Signatures	7-20
Configuring AIC Signatures	7-22
Understanding the AIC Engine	7-22
AIC Engine and Sensor Performance	7-23
Configuring the Application Policy	7-23
AIC Request Method Signatures	7-25
AIC MIME Define Content Type Signatures	7-26

- AIC Transfer Encoding Signatures 7-30
- AIC FTP Commands Signatures 7-31
- Creating an AIC Signature 7-32
- Configuring IP Fragment Reassembly 7-34
 - Understanding IP Fragment Reassembly 7-34
 - IP Fragment Reassembly Signatures and Configurable Parameters 7-34
 - Configuring IP Fragment Reassembly Parameters 7-36
 - Configuring the Method for IP Fragment Reassembly 7-36
- Configuring TCP Stream Reassembly 7-37
 - Understanding TCP Stream Reassembly 7-37
 - TCP Stream Reassembly Signatures and Configurable Parameters 7-38
 - Configuring TCP Stream Reassembly Signatures 7-43
 - Configuring the Mode for TCP Stream Reassembly 7-44
- Configuring IP Logging 7-45
- Creating Custom Signatures 7-46
 - Sequence for Creating a Custom Signature 7-46
 - Example String TCP Engine Signature 7-47
 - Example Service HTTP Engine Signature 7-50
 - Example Meta Engine Signature 7-52
 - Example IPv6 Engine Signature 7-57
 - Example String XL TCP Engine Match Offset Signature 7-58
 - Example String XL TCP Engine Minimum Match Length Signature 7-61

CHAPTER 8

Configuring Event Action Rules 8-1

- Event Action Rules Notes and Caveats 8-1
- Understanding Security Policies 8-2
- Understanding Event Action Rules 8-2
- Signature Event Action Processor 8-3
- Event Actions 8-5
- Event Action Rules Configuration Sequence 8-8
- Working With Event Action Rules Policies 8-8
- Event Action Variables 8-10
 - Understanding Event Action Variables 8-10
 - Adding, Editing, and Deleting Event Action Variables 8-11
- Configuring Target Value Ratings 8-13
 - Calculating the Risk Rating 8-13
 - Understanding Threat Rating 8-15
 - Adding, Editing, and Deleting Target Value Ratings 8-15

Configuring Event Action Overrides	8-17
Understanding Event Action Overrides	8-17
Adding, Editing, Enabling, and Disabling Event Action Overrides	8-18
Configuring Event Action Filters	8-20
Understanding Event Action Filters	8-21
Configuring Event Action Filters	8-21
Configuring OS Identifications	8-26
Understanding Passive OS Fingerprinting	8-26
Passive OS Fingerprinting Configuration Considerations	8-28
Adding, Editing, Deleting, and Moving Configured OS Maps	8-28
Displaying and Clearing OS Identifications	8-32
Configuring General Settings	8-33
Understanding Event Action Summarization	8-33
Understanding Event Action Aggregation	8-34
Configuring the General Settings	8-34
Configuring the Denied Attackers List	8-36
Adding a Deny Attacker Entry to the Denied Attackers List	8-36
Monitoring and Clearing the Denied Attackers List	8-37
Monitoring Events	8-39
Displaying Events	8-39
Clearing Events from Event Store	8-42

CHAPTER 9

Configuring Anomaly Detection	9-1
Anomaly Detection Notes and Caveats	9-1
Understanding Security Policies	9-2
Understanding Anomaly Detection	9-2
Understanding Worms	9-2
Anomaly Detection Modes	9-3
Anomaly Detection Zones	9-4
Anomaly Detection Configuration Sequence	9-5
Anomaly Detection Signatures	9-6
Enabling Anomaly Detection	9-8
Working With Anomaly Detection Policies	9-8
Configuring Anomaly Detection Operational Settings	9-10
Configuring the Internal Zone	9-11
Understanding the Internal Zone	9-12
Configuring the Internal Zone	9-12
Configuring TCP Protocol for the Internal Zone	9-13

- Configuring UDP Protocol for the Internal Zone 9-15
- Configuring Other Protocols for the Internal Zone 9-18
- Configuring the Illegal Zone 9-20
 - Understanding the Illegal Zone 9-20
 - Configuring the Illegal Zone 9-20
 - Configuring TCP Protocol for the Illegal Zone 9-21
 - Configuring UDP Protocol for the Illegal Zone 9-24
 - Configuring Other Protocols for the Illegal Zone 9-26
- Configuring the External Zone 9-28
 - Understanding the External Zone 9-28
 - Configuring the External Zone 9-28
 - Configuring TCP Protocol for the External Zone 9-29
 - Configuring UDP Protocol for the External Zone 9-32
 - Configuring Other Protocols for the External Zone 9-34
- Configuring Learning Accept Mode 9-36
 - The KB and Histograms 9-36
 - Configuring Learning Accept Mode 9-38
- Working With KB Files 9-40
 - Displaying KB Files 9-40
 - Saving and Loading KBs Manually 9-41
 - Copying, Renaming, and Erasing KBs 9-42
 - Displaying the Differences Between Two KBs 9-44
 - Displaying the Thresholds for a KB 9-45
- Displaying Anomaly Detection Statistics 9-47
- Disabling Anomaly Detection 9-48

CHAPTER 10

- Configuring Global Correlation 10-1**
 - Global Correlation Notes and Caveats 10-1
 - Understanding Global Correlation 10-2
 - Participating in the SensorBase Network 10-2
 - Understanding Reputation 10-3
 - Understanding Network Participation 10-4
 - Understanding Efficacy 10-5
 - Understanding Reputation and Risk Rating 10-6
 - Global Correlation Features and Goals 10-6
 - Global Correlation Requirements 10-7
 - Understanding Global Correlation Sensor Health Metrics 10-8
 - Configuring Global Correlation Inspection and Reputation Filtering 10-8

Understanding Global Correlation Inspection and Reputation Filtering	10-9
Configuring Global Correlation Inspection and Reputation Filtering	10-10
Configuring Network Participation	10-11
Troubleshooting Global Correlation	10-13
Disabling Global Correlation	10-13
Displaying Global Correlation Statistics	10-14

CHAPTER 11**Configuring External Product Interfaces** 11-1

External Product Interface Notes and Caveats	11-1
Understanding External Product Interfaces	11-1
Understanding the CSA MC	11-2
External Product Interface Issues	11-3
Configuring the CSA MC to Support the IPS Interface	11-4
Adding External Product Interfaces and Posture ACLs	11-4
Troubleshooting External Product Interfaces	11-8

CHAPTER 12**Configuring IP Logging** 12-1

IP Logging Notes and Caveats	12-1
Understanding IP Logging	12-2
Configuring Automatic IP Logging	12-2
Configuring Manual IP Logging for a Specific IP Address	12-3
Displaying the Contents of IP Logs	12-5
Stopping Active IP Logs	12-6
Copying IP Log Files to Be Viewed	12-7

CHAPTER 13**Displaying and Capturing Live Traffic on an Interface** 13-1

Packet Display And Capture Notes and Caveats	13-1
Understanding Packet Display and Capture	13-2
Displaying Live Traffic on an Interface	13-2
Capturing Live Traffic on an Interface	13-4
Copying the Packet File	13-6
Erasing the Packet File	13-7

CHAPTER 14**Configuring Attack Response Controller for Blocking and Rate Limiting** 14-1

Blocking Notes and Caveats	14-1
Understanding Blocking	14-2

- Understanding Rate Limiting 14-4
- Understanding Service Policies for Rate Limiting 14-5
- Before Configuring ARC 14-5
- Supported Devices 14-6
- Configuring Blocking Properties 14-7
 - Allowing the Sensor to Block Itself 14-8
- Disabling Blocking 14-9
 - Specifying Maximum Block Entries 14-11
 - Specifying the Block Time 14-13
 - Enabling ACL Logging 14-14
 - Enabling Writing to NVRAM 14-15
 - Logging All Blocking Events and Errors 14-16
 - Configuring the Maximum Number of Blocking Interfaces 14-17
 - Configuring Addresses Never to Block 14-19
- Configuring User Profiles 14-20
- Configuring Blocking and Rate Limiting Devices 14-21
 - How the Sensor Manages Devices 14-21
 - Configuring the Sensor to Manage Cisco Routers 14-23
 - Routers and ACLs 14-23
 - Configuring the Sensor to Manage Cisco Routers 14-23
 - Configuring the Sensor to Manage Catalyst 6500 Series Switches and Cisco 7600 Series Routers 14-25
 - Switches and VACLs 14-25
 - Configuring the Sensor to Manage Catalyst 6500 Series Switches and Cisco 7600 Series Routers 14-26
 - Configuring the Sensor to Manage Cisco Firewalls 14-27
- Configuring the Sensor to be a Master Blocking Sensor 14-28
- Configuring Host Blocking 14-31
- Configuring Network Blocking 14-32
- Configuring Connection Blocking 14-33
- Obtaining a List of Blocked Hosts and Connections 14-34

CHAPTER 15

- Configuring SNMP 15-1**
 - SNMP Notes and Caveats 15-1
 - Understanding SNMP 15-1
 - Configuring SNMP 15-2
 - Configuring SNMPv3 Users 15-4
 - Displaying SNMPv3 Users and Engine IDs 15-7

Configuring SNMP Traps 15-8

Supported MIBS 15-11

CHAPTER 16

Working With Configuration Files 16-1

Displaying the Current Configuration 16-1

Displaying the Current Submode Configuration 16-3

Filtering the Current Configuration Output 16-16

Filtering the Current Submode Configuration Output 16-18

Displaying the Contents of a Logical File 16-19

Backing Up and Restoring the Configuration File Using a Remote Server 16-21

Creating and Using a Backup Configuration File 16-24

Erasing the Configuration File 16-24

CHAPTER 17

Administrative Tasks for the Sensor 17-1

Administrative Notes and Caveats 17-2

Recovering the Password 17-2

 Understanding Password Recovery 17-2

 Recovering the Password for the Appliance 17-3

 Using the GRUB Menu 17-3

 Using ROMMON 17-4

 Recovering the Password for the ASA 5500-X IPS SSP 17-4

 Recovering the Password for the ASA 5585-X IPS SSP 17-6

 Disabling Password Recovery 17-8

 Verifying the State of Password Recovery 17-9

 Troubleshooting Password Recovery 17-9

Clearing the Sensor Databases 17-10

Displaying and Deleting SDEE Server Subscriptions 17-11

Displaying the Inspection Load of the Sensor 17-12

Configuring Health Status Information 17-14

Showing Sensor Overall Health Status 17-19

Creating a Banner Login 17-20

Terminating CLI Sessions 17-20

Modifying Terminal Properties 17-21

Configuring Events 17-22

 Displaying Events 17-22

 Clearing Events from the Event Store 17-25

Configuring the System Clock 17-25

- Displaying the System Clock 17-25
- Manually Setting the System Clock 17-26
- Displaying and Clearing the Denied Attackers List 17-27
- Displaying Policy Lists 17-29
- Displaying Statistics 17-30
- Displaying Tech Support Information 17-41
- Displaying Version Information 17-43
- Diagnosing Network Connectivity 17-45
- Resetting the Appliance 17-46
- Displaying Command History 17-47
- Displaying Hardware Inventory 17-47
- Tracing the Route of an IP Packet 17-48
- Displaying Submode Settings 17-49

CHAPTER 18

- Configuring the ASA 5500-X IPS SSP 18-1**
 - Notes and Caveats for ASA 5500-X IPS SSP 18-1
 - Configuration Sequence for the ASA 5500-X IPS SSP 18-2
 - Verifying Initialization for the ASA 5500-X IPS SSP 18-3
 - Creating Virtual Sensors for the ASA 5500-X IPS SSP 18-3
 - The ASA 5500-X IPS SSP and Virtualization 18-4
 - Virtual Sensor Configuration Sequence for ASA 5500-X IPS SSP 18-4
 - Creating Virtual Sensors 18-4
 - Assigning Virtual Sensors to Adaptive Security Appliance Contexts 18-6
 - The ASA 5500-X IPS SSP and Bypass Mode 18-9
 - The ASA 5500-X IPS SSP and the Normalizer Engine 18-9
 - The ASA 5500-X IPS SSP and Jumbo Packets 18-10
 - The ASA 5500-X IPS SSP and Memory Usage 18-11
 - TCP Reset Differences Between IPS Appliances and the ASA 5500-X IPS SSP 18-11
 - Reloading IPS Messages 18-11
 - Reloading, Shutting Down, Resetting, and Recovering the ASA 5500-X IPS SSP 18-12
 - Health and Status Information 18-13
 - ASA 5500-X IPS SSP Failover Scenarios 18-20
 - New and Modified Commands 18-21

CHAPTER 19

- Configuring the ASA 5585-X IPS SSP 19-1**
 - ASA 5585-X IPS SSP Notes and Caveats 19-1

Configuration Sequence for the ASA 5585-X IPS SSP	19-2
Verifying Initialization for the ASA 5585-X IPS SSP	19-3
Creating Virtual Sensors for the ASA 5585-X IPS SSP	19-4
The ASA 5585-X IPS SSP and Virtualization	19-4
The ASA 5585-X IPS SSP Virtual Sensor Configuration Sequence	19-5
Creating Virtual Sensors	19-5
Assigning Virtual Sensors to Adaptive Security Appliance Contexts	19-7
The ASA 5585-X IPS SSP and the Normalizer Engine	19-10
The ASA 5585-X IPS SSP and Bypass Mode	19-10
ASA 5585-X IPS SSP and Jumbo Packets	19-11
TCP Reset Differences Between IPS Appliances and the ASA 5585-X IPS SSP	19-11
Reloading IPS Messages	19-12
Reloading, Shutting Down, Resetting, and Recovering the ASA 5585-X IPS SSP	19-12
Health and Status Information	19-13
Traffic Flow Stopped on IPS Switchports	19-16
Failover Scenarios	19-16

CHAPTER 20**Obtaining Software 20-1**

IPS 7.3 File List	20-1
Obtaining Cisco IPS Software	20-1
IPS Software Versioning	20-3
IPS Software Release Examples	20-5
Accessing IPS Documentation	20-7
Cisco Security Intelligence Operations	20-7

CHAPTER 21**Upgrading, Downgrading, and Installing System Images 21-1**

Upgrade Notes and Caveats	21-1
Upgrades, Downgrades, and System Images	21-2
Supported FTP and HTTP/HTTPS Servers	21-3
Upgrading the Sensor	21-3
IPS 7.3(1)E4 Files	21-3
Manually Upgrading the Sensor	21-4
Upgrading the Recovery Partition	21-6
Configuring Automatic Upgrades	21-7
Configuring Automatic Updates	21-8
Applying an Immediate Update	21-11
Downgrading the Sensor	21-12

- Recovering the Application Partition 21-14
- Installing System Images 21-15
 - ROMMON 21-15
 - TFTP Servers 21-15
 - Connecting an Appliance to a Terminal Server 21-16
 - Installing the System Image for the IPS 4300 Series Sensors 21-16
 - Installing the System Image for the IPS 4500 Series Sensors 21-20
 - Installing the System Image for the ASA 5500-X IPS SSP 21-22
 - Installing the System Image for the ASA 5585-X IPS SSP 21-23
 - Installing the ASA 5585-X IPS SSP System Image Using the hw-module Command 21-24
 - Installing the ASA 5585-X IPS SSP System Image Using ROMMON 21-26

APPENDIX A

System Architecture

- Understanding the IPS System Architecture A-1
- IPS System Design A-1
- System Applications A-3
- User Interaction A-5
- Security Features A-5
- MainApp A-6
 - Understanding the MainApp A-6
 - MainApp Responsibilities A-6
 - Event Store A-7
 - Understanding the Event Store A-7
 - Event Data Structures A-8
 - IPS Events A-9
 - NotificationApp A-9
 - CtlTransSource A-11
 - Attack Response Controller A-12
 - Understanding the ARC A-13
 - ARC Features A-14
 - Supported Blocking Devices A-15
 - ACLs and VACLs A-16
 - Maintaining State Across Restarts A-16
 - Connection-Based and Unconditional Blocking A-17
 - Blocking with Cisco Firewalls A-18
 - Blocking with Catalyst Switches A-19
 - Logger A-19
 - AuthenticationApp A-20
 - Understanding the AuthenticationApp A-20

Authenticating Users	A-20
Configuring Authentication on the Sensor	A-20
Managing TLS and SSH Trust Relationships	A-21
Web Server	A-22
SensorApp	A-22
Understanding the SensorApp	A-23
Inline, Normalization, and Event Risk Rating Features	A-24
SensorApp New Features	A-25
Packet Flow	A-25
Signature Event Action Processor	A-26
CollaborationApp	A-27
Understanding the CollaborationApp	A-28
Update Components	A-28
Error Events	A-29
SwitchApp	A-29
CLI	A-30
User Roles	A-30
Service Account	A-31
Communications	A-31
IDAPI	A-32
IDIOM	A-32
IDCONF	A-33
SDEE	A-33
CIDEE	A-34
Cisco IPS File Structure	A-34
Summary of Cisco IPS Applications	A-35

APPENDIX B

Signature Engines	B-1
Understanding Signature Engines	B-1
Master Engine	B-4
General Parameters	B-4
Alert Frequency	B-7
Event Actions	B-8
Regular Expression Syntax	B-9
AIC Engine	B-10
Understanding the AIC Engine	B-11
AIC Engine and Sensor Performance	B-11
AIC Engine Parameters	B-11

- Atomic Engine **B-14**
 - Atomic ARP Engine **B-14**
 - Atomic IP Advanced Engine **B-15**
 - Atomic IP Engine **B-25**
 - Atomic IPv6 Engine **B-28**
- Fixed Engine **B-29**
- Flood Engine **B-32**
- Meta Engine **B-33**
- Multi String Engine **B-35**
- Normalizer Engine **B-36**
- Service Engines **B-38**
 - Understanding the Service Engines **B-39**
 - Service DNS Engine **B-39**
 - Service FTP Engine **B-40**
 - Service Generic Engine **B-41**
 - Service H225 Engine **B-43**
 - Service HTTP Engine **B-45**
 - Service IDENT Engine **B-47**
 - Service MSRPC Engine **B-48**
 - Service MSSQL Engine **B-50**
 - Service NTP Engine **B-51**
 - Service P2P Engine **B-52**
 - Service RPC Engine **B-52**
 - Service SMB Advanced Engine **B-54**
 - Service SNMP Engine **B-56**
 - Service SSH Engine **B-57**
 - Service TNS Engine **B-58**
- State Engine **B-59**
- String Engines **B-61**
- String XL Engines **B-64**
- Sweep Engines **B-67**
 - Sweep Engine **B-67**
 - Sweep Other TCP Engine **B-69**
- Traffic Anomaly Engine **B-70**
- Traffic ICMP Engine **B-72**
- Trojan Engines **B-73**

APPENDIX C

Troubleshooting C-1

- Cisco Bug Search Tool **C-1**
- Preventive Maintenance **C-2**
 - Understanding Preventive Maintenance **C-2**
 - Creating and Using a Backup Configuration File **C-2**
 - Backing Up and Restoring the Configuration File Using a Remote Server **C-3**
 - Creating the Service Account **C-5**
- Disaster Recovery **C-6**
- Password Recovery **C-7**
 - Understanding Password Recovery **C-8**
 - Recovering the Password for the Appliance **C-8**
 - Using the GRUB Menu **C-8**
 - Using ROMMON **C-9**
 - Recovering the Password for the ASA 5500-X IPS SSP **C-10**
 - Recovering the Password for the ASA 5585-X IPS SSP **C-12**
 - Disabling Password Recovery **C-13**
 - Verifying the State of Password Recovery **C-14**
 - Troubleshooting Password Recovery **C-15**
- Time Sources and the Sensor **C-15**
 - Time Sources and the Sensor **C-15**
 - Synchronizing IPS Clocks with Parent Device Clocks **C-16**
 - Verifying the Sensor is Synchronized with the NTP Server **C-16**
 - Correcting Time on the Sensor **C-17**
- Advantages and Restrictions of Virtualization **C-17**
- Supported MIBs **C-18**
- When to Disable Anomaly Detection **C-18**
- The Analysis Engine Not Responding **C-19**
- Troubleshooting RADIUS Authentication **C-20**
- Troubleshooting Global Correlation **C-20**
- Troubleshooting External Product Interfaces **C-21**
 - External Product Interfaces Issues **C-21**
 - External Product Interfaces Troubleshooting Tips **C-22**
- Troubleshooting the Appliance **C-22**
 - Troubleshooting Loose Connections **C-23**
 - The Analysis Engine is Busy **C-23**
 - Communication Problems **C-24**
 - Cannot Access the Sensor CLI Through Telnet or SSH **C-24**
 - Correcting a Misconfigured Access List **C-26**

- Duplicate IP Address Shuts Interface Down **C-27**
- The SensorApp and Alerting **C-28**
 - The SensorApp is Not Running **C-28**
 - Physical Connectivity, SPAN, or VACL Port Issue **C-30**
 - Unable to See Alerts **C-31**
 - Sensor Not Seeing Packets **C-33**
 - Cleaning Up a Corrupted SensorApp Configuration **C-34**
- Blocking **C-35**
 - Troubleshooting Blocking **C-35**
 - Verifying the ARC is Running **C-36**
 - Verifying ARC Connections are Active **C-37**
 - Device Access Issues **C-39**
 - Verifying the Interfaces and Directions on the Network Device **C-40**
 - Enabling SSH Connections to the Network Device **C-41**
 - Blocking Not Occurring for a Signature **C-41**
 - Verifying the Master Blocking Sensor Configuration **C-42**
- Logging **C-44**
 - Enabling Debug Logging **C-44**
 - Zone Names **C-48**
 - Directing cidLog Messages to SysLog **C-48**
- TCP Reset Not Occurring for a Signature **C-49**
- Software Upgrades **C-51**
 - Which Updates to Apply and Their Prerequisites **C-51**
 - Issues With Automatic Update **C-51**
 - Updating a Sensor with the Update Stored on the Sensor **C-52**
- Troubleshooting the IDM **C-53**
 - Cannot Launch the IDM - Loading Java Applet Failed **C-53**
 - Cannot Launch the IDM-The Analysis Engine Busy **C-54**
 - The IDM, Remote Manager, or Sensing Interfaces Cannot Access Sensor **C-54**
 - Signatures Not Producing Alerts **C-55**
- Troubleshooting the IME **C-55**
 - Time Synchronization on IME and the Sensor **C-56**
 - Not Supported Error Message **C-56**
- Troubleshooting the ASA 5500-X IPS SSP **C-56**
 - Health and Status Information **C-57**
 - Failover Scenarios **C-64**
 - The ASA 5500-X IPS SSP and the Normalizer Engine **C-65**
 - The ASA 5500-X IPS SSP and Memory Usage **C-66**
 - The ASA 5500-X IPS SSP and Jumbo Packets **C-66**

IPS Reloading Messages	C-67
Troubleshooting the ASA 5585-X IPS SSP	C-67
Health and Status Information	C-67
Failover Scenarios	C-70
Traffic Flow Stopped on IPS Switchports	C-72
The ASA 5585-X IPS SSP and the Normalizer Engine	C-72
The ASA 5585-X IPS SSP and Jumbo Packets	C-73
IPS Reloading Messages	C-73
Gathering Information	C-73
Health and Network Security Information	C-73
Tech Support Information	C-74
Understanding the show tech-support Command	C-74
Displaying Tech Support Information	C-75
Tech Support Command Output	C-76
Version Information	C-79
Understanding the show version Command	C-79
Displaying Version Information	C-79
Statistics Information	C-82
Understanding the show statistics Command	C-82
Displaying Statistics	C-83
Interfaces Information	C-94
Understanding the show interfaces Command	C-94
Interfaces Command Output	C-95
Displaying Interface Traffic History	C-96
Events Information	C-98
Sensor Events	C-98
Understanding the show events Command	C-98
Displaying Events	C-99
Clearing Events	C-102
cidDump Script	C-102
Uploading and Accessing Files on the Cisco FTP Site	C-103

APPENDIX D**CLI Error Messages** **D-1**

CLI Error Messages	D-1
CLI Validation Error Messages	D-6

GLOSSARY**INDEX**



Preface

Published: January 15, 2014, OL-30788-01

Revised: April 23, 2014

Contents

This document describes how to configure the sensor using the Cisco IPS 7.3 CLI. It contains the following sections:

- [Audience, page xxiii](#)
- [Organization, page xxiii](#)
- [Conventions, page xxv](#)
- [Related Documentation, page xxvi](#)
- [Obtaining Documentation, Using Cisco Bug Search Tool, and Submitting a Service Request, page xxvi](#)

Audience

This guide is intended for administrators who need to do the following:

- Configure the sensor for intrusion prevention using the CLI.
- Secure their network with IPS sensors.
- Prevent intrusion on their networks and monitor subsequent alerts.

Organization

This guide includes the following sections:

Section	Title	Description
1	“Introducing the CLI Configuration Guide”	Describes the purpose of the CLI Configuration Guide.
2	“Logging In to the Sensor”	Describes how to log in to the various sensors.

Section	Title	Description
3	“Initializing the Sensor”	Describes how to use the setup command to initialize sensors.
4	“Setting Up the Sensor”	Describes how to use the CLI to configure initial settings on the sensor.
5	“Configuring Interfaces”	Describes how to configure promiscuous, inline, inline VLAN pair, and VLAN group interfaces.
6	“Configuring Virtual Sensors”	Describes how to configure virtual sensors.
7	“Configuring Event Action Rules”	Describes how to configure event action rules policies on the sensor.
8	“Defining Signatures”	Describes how to add, clone, and edit signatures.
9	“Configuring Anomaly Detection”	Describes how to configure anomaly detection policies on the sensor.
10	“Configuring Global Correlation”	Describes how to configure global correlation features on the sensor.
11	“Configuring External Product Interfaces”	Describes how to configure external product interfaces for CSA MC.
12	“Configuring IP Logging”	Describes how to configure IP logging on the sensor.
13	“Displaying and Capturing Live Traffic on an Interface”	Describes how to display and capture live traffic on sensor interfaces.
14	“Configuring Attack Response Controller for Blocking and Rate Limiting”	Describes how to configure blocking and rate limiting on Cisco routers, and switches, and how to configure a master blocking sensor.
15	“Configuring SNMP”	Describes how to configure SNMP on the sensor.
16	“Working With Configuration Files”	Describes how to use configuration files on the sensor.
17	“Administrative Tasks for the Sensor”	Describes various administrative procedures to help you keep your sensor working and up to date.
18	“Configuring the ASA 5500-X IPS SSP”	Describes how to configure the ASA 5500-X IPS SSP.
19	“Configuring the ASA 5585-X IPS SSP”	Describes how to configure the ASA 5585-X IPS SSP.
20	“Obtaining Software”	Describes where to go to get the latest IPS software and describes the naming conventions.
21	“Upgrading, Downgrading, and Installing System Images”	Describes how to upgrade sensors and reimage the various sensors.
A	“System Architecture”	Describes the IPS system architecture.
B	“Signature Engines”	Describes the IPS signature engines and their parameters.
C	“Troubleshooting”	Contains troubleshooting tips for IPS hardware and software.

Section	Title	Description
D	“CLI Error Messages”	Lists the CLI error messages.
	“Glossary”	Contains IPS acronyms and terms.

Conventions

This document uses the following conventions:

Convention	Indication
bold font	Commands and keywords and user-entered text appear in bold font .
<i>italic font</i>	Document titles, new or emphasized terms, and arguments for which you supply values are in <i>italic font</i> .
[]	Elements in square brackets are optional.
{ x y z }	Required alternative keywords are grouped in braces and separated by vertical bars.
[x y z]	Optional alternative keywords are grouped in brackets and separated by vertical bars.
string	A nonquoted set of characters. Do not use quotation marks around the string or the string will include the quotation marks.
<code>courier font</code>	Terminal sessions and information the system displays appear in <code>courier font</code> .
< >	Nonprinting characters such as passwords are in angle brackets.
[]	Default responses to system prompts are in square brackets.
!, #	An exclamation point (!) or a pound sign (#) at the beginning of a line of code indicates a comment line.



Note

Means *reader take note*.



Tip

Means *the following information will help you solve a problem*.



Caution

Means *reader be careful*. In this situation, you might perform an action that could result in equipment damage or loss of data.



Timesaver

Means *the described action saves time*. You can save time by performing the action described in the paragraph.



Warning

Means *reader be warned*. In this situation, you might perform an action that could result in bodily injury.

Related Documentation

For information on Cisco IPS 7.3 documentation and how to find it, refer to the documentation roadmap found at this URL:

http://www.cisco.com/c/en/us/td/docs/security/ips/7-3/roadmap/roadmap7_3.html

For all IPS documentation, refer to this URL:

http://www.cisco.com/en/US/products/hw/vpndevc/ps4077/tsd_products_support_series_home.html

For a complete list of the Cisco ASA 5500 series documentation and where to find it, refer to the documentation roadmap found at this URL:

<http://www.cisco.com/en/US/docs/security/asa/roadmap/asaroadmap.html>

Obtaining Documentation, Using Cisco Bug Search Tool, and Submitting a Service Request

For information on obtaining documentation, using the Cisco Bug Search Tool (BST), submitting a service request, and gathering additional information, see *What's New in Cisco Product Documentation* at: <http://www.cisco.com/en/US/docs/general/whatsnew/whatsnew.html>.

Subscribe to *What's New in Cisco Product Documentation*, which lists all new and revised Cisco technical documentation, as an RSS feed and deliver content directly to your desktop using a reader application. The RSS feeds are a free service.



Introducing the CLI Configuration Guide

This chapter introduces the IPS CLI configuration guide, and contains the following sections:

- [Supported IPS Platforms, page 1-1](#)
- [IPS CLI Configuration Guide, page 1-1](#)
- [Sensor Configuration Sequence, page 1-2](#)
- [User Roles, page 1-3](#)
- [CLI Behavior, page 1-5](#)
- [Command Line Editing, page 1-6](#)
- [IPS Command Modes, page 1-7](#)
- [Regular Expression Syntax, page 1-8](#)
- [Generic CLI Commands, page 1-10](#)
- [CLI Keywords, page 1-10](#)

Supported IPS Platforms

For a list of IPS platforms supported by your IPS release, refer to the Release Notes for your version found at this URL:

<http://www.cisco.com/c/en/us/support/security/ips-4200-series-sensors/products-release-notes-list.html>

IPS CLI Configuration Guide

This guide is a task-based configuration guide for the Cisco IPS 7.3 CLI. The term “sensor” is used throughout this guide to refer to all sensor models, unless a procedure refers to a specific appliance or to one of the modules, then the specific model name is used.

For an alphabetical list of all IPS commands, refer to the *Command Reference for Cisco Intrusion Prevention System 7.3*. For information on locating all IPS 7.1 documents on Cisco.com, refer to the *Documentation Roadmap for Cisco Intrusion Prevention System 7.3*.

You can also use an IPS manager to configure your sensor. For information on how to access documentation that describes how to use IPS managers, refer to the *Documentation Roadmap for Cisco Intrusion Prevention System 7.3*.

Sensor Configuration Sequence

Perform the following tasks to configure the sensor:

1. Log in to the sensor.
2. Initialize the sensor by running the **setup** command.
3. Verify the sensor initialization.
4. Create the service account. A service account is needed for special debug situations directed by TAC.



Note Only one user with the role of service is allowed.

5. License the sensor.
6. Perform the other initial tasks, such as adding users and trusted hosts, and so forth.
7. Make changes to the interface configuration if necessary. You configure the interfaces during initialization.
8. Add or delete virtual sensors as necessary. You configure the virtual sensors during initialization.
9. Configure event action rules.
10. Configure the signatures for intrusion prevention.
11. Configure the sensor for global correlation.
12. Configure anomaly detection if needed. You can run anomaly detection using the default values or you can tailor it to suit your network needs.



Note Anomaly detection is disabled by default. You must enable it to configure or apply an anomaly detection policy. Enabling anomaly detection results in a decrease in performance.

13. Set up any external product interfaces if needed. The CSA MC is the only external product supported by the Cisco IPS.
14. Configure IP logging if needed.
15. Configure blocking if needed.
16. Configure SNMP if needed.
17. Perform miscellaneous tasks to keep your sensor running smoothly.
18. Upgrade the IPS software with new signature updates and service packs.
19. Reimage the application partition when needed.

For More Information

- For the procedure for logging in to your sensor, see [Chapter 2, “Logging In to the Sensor.”](#)
- For the procedure for using the **setup** command to initialize your sensor, see [Chapter 3, “Initializing the Sensor.”](#)
- For the procedure for verifying sensor initialization, see [Verifying Initialization, page 3-16.](#)
- For the procedure for obtaining and installing the license key, see [Installing the License Key, page 4-58.](#)

- For the procedures for setting up your sensor, see [Chapter 4, “Setting Up the Sensor.”](#)
- For the procedure for creating the service account, see [Creating the Service Account, page 4-28.](#)
- For the procedures for configuring interfaces on your sensor, see [Chapter 5, “Configuring Interfaces.”](#)
- For the procedures for configuring virtual sensors on your sensor, see [Chapter 6, “Configuring Virtual Sensors.”](#)
- For the procedures for configuring event action rules policies, see [Chapter 8, “Configuring Event Action Rules.”](#)
- For the procedures for configuring signatures for intrusion prevention, see [Chapter 7, “Defining Signatures.”](#)
- For the procedures for configuring global correlation, see [Chapter 10, “Configuring Global Correlation.”](#)
- For the procedure for configuring anomaly detection policies, see [Chapter 9, “Configuring Anomaly Detection.”](#)
- For the procedure for setting up external product interfaces, see [Chapter 11, “Configuring External Product Interfaces.”](#)
- For the procedures for configuring IP logging, see [Chapter 12, “Configuring IP Logging.”](#)
- For the procedures for configuring blocking on your sensor, see [Chapter 14, “Configuring Attack Response Controller for Blocking and Rate Limiting.”](#)
- For the procedures for configuring SNMP on your sensor, see [Chapter 15, “Configuring SNMP.”](#)
- For the administrative procedures, see [Chapter 17, “Administrative Tasks for the Sensor.”](#)
- For more information on how to obtain Cisco IPS software, see [Chapter 20, “Obtaining Software.”](#)
- For the procedures for installing system images, see [Chapter 21, “Upgrading, Downgrading, and Installing System Images.”](#)

User Roles

The Cisco CLI permits multiple users to log in at the same time. You can create and remove users from the local sensor. You can modify only one user account at a time. Each user is associated with a role that controls what that user can and cannot modify. The CLI supports four user roles: administrator, operator, viewer, and service. The privilege levels for each role are different; therefore, the menus and available commands vary for each role.

Administrator

This user role has the highest level of privileges. Administrators have unrestricted view access and can perform the following functions:

- Add users and assign passwords
- Enable and disable control of physical interfaces and virtual sensors
- Assign physical sensing interfaces to a virtual sensor
- Modify the list of hosts allowed to connect to the sensor as a configuring or viewing agent
- Modify sensor address configuration
- Tune signatures

- Assign configuration to a virtual sensor
- Manage routers

Operators

This user role has the second highest level of privileges. Operators have unrestricted view access and can perform the following functions:

- Modify their passwords
- Tune signatures
- Manage routers
- Assign configuration to a virtual sensor

Viewers

This user role has the lowest level of privileges. Viewers can view configuration and event data and can modify their passwords.



Tip

Monitoring applications only require viewer access to the sensor. You can use the CLI to set up a user account with viewer privileges and then configure the event viewer to use this account to connect to the sensor.

Service

This user role does not have direct access to the CLI. Service account users are logged directly into a bash shell. Use this account for support and troubleshooting purposes only. Unauthorized modifications are not supported and require the device to be reimaged to guarantee proper operation. You can create only one user with the service role. In the service account you can also switch to user root by executing `su-`. The root password is synchronized to the service account password. Some troubleshooting procedures may require you to execute commands as the root user.

When you log in to the service account, you receive the following warning:

```
***** WARNING *****
UNAUTHORIZED ACCESS TO THIS NETWORK DEVICE IS PROHIBITED.
This account is intended to be used for support and troubleshooting purposes only.
Unauthorized modifications are not supported and will require this device to be re-imaged
to guarantee proper operation.
*****
```



Note

The service role is a special role that allows you to bypass the CLI if needed. Only a user with administrator privileges can edit the service account.



Note

For IPS 5.0 and later, you can no longer remove the `cisco` account. You can disable it using the `no password cisco` command, but you cannot remove it. To use the `no password cisco` command, there must be another administrator account on the sensor. Removing the `cisco` account through the service account is not supported. If you remove the `cisco` account through the service account, the sensor most likely will not boot up, so to recover the sensor you must reinstall the sensor system image.

CLI Behavior

The following tips help you use the Cisco IPS CLI.

Prompts

- You cannot change the prompt displayed for the CLI commands.
- User interactive prompts occur when the system displays a question and waits for user input. The default input is displayed inside brackets []. To accept the default input, press **Enter**.

Help

- To display the help for a command, type **?** after the command.

The following example demonstrates the **?** function:

```
sensor# configure ?  
terminal      Configure from the terminal  
sensor# configure
```



Note When the prompt returns from displaying help, the command previously entered is displayed without the **?**.

- You can type **?** after an incomplete token to view the valid tokens that complete the command. If there is a trailing space between the token and the **?**, you receive an ambiguous command error:

```
sensor# show c ?  
% Ambiguous command: "show c"
```

If you enter the token without the space, a selection of available tokens for the completion (with no help description) appears:

```
sensor# show c?  
clock configuration  
sensor# show c
```

- Only commands available in the current mode are displayed by help.

Tab Completion

- Only commands available in the current mode are displayed by tab complete and help.
- If you are unsure of the complete syntax for a command, you can type a portion of the command and press **Tab** to complete the command.
- If multiple commands match for tab completion, nothing is displayed.

Recall

- To recall the commands entered in a mode, use the Up Arrow or Down Arrow keys or press **Ctrl-P** or **Ctrl-N**.



Note Help and tab complete requests are not reported in the recall list.

- A blank prompt indicates the end of the recall list.

Case Sensitivity

- The CLI is not case sensitive, but it does echo back the text in the same case you typed it. For example, if you type:

```
sensor# CONF
```

and press **Tab**, the sensor displays:

```
sensor# CONFigure
```



Note CLI commands are not case sensitive, but values are case sensitive. Remember this when you are creating regular expressions in signatures. A regular expression of “STRING” will not match “string” seen in a packet.

Display Options

- `--More--` is an interactive prompt that indicates that the terminal output exceeds the allotted display space. To display the remaining output, press the **spacebar** to display the next page of output or press **Enter** to display the output one line at a time.
- To clear the current line contents and return to a blank command line, press **Ctrl-C**.

For More Information

For more information on CLI command regular expression syntax, see [Regular Expression Syntax, page 1-8](#).

Command Line Editing

[Table 1-1](#) describes the command line editing capabilities provided by the Cisco IPS CLI.

Table 1-1 *Command Line Editing*

Keys	Description
Tab	Completes a partial command name entry. When you type a unique set of characters and press Tab, the system completes the command name. If you type a set of characters that could indicate more than one command, the system beeps to indicate an error. Type a question mark (?) immediately following the partial command (no space). The system provides a list of commands that begin with that string.
Backspace	Erases the character to the left of the cursor.
Enter	At the command line, pressing Enter processes a command. At the <code>--More--</code> prompt on a terminal screen, pressing Enter scrolls down a line.
Spacebar	Enables you to see more output on the terminal screen. Press the Spacebar when you see the line <code>--More--</code> on the screen to display the next screen.
Left arrow	Moves the cursor one character to the left. When you type a command that extends beyond a single line, you can press the Left Arrow key repeatedly to scroll back toward the system prompt and verify the beginning of the command entry.
Right arrow	Moves the cursor one character to the right.
Up Arrow or Ctrl-P	Recalls commands in the history buffer, beginning with the most recent command. Repeat the key sequence to recall successively older commands.

Table 1-1 *Command Line Editing (continued)*

Keys	Description
Down Arrow or Ctrl-N	Returns to more recent commands in the history buffer after recalling commands with the Up Arrow or Ctrl-P. Repeat the key sequence to recall successively more recent commands.
Ctrl-A	Moves the cursor to the beginning of the line.
Ctrl-B	Moves the cursor back one character.
Ctrl-D	Deletes the character at the cursor.
Ctrl-E	Moves the cursor to the end of the command line.
Ctrl-F	Moves the cursor forward one character.
Ctrl-K	Deletes all characters from the cursor to the end of the command line.
Ctrl-L	Clears the screen and redisplay the system prompt and command line
Ctrl-T	Transposes the character to the left of the cursor with the character located at the cursor.
Ctrl-U	Deletes all characters from the cursor to the beginning of the command line.
Ctrl-V	Inserts a code to indicate to the system that the keystroke immediately following should be treated as a command entry, <i>not</i> as an editing key.
Ctrl-W	Deletes the word to the left of the cursor.
Ctrl-Y	Recalls the most recent entry in the delete buffer. The delete buffer contains the last ten items you deleted or cut.
Ctrl-Z	Ends configuration mode and returns you to the EXEC prompt.
Esc-B	Moves the cursor back one word.
Esc-C	Capitalizes the word at the cursor.
Esc-D	Deletes from the cursor to the end of the word.
Esc-F	Moves the cursor forward one word.
Esc-L	Changes the word at the cursor to lowercase.
Esc-U	Capitalizes from the cursor to the end of the word.

IPS Command Modes

The Cisco IPS CLI has the following command modes:

- privileged EXEC—Entered when you log in to the CLI interface.
- global configuration—Entered from privileged EXEC mode by entering **configure terminal**. The command prompt is `sensor(config)#`.
- service mode configuration—Entered from global configuration mode by entering **service service-name**. The command prompt is `sensor(config-ser)#`, where `ser` is the first three characters of the service name.
- multi-instance service mode—Entered from global configuration mode by entering **service service-name log-instance-name**. The command prompt is `sensor(config-log)#` where `log` is the first three characters of the log instance name. The only multi-instance services in the system are anomaly detection, signature definition, and event action rules.

Regular Expression Syntax


Note

The syntax in this section applies only to regular expressions used as part of a CLI command. It does not apply to regular expressions used by signatures.

Regular expressions are text patterns that are used for string matching. Regular expressions contain a mix of plain text and special characters to indicate what kind of matching to do. For example, if you are looking for a numeric digit, the regular expression to search for is “[0-9]”. The brackets indicate that the character being compared should match any one of the characters enclosed within the bracket. The dash (-) between 0 and 9 indicates that it is a range from 0 to 9. Therefore, this regular expression will match any character from 0 to 9, that is, any digit.

To search for a specific special character, you must use a backslash before the special character. For example, the single character regular expression “*” matches a single asterisk.

The regular expressions defined in this section are similar to a subset of the POSIX Extended Regular Expression definitions. In particular, “[.]”, “[==]”, and “[:]” expressions are not supported. Also, escaped expressions representing single characters are supported. A character can be represented as its hexadecimal value, for example, \x61 equals ‘a,’ so \x61 is an escaped expression representing the character ‘a.’

The regular expressions are case sensitive. To match “STRING” or “string” use the following regular expression: “[Ss][Tt][Rr][Ii][Nn][Gg].”

Table 1-2 lists the special characters.

Table 1-2 Regular Expression Syntax

Character	Description
^	Beginning of the string. The expression “^A” will match an “A” only at the beginning of the string.
^	Immediately following the left-bracket ([). Excludes the remaining characters within brackets from matching the target string. The expression “[^0-9]” indicates that the target character should not be a digit.
\$	Matches the end of the string. The expression “abc\$” matches the sub-string “abc” only if it is at the end of the string.
	Allows the expression on either side to match the target string. The expression “alb” matches “a” as well as “b.”
.	Matches any character.
*	Indicates that the character to the left of the asterisk in the expression should match 0 or more times.
+	Similar to * but there should be at least one match of the character to the left of the + sign in the expression.
?	Matches the character to its left 0 or 1 times.
()	Affects the order of pattern evaluation and also serves as a tagged expression that can be used when replacing the matched sub-string with another expression.

Table 1-2 Regular Expression Syntax (continued)

Character	Description
[]	Enclosing a set of characters indicates that any of the enclosed characters may match the target character.
\	Allows specifying a character that would otherwise be interpreted as special. \xHH represents the character whose value is the same as the value represented by (HH) hexadecimal digits [0-9A-Fa-f]. The value must be non-zero. BEL is the same as \x07, BS is \x08, FF is \x0C, LF is \x0A, CR is \x0D, TAB is \x09, and VT is \x0B. For any other character 'c', '\c' is the same as 'c' except that it is never interpreted as special

The following examples demonstrate the special characters:

- **a*** matches any number of occurrences of the letter a, including none.
- **a+** requires that at least one letter a be in the string to be matched.
- **ba?b** matches the string bb or bab.
- ****** matches any number of asterisks (*).

To use multipliers with multiple-character patterns, you enclose the pattern in parentheses.

- **(ab)*** matches any number of the multiple-character string ab.
- **([A-Za-z][0-9])+** matches one or more instances of alphanumeric pairs, but not none (that is, an empty string is not a match).

The order for matches using multipliers (*, +, or ?) is to put the longest construct first. Nested constructs are matched from outside to inside. Concatenated constructs are matched beginning at the left side of the construct. Thus, the regular expression matches A9b3, but not 9Ab3 because the letters are specified before the numbers.

You can also use parentheses around a single- or multiple-character pattern to instruct the software to remember a pattern for use elsewhere in the regular expression.

To create a regular expression that recalls a previous pattern, you use parentheses to indicate memory of a specific pattern and a backslash (\) followed by a digit to reuse the remembered pattern. The digit specifies the occurrence of a parentheses in the regular expression pattern. If you have more than one remembered pattern in your regular expression, \1 indicates the first remembered pattern, and \2 indicates the second remembered pattern, and so on.

The following regular expression uses parentheses for recall:

- **a(.)bc(.)\1\2** matches an *a* followed by any character, followed by *bc* followed by any character, followed by the first *any* character again, followed by the second *any* character again.

For example, the regular expression can match aZbcTZT. The software remembers that the first character is Z and the second character is T and then uses Z and T again later in the regular expression.

Generic CLI Commands

The following CLI commands are generic to the Cisco IPS.

- **configure terminal**—Enters global configuration mode.

Global configuration commands apply to features that affect the system as a whole rather than just one protocol or interface.

```
sensor# configure terminal
sensor(config)#
```

- **service**—Takes you to the following configuration submodes: analysis-engine, anomaly-detection, authentication, event-action-rules, external-product-interfaces, global-correlation, health-monitor, host, interface, logger, network-access, notification, signature-definition, ssh-known-hosts, trusted-certificates, and web-server.



Note The anomaly-detection, event-action-rules, and signature-definition submodes are multiple instance services. One predefined instance is allowed for each. For anomaly-detection, the predefined instance name is ad0. For event-action-rules, the predefined instance name is rules0. For signature-definition, the predefined instance name is sig0. You can create additional instances.

```
sensor# configure terminal
sensor(config)# service event-action-rules rules0
sensor(config-rul)#
```

- **end**—Exits configuration mode or any configuration submodes. It takes you back to the top-level EXEC menu.

```
sensor# configure terminal
sensor(config)# end
sensor#
```

- **exit**—Exits any configuration mode or closes an active terminal session and terminates the EXEC mode. It takes you to the previous menu session.

```
sensor# configure terminal
sensor(config)# service event-action-rules rules0
sensor(config-rul)# exit
sensor(config)# exit
sensor#
```

CLI Keywords

In general, use the **no** form of a command to disable a feature or function. Use the command without the keyword **no** to enable a disabled feature or function. For example, the command **ssh host-key ip_address** adds an entry to the known hosts table, the command **no ssh host-key ip_address** removes the entry from the known hosts table. Refer to the individual commands for a complete description of what the **no** form of that command does.

Service configuration commands can also have a default form. Use the **default** form of the command to return the command setting to its default. This keyword applies to the **service** submenu commands used for application configuration. Entering **default** with the command resets the parameter to the default value. You can only use the **default** keyword with commands that specify a default value in the configuration files.



Logging In to the Sensor

This chapter explains how to log in to the sensor. It contains the following sections:

- [Logging In Notes and Caveats, page 2-1](#)
- [Supported User Roles, page 2-1](#)
- [Logging In to the Appliance, page 2-2](#)
- [Connecting an Appliance to a Terminal Server, page 2-3](#)
- [Logging In to the ASA 5500-X IPS SSP, page 2-4](#)
- [Logging In to the ASA 5585-X IPS SSP, page 2-5](#)
- [Logging In to the Sensor, page 2-6](#)

Logging In Notes and Caveats

The following notes and caveats apply to logging in to the sensor:

- All IPS platforms allow ten concurrent log in sessions.
- The service role is a special role that allows you to bypass the CLI if needed. Only a user with administrator privileges can edit the service account.
- You must initialize the appliance (run the **setup** command) from the console. After networking is configured, SSH and Telnet are available. You can log in to the appliance from a console port.
- You log in to the ASA 5500-X IPS SSP and ASA 5585-X IPS SSP from the adaptive security appliance.

Supported User Roles

You can log in with the following user privileges:

- Administrator
- Operator
- Viewer
- Service

The service role does not have direct access to the CLI. Service account users are logged directly into a bash shell. Use this account for support and troubleshooting purposes only. Unauthorized modifications are not supported and will require the sensor to be reimaged to guarantee proper operation. You can create only one user with the service role.

When you log in to the service account, you receive the following warning:

```
***** WARNING *****
UNAUTHORIZED ACCESS TO THIS NETWORK DEVICE IS PROHIBITED.
This account is intended to be used for support and troubleshooting purposes only.
Unauthorized modifications are not supported and will require this device to be re-imaged
to guarantee proper operation.
*****
```



Note

The service role is a special role that allows you to bypass the CLI if needed. Only a user with administrator privileges can edit the service account.

For More Information

- For the procedure for creating the service account, see [Creating the Service Account, page 4-28](#).
- For the procedures for adding and deleting users, see [Configuring Authentication and User Parameters, page 4-18](#).

Logging In to the Appliance



Note

You can log in to the appliance from a console port. The currently supported Cisco IPS appliances are the IPS 4345, IPS 4360, IPS 4510, IPS 4520, and IPS 4520-XL.

To log in to the appliance, follow these steps:

- Step 1** Connect a console port to the sensor to log in to the appliance.
- Step 2** Enter your username and password at the login prompt.



Note

The default username and password are both **cisco**. You are prompted to change them the first time you log in to the appliance. You must first enter the UNIX password, which is **cisco**. Then you must enter the new password twice.

```
login: cisco
Password:
***NOTICE***
```

This product contains cryptographic features and is subject to United States and local country laws governing import, export, transfer and use. Delivery of Cisco cryptographic products does not imply third-party authority to import, export, distribute or use encryption. Importers, exporters, distributors and users are responsible for compliance with U.S. and local country laws. By using this product you agree to comply with applicable laws and regulations. If you are unable to comply with U.S. and local laws, return this product immediately.

A summary of U.S. laws governing Cisco cryptographic products may be found at: <http://www.cisco.com/wvl/export/crypto/tool/stqrg.html>

If you require further assistance please contact us by sending email to export@cisco.com.


```
***LICENSE NOTICE***
```

```
There is no license key installed on the system.
```

```
Please go to http://www.cisco.com/go/license to obtain a new license or install a license.  
sensor#
```

For More Information

- For the procedure for connecting an appliance to a terminal server, see [Connecting an Appliance to a Terminal Server, page 2-3](#).
- For the procedure for using the **setup** command to initialize the appliance, see [Basic Sensor Setup, page 3-5](#).

Connecting an Appliance to a Terminal Server

A terminal server is a router with multiple, low speed, asynchronous ports that are connected to other serial devices. You can use terminal servers to remotely manage network equipment, including appliances.

To set up a Cisco terminal server with RJ-45 or hydra cable assembly connections, follow these steps:

-
- Step 1** Connect to a terminal server using one of the following methods:
- For terminal servers with RJ-45 connections, connect a rollover cable from the console port on the appliance to a port on the terminal server.
 - For hydra cable assemblies, connect a straight-through patch cable from the console port on the appliance to a port on the terminal server.
- Step 2** Configure the line and port on the terminal server. In enable mode, enter the following configuration, where # is the line number of the port to be configured.
- ```
config t
line #
login
transport input all
stopbits 1
flowcontrol hardware
speed 9600
exit
exit
wr mem
```
- Step 3** Be sure to properly close a terminal session to avoid unauthorized access to the appliance. If a terminal session is not stopped properly, that is, if it does not receive an exit(0) signal from the application that initiated the session, the terminal session can remain open. When terminal sessions are not stopped properly, authentication is not performed on the next session that is opened on the serial port.



#### Caution

Always exit your session and return to a login prompt before terminating the application used to establish the connection.

---

**Caution**

If a connection is dropped or terminated by accident, you should reestablish the connection and exit normally to prevent unauthorized access to the appliance.

## Logging In to the ASA 5500-X IPS SSP

You log in to the ASA 5500-X IPS SSP from the adaptive security appliance.

To session in to the ASA 5500-X IPS SSP from the adaptive security appliance, follow these steps:

**Step 1** Log in to the adaptive security appliance.

**Note**

If the adaptive security appliance is operating in multi-mode, use the **change system** command to get to the system level prompt before continuing.

**Step 2** Session to the IPS. You have 60 seconds to log in before the session times out.

```
asa# session ips
Opening command session with slot 1.
Connected to slot 1. Escape character sequence is 'CTRL-^X'.
```

**Step 3** Enter your username and password at the login prompt.

**Note**

The default username and password are both **cisco**. You are prompted to change them the first time you log in to the module. You must first enter the UNIX password, which is **cisco**. Then you must enter the new password twice.

```
login: cisco
Password:
NOTICE
This product contains cryptographic features and is subject to United States and local
country laws governing import, export, transfer and use. Delivery of Cisco cryptographic
products does not imply third-party authority to import, export, distribute or use
encryption. Importers, exporters, distributors and users are responsible for compliance
with U.S. and local country laws. By using this product you agree to comply with
applicable laws and regulations. If you are unable to comply with U.S. and local laws,
return this product immediately.
```

A summary of U.S. laws governing Cisco cryptographic products may be found at:  
<http://www.cisco.com/wwl/export/crypto/tool/stqrg.html>

If you require further assistance please contact us by sending email to [export@cisco.com](mailto:export@cisco.com).

```
LICENSE NOTICE
There is no license key installed on this IPS platform.
The system will continue to operate with the currently installed
signature set. A valid license must be obtained in order to apply
signature updates. Please go to http://www.cisco.com/go/license
to obtain a new license or install a license.
```

```
asa-ips#
```

- Step 4** To escape from a session and return to the adaptive security appliance prompt, do one of the following:
- Enter **exit**.
  - Press **CTRL-Shift-6-x** (represented as **CTRL^X**).

#### For More Information

For the procedure for using the **setup** command to initialize the ASA 5500-X IPS SSP, see [Advanced Setup for the ASA 5500-X IPS SSP, page 3-14](#).

## Logging In to the ASA 5585-X IPS SSP

You log in to the ASA 5585-X IPS SSP from the adaptive security appliance.

To session in to the ASA 5585-X IPS SSP from the adaptive security appliance, follow these steps:

- Step 1** Log in to the adaptive security appliance.



**Note** If the adaptive security appliance is operating in multi-mode, use the **change system** command to get to the system level prompt before continuing.

- Step 2** Session to the ASA 5585-X IPS SSP. You have 60 seconds to log in before the session times out.

```
asa# session 1
Opening command session with slot 1.
Connected to slot 1. Escape character sequence is 'CTRL-^X'.
```

- Step 3** Enter your username and password at the login prompt.



**Note** The default username and password are both **cisco**. You are prompted to change them the first time you log in to the module. You must first enter the UNIX password, which is **cisco**. Then you must enter the new password twice.

```
login: cisco
Password:
NOTICE
This product contains cryptographic features and is subject to United States and local
country laws governing import, export, transfer and use. Delivery of Cisco cryptographic
products does not imply third-party authority to import, export, distribute or use
encryption. Importers, exporters, distributors and users are responsible for compliance
with U.S. and local country laws. By using this product you agree to comply with
applicable laws and regulations. If you are unable to comply with U.S. and local laws,
return this product immediately.

A summary of U.S. laws governing Cisco cryptographic products may be found at:
http://www.cisco.com/wwl/export/crypto/tool/stqrg.html

If you require further assistance please contact us by sending email to export@cisco.com.

LICENSE NOTICE
There is no license key installed on the system.
Please go to http://www.cisco.com/go/license to obtain a new license or install a license.
ips-ssp#
```

- Step 4** To escape from a session and return to the adaptive security appliance prompt, do one of the following:
- Enter **exit**.
  - Press **CTRL-Shift-6-x** (represented as **CTRL^X**).
- 

#### For More Information

For the procedure for using the **setup** command to initialize the ASA 5585-X IPS SSP, see [Advanced Setup for the ASA 5585-X IPS SSP, page 3-15](#).

## Logging In to the Sensor



### Note

After you have initialized the sensor using the **setup** command and enabled Telnet, you can use SSH or Telnet to log in to the sensor.

---

To log in to the sensor using Telnet or SSH, follow these steps:

---

- Step 1** To log in to the sensor over the network using SSH or Telnet.

```
ssh sensor_ip_address
telnet sensor_ip_address
```

- Step 2** Enter your username and password at the login prompt.

```
login: *****
Password: *****
NOTICE
```

This product contains cryptographic features and is subject to United States and local country laws governing import, export, transfer and use. Delivery of Cisco cryptographic products does not imply third-party authority to import, export, distribute or use encryption. Importers, exporters, distributors and users are responsible for compliance with U.S. and local country laws. By using this product you agree to comply with applicable law s and regulations. If you are unable to comply with U.S. and local laws, return this product immediately.

A summary of U.S. laws governing Cisco cryptographic products may be found at:  
<http://www.cisco.com/wwl/export/crypto/tool/stqrg.html>

If you require further assistance please contact us by sending email to [export@cisco.com](mailto:export@cisco.com).

```
LICENSE NOTICE
```

There is no license key installed on the system.  
Please go to <http://www.cisco.com/go/license> to obtain a new license or install a license.  
sensor#

---



## Initializing the Sensor

---

This chapter describes how to use the **setup** command to initialize the sensor, and contains the following sections:

- [Initializing Notes and Caveats, page 3-1](#)
- [Understanding Initialization, page 3-1](#)
- [Participating in the SensorBase Network, page 3-2](#)
- [Simplified Setup Mode, page 3-3](#)
- [System Configuration Dialog, page 3-3](#)
- [Basic Sensor Setup, page 3-5](#)
- [Advanced Setup, page 3-8](#)
- [Verifying Initialization, page 3-16](#)

## Initializing Notes and Caveats

The following notes and caveats apply to initializing the sensor:

- You must be administrator to use the **setup** command.
- You must have a valid sensor license for automatic signature updates and global correlation features to function. You can still configure and display statistics for the global correlation features, but the global correlation databases are cleared and no updates are attempted. Once you install a valid license, the global correlation features are reactivated.
- The currently supported Cisco IPS appliances are the IPS 4345, IPS 4360, IPS 4510, IPS 4520, and IPS 4520-XL.
- You do not need to configure interfaces on the ASA IPS modules (ASA 5500-X IPS SSP and ASA 5585-X IPS SSP). You should ignore the modify interface default VLAN setting in setup. The separation of traffic across virtual sensors is configured differently for the ASA IPS modules than for other sensors.

## Understanding Initialization

After you install the sensor on your network, you must use the **setup** command to initialize it so that you can communicate with it over the network. You cannot use the IDM or the IME to configure the sensor until you initialize the sensor using the **setup** command.

With the **setup** command, you configure basic sensor settings, including the hostname, IP interfaces, access control lists, global correlation servers, and time settings. You can continue using advanced setup in the CLI to enable Telnet, enable SSHv1 fallback, configure the web server, and assign and enable virtual sensors and interfaces, or you can use the Startup Wizard in the IDM or the IME. After you configure the sensor with the **setup** command, you can change the network settings in the IDM or the IME.

**Note**

You must be administrator to use the **setup** command.

## Participating in the SensorBase Network

The Cisco IPS contains a security capability, Cisco Global Correlation, which uses the immense security intelligence that we have amassed over the years. At regular intervals, the Cisco IPS receives threat updates from the Cisco SensorBase Network, which contain detailed information about known threats on the Internet, including serial attackers, Botnet harvesters, Malware outbreaks, and dark nets. The IPS uses this information to filter out the worst attackers before they have a chance to attack critical assets. It then incorporates the global threat data in to its system to detect and prevent malicious activity even earlier.

If you agree to participate in the SensorBase Network, Cisco will collect aggregated statistics about traffic sent to your IPS. This includes summary data on the Cisco IPS network traffic properties and how this traffic was handled by the Cisco appliances. We do not collect the data content of traffic or other confidential business or personal information. All data is aggregated and sent by secure HTTP to the Cisco SensorBase Network servers in periodic intervals. All data shared with Cisco will be anonymous and treated as strictly confidential.

Table 3-1 shows how we use the data.

**Table 3-1 Cisco Network Participation Data Use**

| Participation Level | Type of Data                                                                                 | Purpose                                                              |
|---------------------|----------------------------------------------------------------------------------------------|----------------------------------------------------------------------|
| Partial             | Protocol attributes (TCP maximum segment size and options string, for example)               | Tracks potential threats and helps us to understand threat exposure. |
|                     | Attack type (signature fired and risk rating, for example)                                   | Used to understand current attacks and attack severity.              |
|                     | Connecting IP address and port                                                               | Identifies attack source.                                            |
|                     | Summary IPS performance (CPU utilization, memory usage, inline vs. promiscuous, for example) | Tracks product efficacy.                                             |
| Full                | Victim IP address and port                                                                   | Detects threat behavioral patterns.                                  |

When you enable Partial or Full Network Participation, the Network Participation Disclaimer appears. You must enter **yes** to participate. If you do not have a license installed, you receive a warning telling you that global correlation inspection and reputation filtering are disabled until the sensor is licensed. You can obtain a license at <http://www.cisco.com/go/license>.

**For More Information**

- For more information about global correlation, see [Chapter 10, “Configuring Global Correlation.”](#)
- For the procedure for obtaining a sensor license, see [Installing the License Key, page 4-58.](#)

## Simplified Setup Mode

The sensor automatically calls the **setup** command when you connect to the sensor using a console cable and the sensor basic network settings have not yet been configured. The sensor does not call automatic setup under the following conditions:

- When initialization has already been successfully completed.
- If you have recovered or downgraded the sensor.
- If you have set the host configuration to default after successfully configuring the sensor using automatic setup.

When you enter the **setup** command, an interactive dialog called the System Configuration Dialog appears on the system console screen. The System Configuration Dialog guides you through the configuration process. The values shown in brackets next to each prompt are the default values last set.

## System Configuration Dialog

When you enter the **setup** command, an interactive dialog called the System Configuration Dialog appears on the system console screen. The System Configuration Dialog guides you through the configuration process. The values shown in brackets next to each prompt are the current values.

You must go through the entire System Configuration Dialog until you come to the option that you want to change. To accept default settings for items that you do not want to change, press **Enter**.

To return to the EXEC prompt without making changes and without going through the entire System Configuration Dialog, press **Ctrl-C**. The System Configuration Dialog also provides help text for each prompt. To access the help text, enter **?** at a prompt.

When you complete your changes, the System Configuration Dialog shows you the configuration that you created during the setup session. It also asks you if you want to use this configuration. If you enter **yes**, the configuration is saved. If you enter **no**, the configuration is not saved and the process begins again. There is no default for this prompt; you must enter either **yes** or **no**.

You can configure daylight savings time either in recurring mode or date mode. If you choose recurring mode, the start and end days are based on week, day, month, and time. If you choose date mode, the start and end days are based on month, day, year, and time. Choosing disable turns off daylight savings time.

**Note**

---

You only need to set the date and time in the System Configuration Dialog if the system is an appliance and is NOT using NTP.

---

**Note**

---

The System Configuration Dialog is an interactive dialog. The default settings are displayed.

---

Example 3-1 shows a sample System Configuration Dialog.

**Example 3-1 Example System Configuration Dialog**

```

--- Basic Setup ---

--- System Configuration Dialog ---

At any point you may enter a question mark '?' for help.
User ctrl-c to abort configuration dialog at any prompt.
Default settings are in square brackets '['].

Current time: Wed Mar 6 00:07:23 2013

Setup Configuration last modified:

Enter host name[sensor]:
Enter IP interface[192.168.1.2/24,192.168.1.1]:
Modify current access list?[no]:
Current access list entries:
 [1] 0.0.0.0/0
Delete:
Permit:
Use DNS server for Auto-Updates from www.cisco.com and Global Correlation?[no]:
 DNS server IP address[171.68.226.120]:
Use HTTP proxy server for Auto-Updates from www.cisco.com and Global Correlation?[no]:
 HTTP proxy server IP address:
 HTTP proxy server Port number:
Modify system clock settings?[no]:
 Modify summer time settings?[no]:
 Use USA SummerTime Defaults?[yes]:
 Recurring, Date or Disable?[Recurring]:
 Start Month[march]:
 Start Week[second]:
 Start Day[sunday]:
 Start Time[02:00:00]:
 End Month[november]:
 End Week[first]:
 End Day[sunday]:
 End Time[02:00:00]:
 DST Zone[]:
 Offset[60]:
 Modify system timezone?[no]:
 Timezone[UTC]:
 UTC Offset[0]:
 Use NTP?[no]:
 NTP Server IP Address[]:
 Use NTP Authentication?[no]:
 NTP Key ID[]:
 NTP Key Value[]:
 Modify system date and time?[no]:
 Local Date as YYYY-MM-DD[2013-03-06]:
 Local Time as HH:MM:SS[]:
 Participation in the SensorBase Network allows Cisco to collect aggregated statistics
 about traffic sent to your IPS.
 SensorBase Network Participation level?[off]:

If you agree to participate in the SensorBase Network, Cisco will collect aggregated
statistics about traffic sent to your IPS.

```



This includes summary data on the Cisco IPS network traffic properties and how this traffic was handled by the Cisco appliances. We do not collect the data content of traffic or other sensitive business or personal information. All data is aggregated and sent via secure HTTP to the Cisco SensorBase Network servers in periodic intervals. All data shared with Cisco will be anonymous and treated as strictly confidential.

The table below describes how the data will be used by Cisco.

Participation Level = "Partial":

- \* Type of Data: Protocol Attributes (e.g. TCP max segment size and options string)  
Purpose: Track potential threats and understand threat exposure
- \* Type of Data: Attack Type (e.g. Signature Fired and Risk Rating)  
Purpose: Used to understand current attacks and attack severity
- \* Type of Data: Connecting IP Address and port  
Purpose: Identifies attack source
- \* Type of Data: Summary IPS performance (CPU utilization memory usage, inline vs. promiscuous, etc)  
Purpose: Tracks product efficacy

Participation Level = "Full" additionally includes:

- \* Type of Data: Victim IP Address and port  
Purpose: Detect threat behavioral patterns

Do you agree to participate in the SensorBase Network?[no]:

## Basic Sensor Setup

You can perform basic sensor setup using the **setup** command, and then finish setting up the sensor using the CLI, IDM, or IME.

To perform basic sensor setup using the **setup** command, follow these steps:

---

**Step 1** Log in to the sensor using an account with administrator privileges.




---

**Note** Both the default username and password are **cisco**.

---

**Step 2** The first time you log in to the sensor you are prompted to change the default password. Passwords must be at least eight characters long and be strong, that is, not be a dictionary word. After you change the password, basic setup begins.

**Step 3** Enter the **setup** command. The System Configuration Dialog is displayed.

**Step 4** Specify the hostname. The hostname is a case-sensitive character string up to 64 characters. Numbers, “\_” and “-” are valid, but spaces are not acceptable. The default is sensor.

**Step 5** Specify the IP interface. The IP interface is in the form of IP Address/Netmask,Gateway: *X.X.X.X/nm,Y.Y.Y.Y*, where *X.X.X.X* specifies the sensor IP address as a 32-bit address written as 4 octets separated by periods, *nm* specifies the number of bits in the netmask, and *Y.Y.Y.Y* specifies the default gateway as a 32-bit address written as 4 octets separated by periods.

**Step 6** Enter **yes** to modify the network access list:

- a. If you want to delete an entry, enter the number of the entry and press **Enter**, or press **Enter** to get to the Permit line.
- b. Enter the IP address and netmask of the network you want to add to the access list.




---

**Note** For example, 10.0.0.0/8 permits all IP addresses on the 10.0.0.0 network (10.0.0.0-10.255.255.255) and 10.1.1.0/24 permits only the IP addresses on the 10.1.1.0 subnet (10.1.1.0-10.1.1.255). If you want to permit access to a single IP address than the entire network, use a 32-bit netmask. For example, 10.1.1.1/32 permits just the 10.1.1.1 address.

---

- c. Repeat Step b until you have added all networks that you want to add to the access list, and then press **Enter** at a blank permit line to go to the next step.

**Step 7** You must configure a DNS server or an HTTP proxy server for automatic updates from www.cisco.com and global correlation to operate:

- a. Enter **yes** to add a DNS server, and then enter the DNS server IP address.
- b. Enter **yes** to add an HTTP proxy server, and then enter the HTTP proxy server IP address and port number.

**Caution**


---

You must have a valid sensor license for automatic signature updates and global correlation features to function. You can still configure and display statistics for the global correlation features, but the global correlation databases are cleared and no updates are attempted. Once you install a valid license, the global correlation features are reactivated.

---

**Step 8** Enter **yes** to modify the system clock settings:

- a. Enter **yes** to modify summertime settings.




---

**Note** Summertime is also known as DST. If your location does not use Summertime, go to Step m.

---

- b. Enter **yes** to choose the USA summertime defaults, or enter **no** and choose recurring, date, or disable to specify how you want to configure summertime settings. The default is recurring.
- c. If you chose recurring, specify the month you want to start summertime settings. Valid entries are january, february, march, april, may, june, july, august, september, october, november, and december. The default is march.
- d. Specify the week you want to start summertime settings. Valid entries are first, second, third, fourth, fifth, and last. The default is second.
- e. Specify the day you want to start summertime settings. Valid entries are sunday, monday, tuesday, wednesday, thursday, friday, and saturday. The default is sunday.
- f. Specify the time you want to start summertime settings. The default is 02:00:00.




---

**Note** The default recurring summertime parameters are correct for time zones in the United States. The default values specify a start time of 2:00 a.m. on the second Sunday in March, and a stop time of 2:00 a.m. on the first Sunday in November. The default summertime offset is 60 minutes.

---

- g. Specify the month you want summertime settings to end. Valid entries are january, february, march, april, may, june, july, august, september, october, november, and december. The default is november.

- h. Specify the week you want the summertime settings to end. Valid entries are first, second, third, fourth, fifth, and last. The default is first.
- i. Specify the day you want the summertime settings to end. Valid entries are sunday, monday, tuesday, wednesday, thursday, friday, and saturday. The default is sunday.
- j. Specify the time you want summertime settings to end. The default is 02:00:00.
- k. Specify the DST zone. The zone name is a character string up to 24 characters long in the pattern [A-Za-z0-9()+;,-/]+\$.
- l. Specify the summertime offset. Specify the summertime offset from UTC in minutes (negative numbers represent time zones west of the Prime Meridian). The default is 60.
- m. Enter **yes** to modify the system time zone.
- n. Specify the standard time zone name. The zone name is a character string up to 24 characters long.
- o. Specify the standard time zone offset. Specify the standard time zone offset from UTC in minutes (negative numbers represent time zones west of the Prime Meridian). The default is 0.
- p. Enter **yes** if you want to use NTP. To use authenticated NTP, you need the NTP server IP address, the NTP key ID, and the NTP key value. If you do not have those at this time, you can configure NTP later. Otherwise, you can choose unauthenticated NTP.

**Step 9** Enter **off**, **partial**, or **full** to participate in the SensorBase Network Participation:

- Off—No data is contributed to the SensorBase Network.
- Partial—Data is contributed to the SensorBase Network, but data considered potentially sensitive is filtered out and never sent.
- Full—All data is contributed to the SensorBase Network except the attacker/victim IP addresses that you exclude.
- 

The SensorBase Network Participation disclaimer appears. It explains what is involved in participating in the SensorBase Network.

**Step 10** Enter **yes** to participate in the SensorBase Network.

```
The following configuration was entered.
service host
network-settings
host-ip 192.168.1.2/24, 192.168.1.1
host-name sensor
telnet-option disabled
sshv1-fallback enabled
access-list 10.0.0.0/8
ftp-timeout 300
no login-banner-text
dns-primary-server enabled
address 171.68.226.120
exit
dns-secondary-server disabled
dns-tertiary-server disabled
http-proxy proxy-server
address 128.107.241.170
port 8080
exit
exit
time-zone-settings
offset -360
standard-time-zone-name CST
exit
```

```

summertime-option recurring
offset 60
summertime-zone-name CDT
start-summertime
month march
week-of-month second
day-of-week sunday
time-of-day 02:00:00
exit
end-summertime
month november
week-of-month first
day-of-week sunday
time-of-day 02:00:00
exit
exit
ntp-option enabled
ntp-keys 1 md5-key 8675309
ntp-servers 10.10.1.2 key-id 1
exit
service global-correlation
network-participation full
exit

```

- [0] Go to the command prompt without saving this config.
- [1] Return to setup without saving this config.
- [2] Save this configuration and exit setup.
- [3] Continue to Advanced setup.

**Step 11** Enter **2** to save the configuration (or **3** to continue with advanced setup using the CLI).

```

Enter your selection[2]: 2
Configuration Saved.

```

**Step 12** If you changed the time setting, enter **yes** to reboot the sensor.

---

#### For More Information

For the procedure for obtaining the most recent IPS software, see [Obtaining Cisco IPS Software](#), page 20-1.

## Advanced Setup

This section describes how to continue with advanced setup in the CLI for the sensor. It contains the following sections:

- [Advanced Setup for the Appliance](#), page 3-9
- [Advanced Setup for the ASA 5500-X IPS SSP](#), page 3-14
- [Advanced Setup for the ASA 5585-X IPS SSP](#), page 3-15

## Advanced Setup for the Appliance



**Note** The currently supported Cisco IPS appliances are the IPS 4345, IPS 4360, IPS 4510, IPS 4520, and IPS 4520-XL



**Note** Adding new subinterfaces is a two-step process. You first organize the interfaces when you edit the virtual sensor configuration. You then choose which interfaces and subinterfaces are assigned to which virtual sensors.

The interfaces change according to the appliance model, but the prompts are the same for all models.

To continue with advanced setup for the appliance, follow these steps:

- Step 1** Log in to the appliance using an account with administrator privileges.
- Step 2** Enter the `setup` command. The System Configuration Dialog is displayed. Press **Enter** or the spacebar to skip to the menu to access advanced setup.
- Step 3** Enter `3` to access advanced setup.
- Step 4** Specify the Telnet server status. The default is disabled.
- Step 5** Specify the SSHv1 fallback setting. The default is enabled.
- Step 6** Specify the web server port. The web server port is the TCP port used by the web server (1 to 65535). The default is 443.



**Note** The web server is configured to use TLS/SSL encryption by default. Setting the port to 80 does not disable the encryption.

- Step 7** Enter **yes** to modify the interface and virtual sensor configuration and to see the current interface configuration.

```
Current interface configuration
Command control: Management0/0
Unassigned:
Promiscuous:
 GigabitEthernet0/0
 GigabitEthernet0/1
 GigabitEthernet0/2
 GigabitEthernet0/3

Virtual Sensor: vs0
Anomaly Detection: ad0
Event Action Rules: rules0
Signature Definitions: sig0

Virtual Sensor: vs1
Anomaly Detection: ad0
Event Action Rules: rules0
Signature Definitions: sig0

Virtual Sensor: vs2
Anomaly Detection: ad0
Event Action Rules: rules0
Signature Definitions: sig0
```

```
[1] Edit Interface Configuration
[2] Edit Virtual Sensor Configuration
[3] Display configuration
Option:
```

**Step 8** Enter **1** to edit the interface configuration.



**Note** The following options let you create and delete interfaces. You assign the interfaces to virtual sensors in the virtual sensor configuration. If you are using promiscuous mode for your interfaces and are not subdividing them by VLAN, no additional configuration is necessary.

```
[1] Remove interface configurations.
[2] Add/Modify Inline Vlan Pairs.
[3] Add/Modify Promiscuous Vlan Groups.
[4] Add/Modify Inline Interface Pairs.
[5] Add/Modify Inline Interface Pair Vlan Groups.
[6] Modify interface default-vlan.
Option:
```

**Step 9** Enter **2** to add inline VLAN pairs and display the list of available interfaces.



**Caution** The new VLAN pair is not automatically added to a virtual sensor.

```
Available Interfaces
[1] GigabitEthernet0/0
[2] GigabitEthernet0/1
[3] GigabitEthernet0/2
[4] GigabitEthernet0/3
Option:
```

**Step 10** Enter **1** to add an inline VLAN pair to GigabitEthernet 0/0, for example.

```
Inline Vlan Pairs for GigabitEthernet0/0
None
```

**Step 11** Enter a subinterface number and description.

```
Subinterface Number:
Description[Created via setup by user asmith]:
```

**Step 12** Enter numbers for VLAN 1 and 2.

```
Vlan1[]: 200
Vlan2[]: 300
```

**Step 13** Press **Enter** to return to the available interfaces menu.



**Note** Entering a carriage return at a prompt without a value returns you to the previous menu.

```
[1] GigabitEthernet0/0
[2] GigabitEthernet0/1
[3] GigabitEthernet0/2
[4] GigabitEthernet0/3
Option:
```



**Note** At this point, you can configure another interface, for example, GigabitEthernet 0/1, for inline VLAN pair.

**Step 14** Press **Enter** to return to the top-level interface editing menu.

```
[1] Remove interface configurations.
[2] Add/Modify Inline Vlan Pairs.
[3] Add/Modify Promiscuous Vlan Groups.
[4] Add/Modify Inline Interface Pairs.
[5] Add/Modify Inline Interface Pair Vlan Groups.
[6] Modify interface default-vlan.
Option:
```

**Step 15** Enter **4** to add an inline interface pair and see these options.

```
Available Interfaces
GigabitEthernet0/1
GigabitEthernet0/2
GigabitEthernet0/3
```

**Step 16** Enter the pair name, description, and which interfaces you want to pair.

```
Pair name: newPair
Description[Created via setup by user asmith:
Interface1[]: GigabitEthernet0/1
Interface2[]: GigabitEthernet0/2
Pair name:
```

**Step 17** Press **Enter** to return to the top-level interface editing menu.

```
[1] Remove interface configurations.
[2] Add/Modify Inline Vlan Pairs.
[3] Add/Modify Promiscuous Vlan Groups.
[4] Add/Modify Inline Interface Pairs.
[5] Add/Modify Inline Interface Pair Vlan Groups.
[6] Modify interface default-vlan.
Option:
```

**Step 18** Press **Enter** to return to the top-level editing menu.

```
[1] Edit Interface Configuration
[2] Edit Virtual Sensor Configuration
[3] Display configuration
Option:
```

**Step 19** Enter **2** to edit the virtual sensor configuration.

```
[1] Remove virtual sensor.
[2] Modify "vs0" virtual sensor configuration.
[3] Create new virtual sensor.
Option:
```

**Step 20** Enter **2** to modify the virtual sensor configuration, vs0.

```
Virtual Sensor: vs0
Anomaly Detection: ad0
Event Action Rules: rules0
Signature Definitions: sig0
```

No Interfaces to remove.

```
Unassigned:
Promiscuous:
```

```

[1] GigabitEthernet0/3
[2] GigabitEthernet0/0
Inline Vlan Pair:
[3] GigabitEthernet0/0:1 (Vlans: 200, 300)
Inline Interface Pair:
[4] newPair (GigabitEthernet0/1, GigabitEthernet0/2)
Add Interface:

```

**Step 21** Enter **3** to add inline VLAN pair GigabitEthernet0/0:1.

**Step 22** Enter **4** to add inline interface pair NewPair.

**Step 23** Press **Enter** to return to the top-level virtual sensor menu.

```

Virtual Sensor: vs0
Anomaly Detection: ad0
Event Action Rules: rules0
Signature Definitions: sig0
Inline Vlan Pair:
GigabitEthernet0/0:1 (Vlans: 200, 300)
Inline Interface Pair:
newPair (GigabitEthernet0/1, GigabitEthernet0/2)

[1] Remove virtual sensor.
[2] Modify "vs0" virtual sensor configuration.
[3] Create new virtual sensor.
Option: GigabitEthernet0/1, GigabitEthernet0/2)
Add Interface:

```

**Step 24** Press **Enter** to return to the top-level interface and virtual sensor configuration menu.

```

[1] Edit Interface Configuration
[2] Edit Virtual Sensor Configuration
[3] Display configuration
Option:

```

**Step 25** Enter **yes** if you want to modify the default threat prevention settings.




---

**Note** The sensor comes with a built-in override to add the deny packet event action to high risk rating alerts. If you do not want this protection, disable automatic threat prevention.

---

```

Virtual sensor newVs is configured to prevent high risk threats in inline mode. (Risk
Rating 90-100)
Virtual sensor vs0 is configured to prevent high risk threats in inline mode.(Risk Rating
90-100)
Do you want to disable automatic threat prevention on all virtual sensors?[no]:

```

**Step 26** Enter **yes** to disable automatic threat prevention on all virtual sensors.

**Step 27** Press **Enter** to exit the interface and virtual sensor configuration.

```

The following configuration was entered.
service host
network-settings
host-ip 192.168.1.2/24,192.168.1.1
host-name sensor
telnet-option disabled
sshv1-fallback enabled
ftp-timeout 300
no login-banner-text
exit
time-zone-settings
offset 0

```



```

standard-time-zone-name UTC
exit
summertime-option disabled
ntp-option disabled
exit
service web-server
port 342
exit
service interface
physical-interfaces GigabitEthernet0/0
admin-state enabled
subinterface-type inline-vlan-pair
subinterface 1
description Created via setup by user asmith
vlan1 200
vlan2 300
exit
exit
exit
physical-interfaces GigabitEthernet0/1
admin-state enabled
exit
physical-interfaces GigabitEthernet0/2
admin-state enabled
exit
physical-interfaces GigabitEthernet0/0
admin-state enabled
exit
inline-interfaces newPair
description Created via setup by user asmith
interface1 GigabitEthernet0/1
interface2 GigabitEthernet0/2
exit
exit
service analysis-engine
virtual-sensor newVs
description Created via setup by user cisco
signature-definition newSig
event-action-rules rules0
anomaly-detection
anomaly-detection-name ad0
operational-mode inactive
exit
physical-interface GigabitEthernet0/0
exit
virtual-sensor vs0
physical-interface GigabitEthernet0/0 subinterface-number 1
logical-interface newPair
service event-action-rules rules0
overrides deny-packet-inline
override-item-status Disabled
risk-rating-range 90-100
exit
exit

```

- [0] Go to the command prompt without saving this config.
- [1] Return back to the setup without saving this config.
- [2] Save this configuration and exit setup.

**Step 28** Enter 2 to save the configuration.

```

Enter your selection[2]: 2
Configuration Saved.

```

**Step 29** Reboot the appliance.

```
sensor# reset
Warning: Executing this command will stop all applications and reboot the node.
Continue with reset? []:
```

**Step 30** Enter **yes** to continue the reboot.

**Step 31** Apply the most recent service pack and signature update. You are now ready to configure your appliance for intrusion prevention.

#### For More Information

For the procedure for obtaining the most recent IPS software, see [Obtaining Cisco IPS Software, page 20-1](#).

## Advanced Setup for the ASA 5500-X IPS SSP

You log in to the ASA 5500-X IPS SSP from the adaptive security appliance.

To session in to the ASA 5500-X IPS SSP from the adaptive security appliance, follow these steps:

**Step 1** Log in to the adaptive security appliance.



**Note** If the adaptive security appliance is operating in multi-mode, use the **change system** command to get to the system level prompt before continuing.

**Step 2** Session to the IPS. You have 60 seconds to log in before the session times out.

```
asa# session ips
Opening command session with slot 1.
Connected to slot 1. Escape character sequence is 'CTRL-^X'.
```

**Step 3** Enter your username and password at the login prompt.



**Note** The default username and password are both **cisco**. You are prompted to change them the first time you log in to the module. You must first enter the UNIX password, which is **cisco**. Then you must enter the new password twice.

```
login: cisco
Password:
NOTICE
```

This product contains cryptographic features and is subject to United States and local country laws governing import, export, transfer and use. Delivery of Cisco cryptographic products does not imply third-party authority to import, export, distribute or use encryption. Importers, exporters, distributors and users are responsible for compliance with U.S. and local country laws. By using this product you agree to comply with applicable laws and regulations. If you are unable to comply with U.S. and local laws, return this product immediately.

A summary of U.S. laws governing Cisco cryptographic products may be found at:  
<http://www.cisco.com/wvl/export/crypto/tool/stqrg.html>

If you require further assistance please contact us by sending email to [export@cisco.com](mailto:export@cisco.com).

```

LICENSE NOTICE
There is no license key installed on this IPS platform.
The system will continue to operate with the currently installed
signature set. A valid license must be obtained in order to apply
signature updates. Please go to http://www.cisco.com/go/license
to obtain a new license or install a license.

asa-ips#

```

- Step 4** To escape from a session and return to the adaptive security appliance prompt, do one of the following:
- Enter **exit**.
  - Press **CTRL-Shift-6-x** (represented as **CTRL^X**).

#### For More Information

For the procedure for obtaining the most recent IPS software, see [Obtaining Cisco IPS Software, page 20-1](#).

## Advanced Setup for the ASA 5585-X IPS SSP

You log in to the ASA 5585-X IPS SSP from the adaptive security appliance.

To session in to the ASA 5585-X IPS SSP from the adaptive security appliance, follow these steps:

- Step 1** Log in to the adaptive security appliance.



**Note** If the adaptive security appliance is operating in multi-mode, use the **change system** command to get to the system level prompt before continuing.

- Step 2** Session to the ASA 5585-X IPS SSP. You have 60 seconds to log in before the session times out.

```

asa# session 1
Opening command session with slot 1.
Connected to slot 1. Escape character sequence is 'CTRL-^X'.

```

- Step 3** Enter your username and password at the login prompt.



**Note** The default username and password are both **cisco**. You are prompted to change them the first time you log in to the module. You must first enter the UNIX password, which is **cisco**. Then you must enter the new password twice.

```

login: cisco
Password:
NOTICE
This product contains cryptographic features and is subject to United States and local
country laws governing import, export, transfer and use. Delivery of Cisco cryptographic
products does not imply third-party authority to import, export, distribute or use
encryption. Importers, exporters, distributors and users are responsible for compliance
with U.S. and local country laws. By using this product you agree to comply with
applicable laws and regulations. If you are unable to comply with U.S. and local laws,
return this product immediately.

```

A summary of U.S. laws governing Cisco cryptographic products may be found at:  
<http://www.cisco.com/wwl/export/crypto/tool/stqrg.html>

If you require further assistance please contact us by sending email to [export@cisco.com](mailto:export@cisco.com).

\*\*\*LICENSE NOTICE\*\*\*

There is no license key installed on the system.

Please go to <http://www.cisco.com/go/license> to obtain a new license or install a license.  
 ips-ssp#

- Step 4** To escape from a session and return to the adaptive security appliance prompt, do one of the following:
- Enter **exit**.
  - Press **CTRL-Shift-6-x** (represented as **CTRL^X**).

#### For More Information

For the procedure for obtaining the most recent IPS software, see [Obtaining Cisco IPS Software](#), page 20-1.

## Verifying Initialization



#### Note

The CLI output is an example of what your configuration may look like. It will not match exactly due to the optional setup choices, sensor model, and IPS version you have installed.

To verify that you initialized your sensor, follow these steps:

**Step 1** Log in to the sensor.

**Step 2** View your configuration.

```
sensor# show configuration
! -----
! Current configuration last modified Fri Oct 10 09:41:17 2014
! -----
! Version 7.3(1)
! Host:
! Realm Keys key1.0
! Signature Definition:
! Signature Update S741.0 2013-09-10
! Threat Profile Version 2
! -----
service interface
exit
! -----
service authentication
exit
! -----
service event-action-rules rules0
exit
! -----
service host
network-settings
host-ip 10.106.1.4/24,10.106.1.1
host-name sensor
```

```
telnet-option enabled
access-list 0.0.0.0/0
dns-primary-server disabled
exit
time-zone-settings
offset 0
standard-time-zone-name UTC
exit
exit
! -----
service logger
exit
! -----
service network-access
exit
! -----
service notification
exit
! -----
service signature-definition sig0
exit
! -----
service ssh-known-hosts
exit
! -----
service trusted-certificates
exit
! -----
service web-server
exit
! -----
service anomaly-detection ad0
exit
! -----
service external-product-interface
exit
! -----
service health-monitor
exit
! -----
service global-correlation
exit
! -----
service aaa
exit
! -----
service analysis-engine
exit
sensor#
```



---

**Note** You can also use the **more current-config** command to view your configuration.

---

**Step 3** Display the self-signed X.509 certificate (needed by TLS).

```
sensor# show tls fingerprint
SHA1: 64:9B:AC:DE:21:62:0C:D3:57:2E:9B:E5:3D:04:8F:A7:FD:CD:6F:27
```

- Step 4** Write down the certificate fingerprints. You need the fingerprints to check the authenticity of the certificate when connecting to this sensor with a web browser.
- 

**For More Information**

For the procedure for logging in to the sensor, see [Chapter 2, “Logging In to the Sensor.”](#)



## Setting Up the Sensor

---

This chapter contains procedures for the setting up the sensor, and contains the following sections:

- [Setup Notes and Caveats, page 4-1](#)
- [Understanding Sensor Setup, page 4-2](#)
- [Changing Network Settings, page 4-2](#)
- [Changing the CLI Session Timeout, page 4-14](#)
- [Changing Web Server Settings, page 4-15](#)
- [Configuring Authentication and User Parameters, page 4-18](#)
- [Configuring Time, page 4-36](#)
- [Configuring SSH, page 4-46](#)
- [Configuring TLS, page 4-51](#)
- [Installing the License Key, page 4-58](#)

### Setup Notes and Caveats

The following notes and caveats apply to setting up the sensor:

- By default SSHv1 fallback is enabled.
- When updating the hostname, the CLI prompt of the current session and other existing sessions is not updated with the new hostname immediately. Subsequent CLI login sessions reflect the new hostname in the prompt.
- Slashes are not valid for hostnames, for example, firewall5/ips.
- Telnet is not a secure access service and therefore is disabled by default on the sensor. However, SSH is always running on the sensor and it is a secure service.
- For automatic and global correlation updates to function, you must have either a DNS server or an HTTP proxy server configured at all times.
- DNS resolution is supported for accessing the global correlation update server as well as [www.cisco.com](http://www.cisco.com) for automatic updates.
- The default web server port is 443 if TLS is enabled and 80 if TLS is disabled.
- The **username** command provides username and password authentication for login purposes only. You cannot use this command to remove a user who is logged in to the system. You cannot use this command to remove yourself from the system.

- You cannot use the **privilege** command to give a user service privileges. If you want to give an existing user service privileges, you must remove that user and then use the **username** command to create the service account.
- Do not make modifications to the sensor through the service account except under the direction of TAC. If you use the service account to configure the sensor, your configuration is not supported by TAC. Adding services to the operating system through the service account affects proper performance and functioning of the other IPS services. TAC does not support a sensor on which additional services have been added.
- You should carefully consider whether you want to create a service account. The service account provides shell access to the system, which makes the system vulnerable. However, you can use the service account to create a password if the administrator password is lost. Analyze your situation to decide if you want a service account existing on the system.
- Administrators may need to disable the password recovery feature for security reasons.
- We recommend that you use an NTP server to regulate time on your sensor. You can use authenticated or unauthenticated NTP. For authenticated NTP, you must obtain the NTP server IP address, NTP server key ID, and the key value from the NTP server. You can set up NTP during initialization or you can configure NTP through the CLI, IDM, IME, or ASDM.
- In addition to a valid Cisco.com username and password, you must also have a Cisco Services for IPS service contract before you can apply for a license key.

## Understanding Sensor Setup

Setting up the sensor involves such tasks as changing sensor initialization information, adding and deleting users, configuring time and setting up NTP, creating a service account, configuring SSH and TLS, and installing the license key. You configured most of these settings when you initialized the sensor using the **setup** command.

### For More Information

For more information on using the **setup** command to initialize the sensor, see [Chapter 3, “Initializing the Sensor.”](#)

## Changing Network Settings

After you initialize your sensor, you may need to change some of the network settings that you configured when you ran the **setup** command. This section describes how to change network settings, and contains the following topics:

- [Changing the Hostname, page 4-3](#)
- [Changing the IP Address, Netmask, and Gateway, page 4-4](#)
- [Enabling and Disabling Telnet, page 4-5](#)
- [Changing the Access List, page 4-6](#)
- [Changing the FTP Timeout, page 4-8](#)
- [Adding a Login Banner, page 4-9](#)
- [Configuring the DNS and Proxy Servers for Global Correlation and Automatic Update, page 4-10](#)
- [Enabling SSHv1 Fallback, page 4-13](#)



## Changing the Hostname



### Note

The CLI prompt of the current session and other existing sessions will not be updated with the new hostname. Subsequent CLI login sessions will reflect the new hostname in the prompt.

Use the **host-name** *host\_name* command in the service host submode to change the hostname of the sensor after you have run the **setup** command. The default is sensor.

To change the sensor hostname, follow these steps:

**Step 1** Log in to the sensor using an account with administrator privileges.

**Step 2** Enter network settings submode.

```
sensor# configure terminal
sensor(config)# service host
sensor(config-hos)# network-settings
```

**Step 3** Change the sensor hostname.

```
sensor(config-hos-net)# host-name firesafe
```

**Step 4** Verify the new hostname.

```
sensor(config-hos-net)# show settings
network-settings

host-ip: 192.0.2.1/24,192.0.2.2 default:
192.168.1.2/24,192.168.1.1
host-name: firesafe default: sensor
telnet-option: enabled default: disabled
sshv1-fallback: disabled default: disabled
access-list (min: 0, max: 512, current: 1)

network-address: 0.0.0.0/0

ftp-timeout: 300 seconds <defaulted>
login-banner-text: <defaulted>

sensor(config-hos-net)#
```

**Step 5** To change the hostname back to the default setting, use the **default** form of the command.

```
sensor(config-hos-net)# default host-name
```

**Step 6** Verify the change to the default hostname sensor.

```
sensor(config-hos-net)# show settings
network-settings

host-ip: 192.0.2.1/24,192.0.2.2 default:
192.168.1.2/24,192.168.1.1
host-name: sensor <defaulted>
telnet-option: enabled default: disabled
sshv1-fallback: disabled default: disabled
access-list (min: 0, max: 512, current: 1)

network-address: 0.0.0.0/0

```

```

ftp-timeout: 300 seconds <defaulted>
login-banner-text: <defaulted>

sensor(config-hos-net)#

```

**Step 7** Exit network settings mode.

```

sensor(config-hos-net)# exit
sensor(config-hos)# exit
Apply Changes:[yes]:

```

**Step 8** Press **Enter** to apply the changes or enter **no** to discard them.

## Changing the IP Address, Netmask, and Gateway

Use the **host-ip** *ip\_address/netmask,default\_gateway* command in the service host submode to change the IP address, netmask, and default gateway after you have run the **setup** command. The default is 192.168.1.2/24,192.168.1.1.

The **host-ip** is in the form of IP Address/Netmask/Gateway: X.X.X.X/nn,Y.Y.Y.Y, where X.X.X.X specifies the sensor IP address as a 32-bit address written as 4 octets separated by periods where X = 0-255, nn specifies the number of bits in the netmask, and Y.Y.Y.Y specifies the default gateway as a 32-bit address written as 4 octets separated by periods where Y = 0-255.

To change the sensor IP address, netmask, and default gateway, follow these steps:

**Step 1** Log in to the sensor using an account with administrator privileges.

**Step 2** Enter network settings mode.

```

sensor# configure terminal
sensor(config)# service host
sensor(config-hos)# network-settings

```

**Step 3** Change the sensor IP address, netmask, and default gateway.

```

sensor(config-hos-net)# host-ip 192.0.2.1/24,192.0.2.2

```



**Note** The default gateway must be in the same subnet as the IP address of the sensor or the sensor generates an error and does not accept the configuration change.

**Step 4** Verify the new information.

```

sensor(config-hos-net)# show settings
network-settings

host-ip: 192.0.2.1/24,192.0.2.2
default: 192.168.1.2/24,192.168.1.1
host-name: sensor default: sensor
telnet-option: enabled default: disabled
sshv1-fallback: disabled default: disabled
access-list (min: 0, max: 512, current: 1)

network-address: 0.0.0.0/0

ftp-timeout: 300 seconds <defaulted>

```

```
login-banner-text: <defaulted>
```

**Step 5** To change the information back to the default setting, use the **default** form of the command.

```
sensor(config-hos-net)# default host-ip
```

**Step 6** Verify that the host IP is now the default of 192.168.1.2/24,192.168.1.1.

```
sensor(config-hos-net)# show settings
network-settings

host-ip: 192.168.1.2/24,192.168.1.1 <defaulted>
host-name: sensor default: sensor
telnet-option: enabled default: disabled
sshv1-fallback: disabled default: disabled
access-list (min: 0, max: 512, current: 1)
```

```

network-address: 0.0.0.0/0

```

```

ftp-timeout: 300 seconds <defaulted>
login-banner-text: <defaulted>
```

```

sensor(config-hos-net)#
```

**Step 7** Exit network settings mode.

```
sensor(config-hos-net)# exit
sensor(config-hos)# exit
Apply Changes:[yes]:
```

**Step 8** Press **Enter** to apply the changes or enter **no** to discard them.

## Enabling and Disabling Telnet



### Caution

Telnet is not a secure access service and therefore is disabled by default. However, SSH is always running on the sensor and it is a secure service.

Use the **telnet-option {enabled | disabled}** command in the service host submode to enable Telnet for remote access to the sensor. The default is disabled.

To enable or disable Telnet services, follow these steps:

**Step 1** Log in to the sensor using an account with administrator privileges.

**Step 2** Enter network settings mode.

```
sensor# configure terminal
sensor(config)# service host
sensor(config-hos)# network-settings
```

**Step 3** Enable Telnet services.

```
sensor(config-hos-net)# telnet-option enabled
sensor(config-hos-net)#
```

**Step 4** Verify that Telnet is enabled.

```
sensor(config-hos-net)# show settings
network-settings

host-ip: 192.0.2.1/24,192.0.2.2
default: 192.168.1.2/24,192.168.1.1
host-name: sensor default: sensor
telnet-option: enabled default: disabled
sshd-fallback: disabled default: disabled
access-list (min: 0, max: 512, current: 1)

network-address: 0.0.0.0/0

ftp-timeout: 300 seconds <defaulted>
login-banner-text: <defaulted>

```

**Step 5** Exit network settings mode.

```
sensor(config-hos-net)#
sensor(config-hos-net)# exit
sensor(config-hos)# exit
Apply Changes?[yes]:
```

**Step 6** Press **Enter** to apply the changes or enter **no** to discard them.



**Note**

---

To Telnet to the sensor, you must enable Telnet and configure the access list to allow the Telnet clients to connect.

---

**For More Information**

For the procedure for configuring the access list, see [Changing the Access List, page 4-6](#).

## Changing the Access List

Use the **access-list** *ip\_address/netmask* command in the service host submode to configure the access list, the list of hosts or networks that you want to have access to your sensor. Use the **no** form of the command to remove an entry from the list. The default access list is empty.

The following hosts must have an entry in the access list:

- Hosts that need to Telnet to your sensor.
- Hosts that need to use SSH with your sensor.
- Hosts, such as the IDM and the IME, that need to access your sensor from a web browser.
- Management stations, such as the CSM, that need access to your sensor.
- If your sensor is a master blocking sensor, the IP addresses of the blocking forwarding sensors must have an entry in the list.

### Changing the Access List

To modify the access list, follow these steps:

**Step 1** Log in to the sensor using an account with administrator privileges.

**Step 2** Enter network settings mode.

```
sensor# configure terminal
sensor(config)# service host
sensor(config-hos)# network-settings
```

**Step 3** Add an entry to the access list. The netmask for a single host is 32.

```
sensor(config-hos-net)# access-list 192.0.2.110/32
```

**Step 4** Verify the change you made to the access-list.

```
sensor(config-hos-net)# show settings
network-settings

host-ip: 192.168.1.2/24,192.168.1.1 <defaulted>
host-name: sensor <defaulted>
telnet-option: enabled default: disabled
sshv1-fallback: disabled default: disabled
access-list (min: 0, max: 512, current: 2)

network-address: 10.1.9.0/24

network-address: 192.0.2.110/32

ftp-timeout: 300 seconds <defaulted>
login-banner-text: <defaulted>

```

**Step 5** Remove the entry from the access list.

```
sensor(config-hos-net)# no access-list 192.0.2.110/32
```

**Step 6** Verify that the host is no longer in the list.

```
sensor(config-hos-net)# show settings
network-settings

host-ip: 192.168.1.2/24,192.168.1.1 <defaulted>
host-name: sensor <defaulted>
telnet-option: enabled default: disabled
sshv1-fallback: disabled default: disabled
access-list (min: 0, max: 512, current: 1)

network-address: 10.1.9.0/24

ftp-timeout: 300 seconds <defaulted>
login-banner-text: <defaulted>

sensor(config-hos-net)#
```

**Step 7** Change the value back to the default.

```
sensor(config-hos-net)# default access-list
```

**Step 8** Verify the value has been set back to the default.

```
sensor(config-hos-net)# show settings
```

```

network-settings

host-ip: 192.168.1.2/24,192.168.1.1 <defaulted>
host-name: sensor <defaulted>
telnet-option: enabled default: disabled
sshd-fallback: disabled default: disabled
access-list (min: 0, max: 512, current: 0)

ftp-timeout: 300 seconds <defaulted>
login-banner-text: <defaulted>

sensor(config-hos-net)#

```

**Step 9** Exit network settings mode.

```

sensor(config-hos-net)# exit
sensor(config-hos)# exit
Apply Changes?[yes]:

```

**Step 10** Press **Enter** to apply the changes or enter **no** to discard them.

## Changing the FTP Timeout



### Note

You can use the FTP client for downloading updates and configuration files from your FTP server.

Use the **ftp-timeout** command in the service host submode to change the number of seconds that the FTP client waits before timing out when the sensor is communicating with an FTP server. The default is 300 seconds.

To change the FTP timeout, follow these steps:

**Step 1** Log in to the sensor using an account with administrator privileges.

**Step 2** Enter network settings mode.

```

sensor# configure terminal
sensor(config)# service host
sensor(config-hos)# network-settings

```

**Step 3** Change the number of seconds of the FTP timeout.

```

sensor(config-hos-net)# ftp-timeout 500

```

**Step 4** Verify the FTP timeout change.

```

sensor(config-hos-net)# show settings
network-settings

host-ip: 192.0.2.1/24,192.0.2.2
default: 192.168.1.2/24,192.168.1.1
host-name: sensor default: sensor
telnet-option: enabled default: disabled
sshd-fallback: disabled default: disabled
access-list (min: 0, max: 512, current: 1)

network-address: 0.0.0.0/0

```

```

ftp-timeout: 500 seconds default: 300
login-banner-text: <defaulted>

sensor(config-hos-net)#

```

**Step 5** Change the value back to the default.

```
sensor(config-hos-net)# default ftp-timeout
```

**Step 6** Verify the value has been set back to the default.

```

sensor(config-hos-net)# show settings
network-settings

host-ip: 192.0.2.1/24,192.0.2.2
default: 192.168.1.2/24,192.168.1.1
host-name: sensor default: sensor
telnet-option: enabled default: disabled
sshd1-fallback: disabled default: disabled
access-list (min: 0, max: 512, current: 1)

network-address: 0.0.0.0/0

ftp-timeout: 300 seconds <defaulted>
login-banner-text: <defaulted>

sensor(config-hos-net)#

```

**Step 7** Exit network settings mode.

```

sensor(config-hos-net)# exit
sensor(config-hos)# exit
Apply Changes:[yes]:

```

**Step 8** Press **Enter** to apply the changes or enter **no** to discard them.

## Adding a Login Banner

Use the **login-banner-text** *text\_message* command to add a login banner that the user sees during login. There is no default. When you want to start a new line in your message, press **Ctrl-V Enter**.

To add a login banner, follow these steps:

**Step 1** Log in to the sensor using an account with administrator privileges.

**Step 2** Enter network settings mode.

```

sensor# configure terminal
sensor(config)# service host
sensor(config-hos)# network-settings

```

**Step 3** Add the banner login text.

```
sensor(config-hos-net)# login-banner-text This is the banner login text message.
```

**Step 4** Verify the banner login text message.

```
sensor(config-hos-net)# show settings
```

```

network-settings

host-ip: 192.0.2.1/24,192.0.2.2
default: 192.168.1.2/24,192.168.1.1
host-name: sensor default: sensor
telnet-option: enabled default: disabled
sshy1-fallback: disabled default: disabled
access-list (min: 0, max: 512, current: 1)

network-address: 0.0.0.0/0

ftp-timeout: 300 seconds <defaulted>
login-banner-text: This is the banner login text message. default:

sensor(config-hos-net)#

```

**Step 5** To remove the login banner text, use the **no** form of the command.

```
sensor(config-hos-net)# no login-banner-text
```

**Step 6** Verify the login text has been removed.

```

sensor(config-hos-net)# show settings
network-settings

host-ip: 192.0.2.1/24,192.0.2.2
default: 192.168.1.2/24,192.168.1.1
host-name: sensor default: sensor
telnet-option: enabled default: disabled
sshy1-fallback: disabled default: disabled
access-list (min: 0, max: 512, current: 1)

network-address: 0.0.0.0/0

ftp-timeout: 300 seconds <defaulted>
login-banner-text: default:

sensor(config-hos-net)#

```

**Step 7** Exit network settings mode.

```

sensor(config-hos-net)# exit
sensor(config-hos)# exit
Apply Changes?[yes]:

```

**Step 8** Press **Enter** to apply the changes or enter **no** to discard them.

## Configuring the DNS and Proxy Servers for Global Correlation and Automatic Update

Use the **http-proxy**, **dns-primary-server**, **dns-secondary-server**, and **dns-tertiary-server** commands in network-settings submode to configure servers to support the automatic update and global correlation features.

You must configure either an HTTP proxy server or DNS server to support automatic update and global correlation. You may need a proxy server to download automatic update and global correlation updates if you use proxy in your network. If you are using a DNS server, you must configure at least one DNS



server and it must be reachable for automatic update and global correlation updates to be successful. You can configure other DNS servers as backup servers. DNS queries are sent to the first server in the list. If it is unreachable, DNS queries are sent to the next configured DNS server.

**Caution**

For automatic and global correlation updates to function, you must have either a DNS server or an HTTP proxy server configured at all times.

**Caution**

DNS resolution is supported for accessing the global correlation update server as well as [www.cisco.com](http://www.cisco.com) for automatic updates.

The following commands apply:

- **http-proxy {no-proxy | proxy-sensor}**—Configures the HTTP proxy server:
  - **address** *ip\_address* —Specifies the IP address of the HTTP proxy server.
  - **port** *port\_number* —Specifies the port number of the HTTP proxy server.
- **dns-primary-server {enabled | disabled}**—Enables a DNS primary server:
  - **address** *ip\_address* —Specifies the IP address of the DNS primary server.
- **dns-secondary-server {enabled | disabled}**—Enables a DNS secondary server:
  - **address** *ip\_address* —Specifies the IP address of the DNS secondary server.
- **dns-tertiary-server {enabled | disabled}**—Enables the DNS tertiary server:
  - **address** *ip\_address* —Specifies the IP address of the DNS tertiary server.

### Configuring DNS and Proxy Servers for Automatic Update and Global Correlation

To configure DNS and proxy servers to support automatic update and global correlation, follow these steps:

**Step 1** Log in to the sensor using an account with administrator privileges.

**Step 2** Enter network settings submode.

```
sensor# configure terminal
sensor(config)# service host
sensor(config-hos)# network-settings
```

**Step 3** Enable a proxy or DNS server to support global correlation:

a. Enable a proxy server.

```
sensor(config-hos-net)# http-proxy proxy-server
sensor(config-hos-net-pro)# address 10.10.10.1
sensor(config-hos-net-pro)# port 65
sensor(config-hos-net-pro)#
```

b. Enable a DNS server.

```
sensor(config-hos-net)# dns-primary-server enabled
sensor(config-hos-net-ena)# address 10.10.10.1
sensor(config-hos-net-ena)#
```

**Step 4** Verify the settings.

```
sensor(config-hos-net)# show settings
network-settings
```

```

host-ip: 10.89.147.24/25,10.89.147.126 default: 192.168.1.2/24,192.168.1.1
host-name: sensor <defaulted>
telnet-option: enabled default: disabled
sshv1-fallback: disabled default: disabled
access-list (min: 0, max: 512, current: 1)

network-address: 0.0.0.0/0

ftp-timeout: 300 seconds <defaulted>
login-banner-text: <defaulted>
dns-primary-server

enabled

address: 10.10.10.1

dns-secondary-server

disabled

dns-tertiary-server

disabled

http-proxy

proxy-server

address: 10.10.10.1
port: 65

sensor(config-hos-net)#

```

**Step 5** Exit network settings mode.

```

sensor(config-hos-net)# exit
sensor(config-hos)# exit
Apply Changes?[yes]:

```

**Step 6** Press **Enter** to apply the changes or enter **no** to discard them.

---

#### For More Information

- For the procedure for configuring automatic update, see [Configuring Automatic Upgrades, page 21-7](#).
- For more information on global correlation features, see [Chapter 10, “Configuring Global Correlation.”](#)

## Enabling SSHv1 Fallback



### Note

The IPS supports a management connection through both SSHv1 and SSHv2.

Use the `sshv1-fallback {enabled | disabled}` command in the service host submode to enable/disable the sensor to fall back to SSH protocol version 1. Fallback to SSHv1 is provided in case the peer client/server does not support SSHv2.

To enable or disable SSHv1 fallback, follow these steps:

**Step 1** Log in to the sensor using an account with administrator privileges.

**Step 2** Enter network settings mode.

```
sensor# configure terminal
sensor(config)# service host
sensor(config-hos)# network-settings
```

**Step 3** Enable SSHv1 fallback.

```
sensor(config-hos-net)# sshv1-fallback enabled
sensor(config-hos-net)#
```

**Step 4** Verify that SSHv1 fallback is enabled.

```
sensor(config-hos-net)# show settings
network-settings

host-ip: 10.106.164.52/24,10.106.164.1 default: 192.168.1.2/24,192.168.1.1
host-name: p32-ips4240-52 default: sensor
telnet-option: enabled default: disabled
sshv1-fallback: enabled default: enabled
access-list (min: 0, max: 512, current: 1)

network-address: 0.0.0.0/0

ftp-timeout: 300 seconds <defaulted>
login-banner-text: mmmm default:
sensor(config-hos-net)#
```

**Step 5** Exit network settings mode.

```
sensor(config-hos-net)# exit
sensor(config-hos)# exit
Apply Changes:[yes]:
```

**Step 6** Press **Enter** to apply the changes or enter **no** to discard them.

### For More Information

For more information about configuring SSH, see [Configuring SSH, page 4-46](#).

## Changing the CLI Session Timeout

Use the **cli-inactivity-timeout** command in the service authentication submode to change the number of seconds that the CLI waits before timing out. Setting the CLI session timeout increases the security of a CLI session. The default is 0 seconds, which means that it is an unlimited value and thus will never time out. The valid range is 0 to 100,000 minutes.

To change the CLI session timeout, follow these steps:

**Step 1** Log in to the sensor using an account with administrator privileges.

**Step 2** Enter authentication mode.

```
sensor# configure terminal
sensor(config)# service authentication
```

**Step 3** Change the number of seconds of the CLI session timeout.

```
sensor(config-aut)# cli-inactivity-timeout 5000
```

**Step 4** Verify the CLI session timeout change.

```
sensor(config-aut)# show settings
attemptLimit: 0 <defaulted>
password-strength

size: 8-64 <defaulted>
digits-min: 0 <defaulted>
uppercase-min: 0 <defaulted>
lowercase-min: 0 <defaulted>
other-min: 0 <defaulted>
number-old-passwords: 0 <defaulted>

permit-packet-logging: true <defaulted>
cli-inactivity-timeout: 5000 default: 0
sensor(config-aut)#
```

**Step 5** Change the value back to the default.

```
sensor(config-aut)# default cli-inactivity-timeout
```

**Step 6** Verify the value has been set back to the default.

```
sensor(config-aut)# show settings
attemptLimit: 0 <defaulted>
password-strength

size: 8-64 <defaulted>
digits-min: 0 <defaulted>
uppercase-min: 0 <defaulted>
lowercase-min: 0 <defaulted>
other-min: 0 <defaulted>
number-old-passwords: 0 <defaulted>

permit-packet-logging: true <defaulted>
cli-inactivity-timeout: 0 <defaulted>
sensor(config-aut)#
```

**Step 7** Exit authentication mode.

```
sensor(config-aut)# exit
Apply Changes?[yes]:
```

**Step 8** Press **Enter** to apply the changes or enter **no** to discard them.

---

## Changing Web Server Settings

**Note**

The default web server port is 443 if TLS is enabled and 80 if TLS is disabled.

---

After you run the **setup** command, you can change the following web server settings: the web server port, whether TLS encryption is being used, the HTTP server header message, restriction of TLS client ciphers, web session inactivity timeout, and logging of web session inactivity timeouts.

HTTP is the protocol that web clients use to make requests from web servers. The HTTP specification requires a server to identify itself in each response. Attackers sometimes exploit this protocol feature to perform reconnaissance. If the IPS web server identified itself by providing a predictable response, an attacker might learn that an IPS sensor is present.

We recommend that you not reveal to attackers that you have an IPS sensor. Change the server-id to anything that does not reveal any information, especially if your web server is available to the Internet. For example, if you forward a port through a firewall so you can monitor a sensor remotely, you need to set the server-id.

The following commands apply:

- **enable-tls {false | true}**—Enables encryption (TLSv1) on the system. The default is enabled.
- **enable-web-session-inactivity-timeout-logging {false | true}**—Enables logging for web session inactivity timeouts. The default is disabled.
- **port *port\_number***—Specifies the port on which the web server listens for connections. The valid range is 1 to 65535. The default is 443.
- **server-id *server\_id***—Specifies the textual message the web server returns in the HTTP Server header. The default is HTTP/1.1 compliant configurable-service.
- **tls-client-ciphers-restriction {false | true}**—Enables the client to use only restricted mode ciphers; disabling allows all ciphers. The default is enabled. When IPS acts as a TLS client, you can configure restriction on the TLS ciphers.



**Note** Changes take place for the next sessions only. The current web session is not affected.

---

When enabled, the client can use the following restricted ciphers:

- TLS\_RSA\_WITH\_AES\_128\_CBC\_SHA
- TLS\_RSA\_WITH\_AES\_256\_CBC\_SHA
- TLS\_DHE\_RSA\_WITH\_AES\_128\_CBC\_SHA
- TLS\_DHE\_RSA\_WITH\_AES\_256\_CBC\_SHA

When disabled, the client can use the following ciphers:

- TLS\_DHE\_RSA\_WITH\_AES\_256\_CBC\_SHA256
- TLS\_DHE\_DSS\_WITH\_AES\_256\_CBC\_SHA256
- TLS\_DHE\_RSA\_WITH\_AES\_256\_CBC\_SHA

- TLS\_DHE\_DSS\_WITH\_AES\_256\_CBC\_SHA
  - TLS\_RSA\_WITH\_AES\_256\_CBC\_SHA
  - TLS\_ECDHE\_RSA\_WITH\_AES\_256\_CBC\_SHA
  - TLS\_ECDH\_RSA\_WITH\_AES\_256\_CBC\_SHA
  - TLS\_ECDHE\_ECDSA\_WITH\_AES\_256\_CBC\_SHA
  - TLS\_ECDH\_ECDSA\_WITH\_AES\_256\_CBC\_SHA
  - TLS\_ECDHE\_RSA\_WITH\_3DES\_EDE\_CBC\_SHA
  - TLS\_ECDH\_RSA\_WITH\_3DES\_EDE\_CBC\_SHA
  - TLS\_ECDHE\_ECDSA\_WITH\_3DES\_EDE\_CBC\_SHA
  - TLS\_ECDH\_ECDSA\_WITH\_3DES\_EDE\_CBC\_SHA
  - TLS\_DHE\_RSA\_WITH\_3DES\_EDE\_CBC\_SHA
  - TLS\_DHE\_DSS\_WITH\_3DES\_EDE\_CBC\_SHA
  - TLS\_RSA\_WITH\_3DES\_EDE\_CBC\_SHA
  - TLS\_DHE\_RSA\_WITH\_AES\_128\_CBC\_SHA256
  - TLS\_DHE\_DSS\_WITH\_AES\_128\_CBC\_SHA256
  - TLS\_DHE\_RSA\_WITH\_AES\_128\_CBC\_SHA
  - TLS\_DHE\_DSS\_WITH\_AES\_128\_CBC\_SHA
  - TLS\_RSA\_WITH\_AES\_128\_CBC\_SHA
  - TLS\_ECDHE\_RSA\_WITH\_AES\_128\_CBC\_SHA
  - TLS\_ECDH\_RSA\_WITH\_AES\_128\_CBC\_SHA
  - TLS\_ECDHE\_ECDSA\_WITH\_AES\_128\_CBC\_SHA
  - TLS\_ECDH\_ECDSA\_WITH\_AES\_128\_CBC\_SHA
- **websession-inactivity-timeout seconds**—Specifies the duration in seconds at which inactive web sessions time out. The valid range is 600 to 3600 seconds. The default is 3600 seconds.

### Changing Web Server Settings

To change the web server settings, follow these steps:

---

**Step 1** Log in to the sensor using an account with administrator privileges.

**Step 2** Enter web server mode.

```
sensor# configure terminal
sensor(config)# service web-server
```

**Step 3** Change the port number.

```
sensor(config-web)# port 8080
```

If you change the port number from the default of 443 to 8080, you receive this message:

```
Warning: The web server's listening port number has changed from 443 to 8080. This change
will not take effect until the web server is re-started
```

**Step 4** Enable TLS.

```
sensor(config-web)# enable-tls true
```

If you disable TLS, you receive this message:

```
Warning: TLS protocol support has been disabled. This change will not take effect until
the web server is re-started.
```

**Step 5** Change the HTTP server header.

```
sensor(config-web)# server-id Nothing to see here. Move along.
```

**Step 6** Specify the web session inactivity timeout.

```
sensor(config-web)# websession-inactivity-timeout 800
```

**Step 7** Turn on logging for web session inactivity timeouts.

```
sensor(config-web)# enable-websession-inactivity-timeout-logging true
```

**Step 8** Turn on TLS client ciphers restriction.

```
sensor(config-web)# tls-client-ciphers-restriction enable
```

**Step 9** Verify the web server changes.

```
sensor(config-web)# show settings
enable-tls: true default: true
strict-tls-server-validation: disable <defaulted>
port: 8080 default: 443
server-id: Nothing to see here. Move along. default: HTTP/1.1 compliant
configurable-service (min: 0, max: 99, current: 0)

websession-inactivity-timeout: 800 default: 3600
enable-websession-inactivity-timeout-logging: true default: false
tls-client-ciphers-restriction: enable default: enable
sensor(config-web)#
```

**Step 10** To revert to the defaults, use the **default** form of the commands.

```
sensor(config-web)# default port
sensor(config-web)# default enable-tls
sensor(config-web)# default server-id
```

**Step 11** Verify the defaults have been replaced.

```
sensor(config-web)# show settings
enable-tls: true <defaulted>
port: 443 <defaulted>
server-id: HTTP/1.1 compliant <defaulted>
configurable-service (min: 0, max: 99, current: 0)

websession-inactivity-timeout: 3600 <defaulted>
enable-websession-inactivity-timeout-logging: false <defaulted>
tls-client-ciphers-restriction: enable <defaulted>
sensor(config-web)#
```

**Step 12** Exit web server submenu.

```
sensor(config-web)# exit
Apply Changes:[yes]:
```

**Step 13** Press **Enter** to apply the changes or enter **no** to discard them.

**Note**

If you change the port or enable TLS settings, you must reset the sensor to make the web server use the new settings.

**For More Information**

- For the procedure for enabling SSHv1 fallback, see [Enabling SSHv1 Fallback, page 4-13](#).
- For the procedure for resetting the appliance, see [Resetting the Appliance, page 17-46](#).
- For the procedure for resetting the ASA 5500-X IPS SSP, see [Reloading, Shutting Down, Resetting, and Recovering the ASA 5500-X IPS SSP, page 18-12](#).
- For the procedure for resetting the ASA 5585-X IPS SSP, see [Reloading, Shutting Down, Resetting, and Recovering the ASA 5585-X IPS SSP, page 19-12](#).

## Configuring Authentication and User Parameters

This section explains how to create users, configure RADIUS authentication, create the service account, configure passwords, specify privilege level, view a list of users, configure password policy, and lock and unlock user accounts. It contains the following topics:

- [Adding and Removing Users, page 4-18](#)
- [Configuring Authentication, page 4-20](#)
- [Configuring Packet Command Restriction, page 4-26](#)
- [Creating the Service Account, page 4-28](#)
- [The Service Account and RADIUS Authentication, page 4-29](#)
- [RADIUS Authentication Functionality and Limitations, page 4-30](#)
- [Configuring Passwords, page 4-30](#)
- [Changing User Privilege Levels, page 4-31](#)
- [Showing User Status, page 4-32](#)
- [Configuring the Password Policy, page 4-32](#)
- [Locking User Accounts, page 4-34](#)
- [Unlocking User Accounts, page 4-35](#)

## Adding and Removing Users

Use the **username** command to create users on the local system. You can add a new user, set the privilege level—administrator, operator, viewer—and set the password for the new user. Use the **no** form of this command to remove a user from the system. This removes the user from CLI and web access.

**Caution**

The **username** command provides username and password authentication for login purposes only. You cannot use this command to remove a user who is logged in to the system. You cannot use this command to remove yourself from the system.



If you do not specify a password, the system prompts you for one. Use the **password** command to change the password for existing users. Use the **privilege** command to change the privilege for existing users.

The username follows the pattern `^[A-Za-z0-9()+:.,_-]+`, which means the username must start with a letter or number, and can include any letter A to Z (capital or small), any number 0 to 9, - and \_, and can contain 1 to 64 characters. A valid password is 8 to 32 characters long. All characters except space are allowed.

You receive the following error messages if you do not create a valid password:

- Error: `setEnabledAuthenticationTokenStatus` : The password is too short.
- Error: `setEnabledAuthenticationTokenStatus` : Failure setting the account's password: it does not contain enough DIFFERENT characters

**Note**

You cannot use the **privilege** command to give a user service privileges. If you want to give an existing user service privileges, you must remove that user and then use the **username** command to create the service account.

To add and remove users, follow these steps:

**Step 1** Log in to the CLI using an account with administrator privileges.

**Step 2** Enter configuration mode.

```
sensor# configure terminal
```

**Step 3** Specify the parameters for the user.

```
sensor(config)# username username password password privilege
administrator/operator/viewer
```

For example, to add the user “tester” with a privilege level of administrator and the password “testpassword,” enter the following command:

**Note**

If you do not want to see the password in clear text, wait for the password prompt. Do not enter the password along with the username and privilege.

```
sensor(config)# username tester privilege administrator
Enter Login Password: *****
Re-enter Login Password: *****
sensor(config)#
```

**Note**

If you do not specify a privilege level for the user, the user is assigned the default viewer privilege.

**Step 4** Verify that the user has been added. A list of users is displayed.

```
sensor(config)# exit
sensor# show users all
CLI ID User Privilege
* 13491 cisco administrator
jsmith operator
jtaylor service
jroberts viewer
sensor#
```

**Step 5** To remove a user, use the **no** form of the command.

```
sensor# configure terminal
sensor(config)# no username jsmith
```



**Note** You cannot use this command to remove yourself from the system.

**Step 6** Verify that the user has been removed. The user `jsmith` has been removed.

```
sensor(config)# exit
sensor# show users all
 CLI ID User Privilege
* 13491 cisco administrator
 jtaylor service
 jroberts viewer
sensor#
```

#### For More Information

- For the procedure for creating the service account, see [Creating the Service Account, page 4-28](#).
- For the procedure for configuring local or RADIUS authentication, see [Configuring Authentication, page 4-20](#).

## Configuring Authentication



#### Caution

Make sure you have a RADIUS server already configured before you configure RADIUS authentication on the sensor. IPS has been tested with CiscoSecure ACS 4.2 and 5.1 servers. Refer to your RADIUS server documentation for information on how to set up a RADIUS server.

You can create and remove users from the local sensor. You can only modify one user account at a time. Each user is associated with a role that controls what that user can and cannot modify. The requirements that must be used for user passwords are set with the **password** command.

Users are authenticated through AAA either locally or through RADIUS servers. Local authentication is enabled by default. You must configure RADIUS authentication before it is active.

You must specify the user role that is authenticated through RADIUS either by configuring the user role on the RADIUS server or specifying a default user role. The username and password are sent in an authentication request to the configured RADIUS server. The response of the server determines whether the login is authenticated.



#### Note

If the sensor is not configured to use a default user role and the sensor user role information is not in the Accept Message of the CiscoSecure ACS server, the sensor rejects RADIUS authentication even if the CiscoSecure ACS server accepts the username and password.

You can configure a primary RADIUS server and a secondary RADIUS server. The secondary RADIUS server authenticates and authorizes users if the primary RADIUS server is unresponsive.

You can also configure the sensor to use local authentication (local fallback) if no RADIUS servers are responding. In this case, the sensor authenticates against the locally configured user accounts. The sensor will only use local authentication if the RADIUS servers are not available, not if the RADIUS server rejects the authentication requests of the user. You can also configure how users connected through the console port are authenticated—through local user accounts, through RADIUS first and if that fails through local user accounts, or through RADIUS alone.

To configure a RADIUS server, you must have the IP address, port, and shared secret of the RADIUS server. You must also either have the NAS-ID of the RADIUS server, or have the RADIUS server configured to authenticate clients without a NAS-ID or with the default IPS NAS-ID of `cisco-ips`.

**Note**

Enabling RADIUS authentication on the sensor does not disconnect already established connections. RADIUS authentication is only enforced for new connections to the sensor. Existing CLI, IDM, and IME connections remain established with the login credentials used prior to configuring RADIUS authentication. To force disconnection of these established connections, you must reset the sensor after RADIUS is configured.

**RADIUS Authentication Options**

Use the `aaa` command in service `aaa` submode to configure either local authentication or authentication using a RADIUS server.

The following commands apply:

- **local**—Lets you specify local authentication. To continue to create users, use the `password` command.
- **radius**—Lets you specify RADIUS as the method of authentication:
  - **nas-id**—Identifies the service requesting authentication. The value can be `no nas-id`, `cisco-ips`, or a NAS-ID already configured on the RADIUS server. The default is `cisco-ips`.
  - **default-user-role**—Lets you assign a default user role on the sensor that is only applied when there is NOT a Cisco `av` pair specifying the user role. The value can be `unspecified`, `viewer`, `operator`, or `administrator`. Service cannot be the default user role. The default is `unspecified`.

If you do not want to configure a default user role on the sensor that is applied in the absence of a Cisco `av` pair, you need to configure the Cisco IOS/PIX 6.x RADIUS Attributes [009\001] `cisco-av-pair` under the group or user profile with one of the following commands: `ips-role=viewer`, `ips-role=operator`, `ips-role=administrator`, `ips-role=service`, or `ips-role=unspecified`. The default is `ips-role=unspecified`.

**Note**

If the sensor is not configured to use a default user role and the sensor user role information is not in the Accept Message of the CiscoSecure ACS server, the sensor rejects RADIUS authentication even if the CiscoSecure ACS server accepts the username and password.

**Note**

The default user role is used only when the user has not been configured with a specific role on the ACS server. Local users are always configured with a specific role so the default user role will never apply to locally authenticated users.

**Caution**

Do not add multiple Cisco av-pairs with the same key or you cannot log in. You must have only one instance of ips-role=value. For example, do not use the following configuration:

```
ips-role= administer
ips-role=ad
```

- **local-fallback {enabled | disabled}**—Lets you default to local authentication if the RADIUS servers are not responding. The default is **enabled**.
- **primary-server**—Lets you configure the main RADIUS server:
  - **server-address**—IP address of the RADIUS server.
  - **server-port**—Port of the RADIUS server. If not specified, the default RADIUS port is used.
  - **timeout** (seconds)—Specifies the number of seconds the sensor waits for a response from a RADIUS server before it considers the server to be unresponsive.
  - **shared-secret**—The secret value configured on the RADIUS server. You must obtain the secret value of the RADIUS server to enter with the **shared-secret** command.



**Note** You must have the same secret value configured on both the RADIUS server and the IPS sensor so that the server can authenticate the requests of the client and the client can authenticate the responses of the server.

- **secondary-server {enabled | disabled}**— (Optional) Lets you configure a secondary RADIUS server:
  - **server-address**—IP address of the RADIUS server.
  - **server-port**—Port of the RADIUS server. If not specified, the default RADIUS port is used.
  - **timeout** (seconds)—Specifies the number of seconds the sensor waits for a response from a RADIUS server before it considers the server to be unresponsive.
  - **shared-secret**—The secret value configured on the RADIUS server. You must obtain the secret value of the RADIUS server to enter with the **shared-secret** command.



**Note** You must have the same secret value configured on both the RADIUS server and the IPS sensor so that the server can authenticate the requests of the client and the client can authenticate the responses of the server.

- **console-authentication**—Lets you choose how users connected through the console port are authenticated:
  - **local**—Users connected through the console port are authenticated through local user accounts.
  - **radius-and-local**—Users connected through the console port are authenticated through RADIUS first. If RADIUS fails, local authentication is attempted. This is the default.
  - **radius**—Users connected through the console port are authenticated by RADIUS. If you also have **local-fallback** enabled, users can also be authenticated through the local user accounts.

### Configuring Local or RADIUS Authentication



#### Caution

Make sure you have a RADIUS server already configured before you configure RADIUS authentication on the sensor. IPS has been tested with CiscoSecure ACS 4.2 and 5.1 servers. Refer to your RADIUS server documentation for information on how to set up a RADIUS server.



#### Note

Enabling RADIUS authentication on the sensor does not disconnect already established connections. RADIUS authentication is only enforced for new connections to the sensor. Existing CLI, IDM, and IME connections remain established with the login credentials used prior to configuring RADIUS authentication. To force disconnection of these established connections, you must reset the sensor after RADIUS is configured.

To configure local or RADIUS AAA authentication on the sensor, follow these steps:

**Step 1** Log in to the CLI using an account with administrator privileges.

**Step 2** Enter configuration mode.

```
sensor# configure terminal
```

**Step 3** Enter AAA submode.

```
sensor(config)# service aaa
sensor(config-aaa)#
```

**Step 4** Configure local authentication. To continue to create users on the local system, enter **yes** to save your configuration, and use the **username** command in configure terminal mode. To configure AAA RADIUS authentication, go to Step 5.

```
sensor(config-aaa)# aaa local
sensor(config-aaa)# exit
Apply Changes?[yes]:yes
```

**Step 5** Configure AAA RADIUS authentication:

- a. Enter RADIUS authentication submode.

```
sensor(config-aaa)# aaa radius
sensor(config-aaa-rad)#
```

- b. Enter the Network Access ID. The NAS-ID is an identifier that clients send to servers to communicate the type of service they are attempting to authenticate. The value can be **no nas-id**, **cisco-ips**, or a NAS-ID already configured on the RADIUS server. The default is **cisco-ips**.

```
sensor(config-aaa-rad)# nas-id cisco-ips
sensor(config-aaa-rad)#
```

- c. (Optional) Configure a default user role if you are not configuring a Cisco av pair. You can configure a default user role on the sensor that is only applied when there is NOT a Cisco av pair specifying the user role. The values are **unspecified**, **viewer**, **operator**, or **administrator**. The default is **unspecified**.

```
sensor(config-aaa-rad)# default-user-role operator
sensor(config-aaa-rad)#
```



#### Note

Service cannot be the default role.

- d. Configure a Cisco av pair. If you do not want to configure a default user role on the sensor that is applied in the absence of a Cisco av pair, you need to configure the Cisco IOS/PIX 6.x RADIUS Attributes [009\001] cisco-av-pair under the group or user profile with one of the following commands:

- **ips-role=viewer**
- **ips-role=operator**
- **ips-role=administrator**
- **ips-role=service**



**Note** If the sensor is not configured to use a default user role and the sensor user role information is not in the Accept Message of the CiscoSecure ACS server, the sensor rejects RADIUS authentication even if the CiscoSecure ACS server accepts the username and password.



**Note** The default user role is used only when the user has not been configured with a specific role on the ACS server. Local users are always configured with a specific role so the default user role will never apply to locally authenticated users.



**Caution**

Do not add multiple Cisco av-pairs with the same key or you cannot log in. You must have only one instance of ips-role=value. For example, do not use the following configuration:

```
ips-role= administrator
ips-role=ad
```

- e. Configure the sensor to switch over to local authentication if the RADIUS server becomes unresponsive.

```
sensor(config-aaa-rad)# local-fallback enabled
sensor(config-aaa-rad)#
```

**Step 6** Configure the primary RADIUS server:

- a. Enter primary server submode.

```
sensor(config-aaa-rad)# primary-server
sensor(config-aaa-rad-pri)#
```

- b. Enter the RADIUS server IP address.

```
sensor(config-aaa-rad-pri)# server-address 10.1.2.3
sensor(config-aaa-rad-pri)#
```

- c. Enter the RADIUS server port. If not specified, the default RADIUS port is used.

```
sensor(config-aaa-rad-pri)# server-port 1812
sensor(config-aaa-rad-pri)#
```

- d. Enter the amount of time in seconds you want to wait for the RADIUS server to respond.

```
sensor(config-aaa-rad-pri)# time-out 5
sensor(config-aaa-rad-pri)#
```

- e. Enter the secret value that you obtained from the RADIUS server. The shared secret is a piece of data known only to the parties involved in a secure communication.

```
sensor(config-aaa-rad-pri)# shared-secret kkkk
sensor(config-aaa-rad-pri)#
```



**Note** You must have the same secret value configured on both the RADIUS server and the IPS sensor so that the server can authenticate the requests of the client and the client can authenticate the responses of the server.

**Step 7** (Optional) Enable a secondary RADIUS server to perform authentication in case the primary RADIUS server is not responsive:

- a. Enter secondary server submode.

```
sensor(config-aaa-rad)# secondary-server enabled
sensor(config-aaa-rad-sec)#
```

- b. Enter the IP address of the second RADIUS server.

```
sensor(config-aaa-rad-sec)# server-address 10.4.5.6
sensor(config-aaa-rad-sec)#
```

- c. Enter the RADIUS server port. If not specified, the default RADIUS port is used.

```
sensor(config-aaa-rad-sec)# server-port 1812
sensor(config-aaa-rad-sec)#
```

- d. Enter the amount of time in seconds you want to wait for the RADIUS server to respond.

```
sensor(config-aaa-rad-sec)# time-out 8
sensor(config-aaa-rad-sec)#
```

- e. Enter the secret value you obtained for this RADIUS server. The shared secret is a piece of data known only to the parties involved in a secure communication.

```
sensor(config-aaa-rad-sec)# shared-secret yyyyy
sensor(config-aaa-rad-sec)#
```



**Note** You must have the same secret value configured on both the RADIUS server and the IPS sensor so that the server can authenticate the requests of the client and the client can authenticate the responses of the server.

**Step 8** Specify the type of console authentication.

```
sensor(config-aaa-rad)# console-authentication radius-and-local
sensor(config-aaa-rad)#
```

You can choose local, local and RADIUS, or RADIUS.

**Step 9** Verify the settings:

```
sensor(config-aaa-rad)# show settings
radius

primary-server

server-address: 10.1.2.3
server-port: 1812 <defaulted>
shared-secret: kkkk
timeout: 3 <defaulted>

secondary-server
```

```

enabled

server-address: 10.4.5.6
server-port: 1816 default: 1812
shared-secret: yyyy
timeout: 8 default: 3

nas-id: cisco-ips default: cisco-ips
local-fallback: enabled default: enabled
console-authentication: radius-and-local <defaulted>
default-user-role: operator default: unspecified

sensor(config-aaa-rad)#

```

**Step 10** Exit AAA mode.

```

sensor(config-aaa-rad)# exit
sensor(config-aaa)# exit
Apply Changes?[yes]:

```

**Step 11** Press **Enter** to apply the changes or enter **no** to discard them.

#### For More Information

- For the procedure for adding and removing users, see [Adding and Removing Users, page 4-18](#).
- For the procedure for configuring passwords, see [Configuring Passwords, page 4-30](#).
- For the procedure for specifying password requirements, see [Configuring the Password Policy, page 4-32](#).
- For detailed information on RADIUS and the service account, see [The Service Account and RADIUS Authentication, page 4-29](#).

## Configuring Packet Command Restriction

Use the **permit-packet-logging** command to restrict the use of packet capture-related commands—packet capture/display and IP logging—for local and AAA RADIUS users. The default is to permit packet capture/display and IP log commands. Local users with the correct permissions can use the packet capture/display and IP log commands. AAA RADIUS users with the correct av-pair can use the packet capture/display and IP log commands. This command is supported in IPS 7.1(3)E4 and later.



#### Note

IP log actions configured for signatures are not impacted by the packet command restriction feature.

When you modify the packet command restriction option, you receive the following warning:

```

Modified packet settings would take effect only for new sessions, existing sessions will
continue with previous settings.

```

The following commands apply:

- **permit-packet-logging true**—Allows users to execute packet-related commands based on privilege level.
- **permit-packet-logging false**—Restricts all users from executing any packet-related commands.



**AAA RADIUS Users**

AAA RADIUS users with the correct av-pair are authorized to execute packet capture/display and IP logging commands. RADIUS users with no av-pair value are restricted. The correct av-pair, **permit-packet-logging=true**, allows users to execute packet-related commands based on privilege level. This av-pair is in addition to the authentication role related av-pair:

- **ips-role=viewer**
- **ips-role=operator**
- **ips-role=administrator**
- **ips-role=service**

**Status Events**

As part of the packet command restriction option, status events are triggered for the following actions:

- When an administrator enables or disables the packet command restriction.
- When an authorized user executes any of the restricted commands.
- When an unauthorized user executes any of the restricted commands.

**Configuring Packet Command Restrictions**

To permit or restrict packet command restrictions, follow these steps:

---

**Step 1** Log in to the sensor using an account with administrator privileges.

**Step 2** Enter authentication submode.

```
sensor# configure terminal
sensor(config)# service authentication
sensor(config-aut)#
```

**Step 3** Allow AAA RADIUS users with the correct av-pair (**permit-packet-logging=true**) and local users with the correct privilege levels to execute all packet capture/display and IP log commands.

```
sensor(config-aut)# permit-packet-logging true
```




---

**Note** Existing CLI sessions are not affected by the changes made in restriction settings.

---

**Step 4** Check your new setting.

```
sensor(config-aut)# show settings
attemptLimit: 0 <defaulted>
password-strength

size: 8-64 <defaulted>
digits-min: 0 <defaulted>
uppercase-min: 0 <defaulted>
lowercase-min: 0 <defaulted>
other-min: 0 <defaulted>
number-old-passwords: 0 <defaulted>

permit-packet-logging: true default: true
cli-inactivity-timeout: 0 <defaulted>
sensor(config-aut)#
```

**Step 5** Restrict all users from executing packet capture/display and IP log commands.

```
sensor(config-aut)# permit-packet-logging false
```

**Step 6** Check your new setting.

```

sensor(config-aut)# show settings
 attemptLimit: 0 <defaulted>
 password-strength

 size: 8-64 <defaulted>
 digits-min: 0 <defaulted>
 uppercase-min: 0 <defaulted>
 lowercase-min: 0 <defaulted>
 other-min: 0 <defaulted>
 number-old-passwords: 0 <defaulted>

 permit-packet-logging: false default: true
 cli-inactivity-timeout: 0 <defaulted>
sensor(config-aut)#

```

**Step 7** Exit authentication mode.

```

sensor(config-aut)# exit
Apply Changes:[yes]:

```

**Step 8** Press **Enter** to apply the changes or enter **no** to discard them.

## Creating the Service Account

You can create a service account for TAC to use during troubleshooting. Although more than one user can have access to the sensor, only one user can have service privileges on a sensor. The service account is for support purposes only.

The root user password is synchronized to the service account password when the service account is created. To gain root access you must log in with the service account and switch to user root with the **su - root** command.

**Caution**

Do not make modifications to the sensor through the service account except under the direction of TAC. If you use the service account to configure the sensor, your configuration is not supported by TAC. Adding services to the operating system through the service account affects proper performance and functioning of the other IPS services. TAC does not support a sensor on which additional services have been added.

**Caution**

You should carefully consider whether you want to create a service account. The service account provides shell access to the system, which makes the system vulnerable. However, you can use the service account to create a password if the administrator password is lost. Analyze your situation to decide if you want a service account existing on the system.

**Note**

For IPS 5.0 and later, you can no longer remove the **cisco** account. You can disable it using the **no password cisco** command, but you cannot remove it. To use the **no password cisco** command, there must be another administrator account on the sensor. Removing the **cisco** account through the service account is not supported. If you remove the **cisco** account through the service account, the sensor most likely will not boot up, so to recover the sensor you must reinstall the sensor system image.

**Creating the Service Account**

To create the service account, follow these steps:

- 
- Step 1** Log in to the CLI using an account with administrator privileges.
- Step 2** Enter configuration mode.
- ```
sensor# configure terminal
```
- Step 3** Specify the parameters for the service account. The username follows the pattern `^[A-Za-z0-9()+;.,_-]+`, which means the username must start with a letter or number, and can include any letter A to Z (capital or small), any number 0 to 9, - and _, and can contain 1 to 64 characters.
- ```
sensor(config)# user username privilege service
```
- Step 4** Specify a password when prompted. A valid password is 8 to 32 characters long. All characters except space are allowed. If a service account already exists for this sensor, the following error is displayed and no service account is created.
- ```
Error: Only one service account may exist
```
- Step 5** Exit configuration mode.
- ```
sensor(config)# exit
sensor#
```

When you use the service account to log in to the CLI, you receive this warning.

```
***** WARNING *****
UNAUTHORIZED ACCESS TO THIS NETWORK DEVICE IS PROHIBITED. This account is intended to be
used for support and troubleshooting purposes only. Unauthorized modifications are not
supported and will require this device to be reimaged to guarantee proper operation.

```

## The Service Account and RADIUS Authentication

If you are using RADIUS authentication and want to create and use a service account, you must create the service account both on your sensor and on the RADIUS server. You must use local authentication to access the service account on the sensor. The service account must be created manually as a local account on the sensor. Then when you configure RADIUS authentication, the service account must also be configured manually on the RADIUS server with the accept message set to `ip-role=service`.

When you log in to the service account, you are authenticated against both the sensor account and the RADIUS server account. By whatever method you use to access the service account—serial console port, direct monitor/keyboard (for sensors that support it), or a network connection, such as SSH or Telnet—you have to log in using local authentication.

## RADIUS Authentication Functionality and Limitations

The current AAA RADIUS implementation has the following functionality and limitations:

- Authentication with a RADIUS server—However, you cannot change the password of the RADIUS server from the IPS.
- Authorization—You can perform role-based authorization by specifying the IPS role of the user on the RADIUS server.
- Accounting—The login attempts of the user and the configuration changes are logged as events locally on the IPS. However, these account messages are not communicated to the RADIUS server.

## Configuring Passwords

Use the **password** command to update the password on the local sensor. You can also use this command to change the password for an existing user or to reset the password for a locked account. A valid password is 6 to 127 characters long. All characters except space are allowed.

To change the password, follow these steps:

---

**Step 1** To change the password for another user or reset the password for a locked account, follow these steps:

- Log in to the CLI using an account with administrator privileges.
- Enter configuration mode.

```
sensor# configure terminal
```

- Change the password for a specific user. This example modifies the password for the user “tester.”

```
sensor(config)# password tester
Enter New Login Password: *****
Re-enter New Login Password: *****
```

**Step 2** To change your password, follow these steps:

- Log in to the CLI.
- Enter configuration mode.

```
sensor# configure terminal
```

- Change your password.

```
sensor(config)# password
Enter Old Login Password: *****
Enter New Login Password: *****
Re-enter New Login Password: *****
```

---

### For More Information

For the procedures for recovering sensor passwords, see [Recovering the Password, page 17-2](#).

## Changing User Privilege Levels



### Note

You cannot use the **privilege** command to give a user service privileges. If you want to give an existing user service privileges, you must remove that user and then use the **username** command to create the service account. There can only be one person with service privileges.

Use the **privilege** command to change the privilege level—administrator, operator, viewer—for a user. To change the privilege level for a user, follow these steps:

**Step 1** Log in to the CLI using an account with administrator privileges.

**Step 2** Verify the current privilege of the user *jsmith*.

```
sensor# show users all
 CLI ID User Privilege
* 13491 cisco administrator
 jsmith viewer
 operator operator
 service service
 viewer viewer

sensor#
a
```

**Step 3** Change the privilege level from viewer to operator.

```
sensor# configure terminal
sensor(config)# privilege user jsmith operator
Warning: The privilege change does not apply to current CLI sessions. It will be applied
to subsequent logins.
sensor(config)#
```

**Step 4** Verify that the privilege of the user has been changed. The privilege of the user *jsmith* has been changed from viewer to operator.

```
sensor(config)# exit
sensor# show users all

 CLI ID User Privilege
* 13491 cisco administrator
 jsmith operator
 operator operator
 service service
 viewer viewer

sensor#
```

**Step 5** Display your current level of privilege.

```
sensor# show privilege
Current privilege level is administrator
```

### For More Information

For the procedure for creating the service account, see [Creating the Service Account](#), page 4-28.

## Showing User Status



### Note

All IPS platforms allow ten concurrent log in sessions.

Use the **show users** command to view information about the username and privilege of all users logged in to the sensor, and all user accounts on the sensor regardless of login status. An asterisk (\*) indicates the current user. If an account is locked, the username is surrounded by parentheses. A locked account means that the user failed to enter the correct password after the configured attempts.

To show user information, follow these steps:

**Step 1** Log in to the CLI using an account with administrator privileges.

**Step 2** Verify the users logged in to the sensor.

```
sensor# show users
 CLI ID User Privilege
* 13491 cisco administrator
sensor#
```

**Step 3** Verify all users. The account of the user `jsmith` is locked.

```
sensor# show users all
 CLI ID User Privilege
* 13491 cisco administrator
 5824 (jsmith) viewer
 9802 tester operator
sensor#
```

**Step 4** To unlock the account of `jsmith`, reset the password.

```
sensor# configure terminal
sensor(config)# password jsmith
Enter New Login Password: *****
Re-enter New Login Password: *****
```

## Configuring the Password Policy

As sensor administrator, you can configure how passwords are created. All user-created passwords must conform to the policy that you set up. You can set login attempts and the size and minimum characters requirements for a password. The minimum password length is eight characters. If you forget your password, there are various ways to recover the password depending on your sensor platform.



### Caution

If the password policy includes minimum numbers of character sets, such as upper case or number characters, the sum of the minimum number of required character sets cannot exceed the minimum password size. For example, you cannot set a minimum password size of eight and also require that passwords must contain at least five lowercase and five uppercase characters.

**Example**

For example, you can set a policy where passwords must have at least 10 characters and no more than 40, and must have a minimum of 2 upper case and 2 numeric characters. Once that policy is set, every password configured for each user account must conform to this password policy.

To set up a password policy, follow these steps:

- 
- Step 1** Log in to the sensor using an account with administrator privileges.
- Step 2** Enter password strength authentication submode.
- ```
sensor# configure terminal
sensor(config)# service authentication
sensor(config-aut)# password-strength
```
- Step 3** Set the minimum number of numeric digits that must be in a password. The range is 0 to 64.
- ```
sensor(config-aut-pas)# digits-min 6
```
- Step 4** Set the minimum number of nonalphanumeric printable characters that must be in a password. The range is 0 to 64.
- ```
sensor(config-aut-pas)# other-min 3
```
- Step 5** Set the minimum number of uppercase alphabet characters that must be in a password. The range is 0 to 64.
- ```
sensor(config-aut-pas)# uppercase-min 3
```
- Step 6** Set the minimum number of lower-case alphabet characters that must be in a password.
- ```
sensor(config-aut-pas)# lowercase-min 3
```
- Step 7** Set the number of old passwords to remember for each account. A new password cannot match any of the old passwords of an account.
- ```
sensor(config-aut-pas)# number-old-passwords 3
```
- Step 8** Check your new setting.
- ```
sensor(config-aut-pas)# show settings
password-strength
-----
size: 8-64 <defaulted>
digits-min: 6 default: 0
uppercase-min: 3 default: 0
lowercase-min: 3 default: 0
other-min: 3 default: 0
number-old-passwords: 3 default: 0
-----
sensor(config-aut-pas)#
```
-

For More Information

For the procedures for recovering sensor passwords, see [Recovering the Password, page 17-2](#).

Locking User Accounts


Note

When you configure account locking, local authentication, as well as RADIUS authentication, is affected. After a specified number of failed attempts to log in locally or in to a RADIUS account, the account is locked locally on the sensor. For local accounts, you can reset the password or use the **unlock user *username*** command to unlock the account. For RADIUS user accounts, you must use the **unlock user *username*** command to unlock the account.


Note

For RADIUS users, the attempt limit feature is enforced only after the RADIUS user's first successful login to the sensor.

Use the **attemptLimit *number*** command in authentication submode to lock accounts so that users cannot keep trying to log in after a certain number of failed attempts. The default is 0, which indicates unlimited authentication attempts. For security purposes, you should change this number.

To configure account locking, follow these steps:

Step 1 Log in to the sensor using an account with administrator privileges.

Step 2 Enter service authentication submode.

```
sensor# configure terminal
sensor(config)# service authentication
```

Step 3 Set the number of attempts users will have to log in to accounts.

```
sensor(config-aut)# attemptLimit 3
```

Step 4 Check your new setting.

```
sensor(config-aut)# show settings
  attemptLimit: 3 defaulted: 0
sensor(config-aut)#
```

Step 5 Set the value back to the system default setting.

```
sensor(config-aut)# default attemptLimit
```

Step 6 Check that the setting has returned to the default.

```
sensor(config-aut)# show settings
  attemptLimit: 0 <defaulted>
sensor(config-aut)#
```

Step 7 Check to see if any users have locked accounts. The account of the user `jsmith` is locked as indicated by the parentheses.


Note

When you apply a configuration that contains a non-zero value for `attemptLimit`, a change is made in the SSH server that may subsequently impact your ability to connect with the sensor. When `attemptLimit` is non-zero, the SSH server requires the client to support challenge-response authentication. If you experience problems after your SSH client connects but before it prompts for a password, you need to enable challenge-response authentication. Refer to the documentation for your SSH client for instructions.


```

sensor(config-aut)# exit
sensor(config)# exit
sensor# show users all
  CLI ID   User      Privilege
*   1349   cisco     administrator
    5824   (jsmith)  viewer
    9802   tester    operator

```

Step 8 To unlock the account of jsmith, reset the password.

```

sensor# configure terminal
sensor(config)# password jsmith
Enter New Login Password: *****
Re-enter New Login Password: *****

```

For More Information

For the procedure for unlocking user accounts, see [Unlocking User Accounts, page 4-35](#).

Unlocking User Accounts

Use the **unlock user *username*** command in global configuration mode to unlock local and RADIUS accounts for users who have been locked out after a specified number of failed attempts.

To configure account unlocking, follow these steps:

Step 1 Log in to the sensor using an account with administrator privileges.

Step 2 Check to see if any users have locked accounts. The account of the user jsmith is locked as indicated by the parentheses.

```

sensor# show users all
  CLI ID   User      Privilege
*   1349   cisco     administrator
    5824   (jsmith)  viewer
    9802   tester    operator

```

Step 3 Enter global configuration mode.

```

sensor# configure terminal
sensor(config)#

```

Step 4 Unlock the account.

```

sensor(config)# unlock user jsmith

```

Step 5 Check your new setting. The account of the user jsmith is now unlocked as indicated by the lack of parenthesis.

```

sensor# show users all
  CLI ID   User      Privilege
*   1349   cisco     administrator
    5824   jsmith    viewer
    9802   tester    operator

```

For More Information

For the procedure for locking the user accounts, see [Locking User Accounts, page 4-34](#).

Configuring Time

This section describes the importance of having a reliable time source for the sensor. It contains the following topics:

- [Time Sources and the Sensor, page 4-36](#)
- [Synchronizing IPS Module System Clocks with the Parent Device System Clock, page 4-37](#)
- [Correcting Time on the Sensor, page 4-37](#)
- [Configuring Time on the Sensor, page 4-37](#)
- [Configuring NTP, page 4-43](#)

Time Sources and the Sensor

**Note**

We recommend that you use an NTP server to regulate time on your sensor. You can use authenticated or unauthenticated NTP. For authenticated NTP, you must obtain the NTP server IP address, NTP server key ID, and the key value from the NTP server. You can set up NTP during initialization or you can configure NTP through the CLI, IDM, IME, or ASDM.

The sensor requires a reliable time source. All events (alerts) must have the correct UTC and local time stamp, otherwise, you cannot correctly analyze the logs after an attack. When you initialize the sensor, you set up the time zones and summertime settings. This section provides a summary of the various ways to set the time on sensors.

The IPS Standalone Appliances

- Use the **clock set** command to set the time. This is the default.
- Configure the appliance to get its time from an NTP time synchronization source.

**Note**

The currently supported Cisco IPS appliances are the IPS 4345, IPS 4360, IPS 4510, IPS 4520, and IPS 4520-XL.

The ASA IPS Modules

- The ASA 5500-X IPS SSP and ASA 5585-X IPS SSP automatically synchronize their clocks with the clock in the adaptive security appliance in which they are installed. This is the default.
- Configure them to get their time from an NTP time synchronization source, such as a Cisco router other than the parent router.

Synchronizing IPS Module System Clocks with the Parent Device System Clock

The ASA IPS modules (ASA 5500-X IPS SSP and ASA 5585-X IPS SSP) synchronize their clocks to the parent chassis clock (adaptive security appliance) each time the IPS boots up and any time the parent chassis clock is set. The IPS clock and parent chassis clock tend to drift apart over time. The difference can be as much as several seconds per day. To avoid this problem, make sure that both the IPS clock and the parent clock are synchronized to an external NTP server. If only the IPS clock or only the parent chassis clock is synchronized to an NTP server, the time drift occurs.

Correcting Time on the Sensor

If you set the time incorrectly, your stored events will have the incorrect time because they are stamped with the time the event was created. The Event Store time stamp is always based on UTC time. If during the original sensor setup, you set the time incorrectly by specifying 8:00 p.m. rather than 8:00 a.m., when you do correct the error, the corrected time will be set backwards. New events might have times older than old events.

For example, if during the initial setup, you configure the sensor as central time with daylight saving time enabled and the local time is 8:04 p.m., the time is displayed as 20:04:37 CDT and has an offset from UTC of -5 hours (01:04:37 UTC, the next day). A week later at 9:00 a.m., you discover the error: the clock shows 21:00:23 CDT. You then change the time to 9:00 a.m. and now the clock shows 09:01:33 CDT. Because the offset from UTC has not changed, it requires that the UTC time now be 14:01:33 UTC, which creates the time stamp problem.

To ensure the integrity of the time stamp on the event records, you must clear the event archive of the older events by using the **clear events** command.

**Note**

You cannot remove individual events.

Configuring Time on the Sensor

This section describes how to configure time on the sensor so that your events are time-stamped correctly. It contains the following topics:

- [Displaying the System Clock, page 4-37](#)
- [Manually Setting the System Clock, page 4-38](#)
- [Configuring Recurring Summertime Settings, page 4-39](#)
- [Configuring Nonrecurring Summertime Settings, page 4-41](#)
- [Configuring Time Zones Settings, page 4-43](#)

Displaying the System Clock

Use the **show clock [detail]** command to display the system clock. You can use the **detail** option to indicate the clock source (NTP or system) and the current summertime setting (if any). The system clock keeps an authoritative flag that indicates whether the time is authoritative (believed to be accurate). If the system clock has been set by a timing source, such as NTP, the flag is set.

Table 4-1 lists the system clock flags.

Table 4-1 System Clock Flags

Symbol	Description
*	Time is not authoritative.
(blank)	Time is authoritative.
.	Time is authoritative, but NTP is not synchronized.

To display the system clock, follow these steps:

Step 1 Log in to the CLI.

Step 2 Display the system clock.

```
sensor# show clock
*19:04:52 UTC Thu Apr 03 2008
```

Step 3 Display the system clock with details. The following example indicates that the sensor is getting its time from NTP and that is configured and synchronized.

```
sensor# show clock detail
20:09:43 UTC Thu Apr 03 2011
Time source is NTP
Summer time starts 03:00:00 UTC Sun Mar 09 2011
Summer time stops 01:00:00 UTC Sun Nov 02 2011
```

Step 4 Display the system clock with details. The following example indicates that no time source is configured.

```
sensor# show clock detail
*20:09:43 UTC Thu Apr 03 2011
No time source
Summer time starts 03:00:00 UTC Sun Mar 09 2011
Summer time stops 01:00:00 UTC Sun Nov 02 2011
```

Manually Setting the System Clock



Note

You do not need to set the system clock if your sensor is synchronized by a valid outside timing mechanism such as an NTP clock source.

Use the **clock set *hh:mm [:ss] month day year*** command to manually set the clock on the appliance. Use this command if no other time sources are available. The **clock set** command does not apply to the following platforms, because they get their time from the adaptive security appliance in which they are installed:

- ASA 5500-X IPS SSP
- ASA 5585-X IPS SSP

To manually set the clock on the appliance, follow these steps:

Step 1 Log in to the CLI using an account with administrator privileges.

Step 2 Set the clock manually.

```
sensor# clock set 13:21 Mar 29 2011
```



Note The time format is 24-hour time.

Configuring Recurring Summertime Settings



Note Summertime is a term for daylight saving time.

Use the **summertime-option recurring** command to configure the sensor to switch to summertime settings on a recurring basis. The default is recurring.

To configure the sensor to switch to summertime settings on a recurring basis, follow these steps:

Step 1 Log in to the sensor using an account with administrator privileges.

Step 2 Enter summertime recurring submode.

```
sensor# configure terminal
sensor(config)# service host
sensor(config-hos)# summertime-option recurring
```

Step 3 Enter start summertime submode.

```
sensor(config-hos-rec)# start-summertime
```

Step 4 Configure the start summertime parameters:

- a. Enter the day of the week you want to start summertime settings.

```
sensor(config-hos-rec-sta)# day-of-week monday
```

- b. Enter the month you want to start summertime settings.

```
sensor(config-hos-rec-sta)# month april
```

- c. Enter the time of day you want to start summertime settings. The format is hh:mm:ss.

```
sensor(config-hos-rec-sta)# time-of-day 12:00:00
```

- d. Enter the week of the month you want to start summertime settings. The values are first through fifth, or last.

```
sensor(config-hos-rec-sta)# week-of-month first
```

- e. Verify your settings.

```
sensor(config-hos-rec-sta)# show settings
start-summertime
-----
month: april default: april
```

```

week-of-month: first default: first
day-of-week: monday default: sunday
time-of-day: 12:00:00 default: 02:00:00
-----
sensor(config-hos-rec-sta)#

```

Step 5 Enter end summertime submode.

```

sensor(config-hos-rec-sta)# exit
sensor(config-hos-rec)# end-summertime

```

Step 6 Configure the end summertime parameters:

a. Enter the day of the week you want to end summertime settings.

```

sensor(config-hos-rec-end)# day-of-week friday

```

b. Enter the month you want to end summertime settings.

```

sensor(config-hos-rec-end)# month october

```

c. Enter the time of day you want to end summertime settings. The format is hh:mm:ss.

```

sensor(config-hos-rec-end)# time-of-day 05:15:00

```

d. Enter the week of the month you want to end summertime settings. The values are first through fifth, or last.

```

sensor(config-hos-rec-end)# week-of-month last

```

e. Verify your settings.

```

sensor(config-hos-rec-end)# show settings
end-summertime
-----
month: october default: october
week-of-month: last default: last
day-of-week: friday default: sunday
time-of-day: 05:15:00 default: 02:00:00
-----
sensor(config-hos-rec-end)#

```

Step 7 Specify the local time zone used during summertime.

```

sensor(config-hos-rec-end)# exit
sensor(config-hos-rec)# summertime-zone-name CDT

```

Step 8 Specify the offset.

```

sensor(config-hos-rec)# offset 60

```

Step 9 Verify your settings.

```

sensor(config-hos-rec)# show settings
recurring
-----
offset: 60 minutes default: 60
summertime-zone-name: CDT
start-summertime
-----
month: april default: april
week-of-month: first default: first
day-of-week: monday default: sunday
time-of-day: 12:00:00 default: 02:00:00
-----
end-summertime

```

```

-----
month: october default: october
week-of-month: last default: last
day-of-week: friday default: sunday
time-of-day: 05:15:00 default: 02:00:00
-----

```

Step 10 Exit recurring summertime submode.

```

sensor(config-hos-rec)# exit
sensor(config-hos)# exit
Apply Changes:[yes]:

```

Step 11 Press **Enter** to apply the changes or enter **no** to discard them.

Configuring Nonrecurring Summertime Settings



Note

Summertime is a term for daylight saving time.

Use the **summertime-option non-recurring** command to configure the sensor to switch to summer time settings on a one-time basis. The default is recurring.

To configure the sensor to switch to summertime settings on a one-time basis, follow these steps:

Step 1 Log in to the sensor using an account with administrator privileges.

Step 2 Enter summertime non-recurring submode.

```

sensor# configure terminal
sensor(config)# service host
sensor(config-hos)# summertime-option non-recurring

```

Step 3 Enter start summertime submode.

```

sensor(config-hos-non)# start-summertime

```

Step 4 Configure the start summertime parameters:

- a. Enter the date you want to start summertime settings. The format is yyyy-mm-dd.

```

sensor(config-hos-non-sta)# date 2004-05-15

```

- b. Enter the time you want to start summertime settings. The format is hh:mm:ss.

```

sensor(config-hos-non-sta)# time 12:00:00

```

- c. Verify your settings.

```

sensor(config-hos-non-sta)# show settings
start-summertime

```

```

-----
date: 2004-05-15
time: 12:00:00
-----

```

```

sensor(config-hos-non-sta)#

```

Step 5 Enter end summertime submode.

```

sensor(config-hos-non-sta)# exit

```

```
sensor(config-hos-non)# end-summertime
```

Step 6 Configure the end summertime parameters:

- a. Enter the date you want to end summertime settings. The format is yyyy-mm-dd.

```
sensor(config-hos-non-end)# date 2004-10-31
```

- b. Enter the time you want to end summertime settings. The format is hh:mm:ss.

```
sensor(config-hos-non-end)# time 12:00:00
```

- c. Verify your settings.

```
sensor(config-hos-non-end)# show settings
end-summertime
```

```
-----
date: 2004-10-31
time: 12:00:00
-----
```

```
sensor(config-hos-non-end)#
```

Step 7 Specify the local time zone used during summertime.

```
sensor(config-hos-non-end)# exit
sensor(config-hos-non)# summertime-zone-name CDT
```

Step 8 Specify the offset.

```
sensor(config-hos-non)# offset 60
```

Step 9 Verify your settings.

```
sensor(config-hos-non)# show settings
non-recurring
```

```
-----
offset: 60 minutes default: 60
summertime-zone-name: CDT
start-summertime
```

```
-----
date: 2004-05-15
time: 12:00:00
-----
```

```
end-summertime
```

```
-----
date: 2004-10-31
time: 12:00:00
-----
```

```
sensor(config-hos-non)#
```

Step 10 Exit non-recurring summertime submode.

```
sensor(config-hos-non)# exit
sensor(config-hos)# exit
Apply Changes?[yes]:
```

Step 11 Press **Enter** to apply the changes or enter **no** to discard them.

Configuring Time Zones Settings

Use the **time-zone-settings** command to configure the time zone settings on the sensor, such as the time zone name the sensor displays whenever summertime settings are not in effect and the offset.

To configure the time zone settings on the sensor, follow these steps:

-
- Step 1** Log in to the sensor using an account with administrator privileges.
- Step 2** Enter time zone settings submode.
- ```
sensor# configure terminal
sensor(config)# service host
sensor(config-hos)# time-zone-settings
```
- Step 3** Configure the time zone name that is displayed whenever summertime settings are not in effect. The default is UTC.
- ```
sensor(config-hos-tim)# standard-time-zone-name CST
```
- Step 4** Configure the offset in minutes. The offset is the number of minutes you add to UTC to get the local time. The default is 0.
- ```
sensor(config-hos-tim)# offset -360
```
- Step 5** Verify your settings.
- ```
sensor(config-hos-tim)# show settings
time-zone-settings
-----
offset: -360 minutes default: 0
standard-time-zone-name: CST default: UTC
-----
sensor(config-hos-tim)#
```
- Step 6** Exit time zone settings submode.
- ```
sensor(config-hos-tim)# exit
sensor(config-hos)# exit
Apply Changes?[yes]:
```
- Step 7** Press **Enter** to apply the changes or enter **no** to discard them.
- 

## Configuring NTP

This section describes how to configure a Cisco router to be an NTP server and how to configure the sensor to use an NTP server as its time source. It contains the following topics:

- [Configuring a Cisco Router to be an NTP Server, page 4-43](#)
- [Configuring the Sensor to Use an NTP Time Source, page 4-45](#)

## Configuring a Cisco Router to be an NTP Server

The sensor requires an authenticated connection with an NTP server if it is going to use the NTP server as its time source. The sensor supports only the MD5 hash algorithm for key encryption. Use the following procedure to activate a Cisco router to act as an NTP server and use its internal clock as the time source.

**Caution**

The sensor NTP capability is designed to be compatible with Cisco routers acting as NTP servers. The sensor may work with other NTP servers, but is not tested or supported.

**Note**

Remember the NTP server key ID and key values. You need them along with the NTP server IP address when you configure the sensor to use the NTP server as its time source.

To set up a Cisco router to act as an NTP server, follow these steps:

**Step 1** Log in to the router.

**Step 2** Enter configuration mode.

```
router# configure terminal
```

**Step 3** Create the key ID and key value. The key ID can be a number between 1 and 65535. The key value is text (numeric or character). It is encrypted later.

```
router(config)# ntp authentication-key key_ID md5 key_value
```

Example

```
router(config)# ntp authentication-key 100 md5 attack
```

**Note**

The sensor only supports MD5 keys.

**Note**

Keys may already exist on the router. Use the **show running configuration** command to check for other keys. You can use those values for the trusted key in Step 4.

**Step 4** Designate the key you just created in Step 3 as the trusted key (or use an existing key). The trusted key ID is the same number as the key ID in Step 3.

```
router(config)# ntp trusted-key key_ID
```

Example

```
router(config)# ntp trusted-key 100
```

**Step 5** Specify the interface on the router with which the sensor will communicate.

```
router(config)# ntp source interface_name
```

Example

```
router(config)# ntp source FastEthernet 1/0
```

**Step 6** Specify the NTP master stratum number to be assigned to the sensor. The NTP master stratum number identifies the relative position of the server in the NTP hierarchy. You can choose a number between 1 and 15. It is not important to the sensor which number you choose.

```
router(config)# ntp master stratum_number
```

## Example

```
router(config)# ntp master 6
```

## Configuring the Sensor to Use an NTP Time Source

The sensor requires a consistent time source. We recommend that you use an NTP server. Use the following procedure to configure the sensor to use the NTP server as its time source. You can use authenticated or unauthenticated NTP.

**Note**

For authenticated NTP, you must obtain the NTP server IP address, NTP server key ID, and the key value from the NTP server.

**Caution**

The sensor NTP capability is designed to be compatible with Cisco routers acting as NTP servers. The sensor may work with other NTP servers, but is not tested or supported.

To configure the sensor to use an NTP server as its time source, follow these steps:

- 
- Step 1** Log in to the CLI using an account with administrator privileges.
- Step 2** Enter configuration mode.
- ```
sensor# configure terminal
```
- Step 3** Enter service host mode.
- ```
sensor(config)# service host
```
- Step 4** Configure unauthenticated NTP:
- a. Enter NTP configuration mode.
 

```
sensor(config-hos)# ntp-option enabled-ntp-unauthenticated
```
  - b. Specify the NTP server IP address.
 

```
sensor(config-hos-ena)# ntp-server ip_address
```
  - c. Verify the unauthenticated NTP settings.
 

```
sensor(config-hos-ena)# show settings
enabled-ntp-unauthenticated

ntp-server: 10.89.147.45

sensor(config-hos-ena)#
```
- Step 5** Configure authenticated NTP:
- a. Enter NTP configuration mode.
 

```
sensor(config-hos)# ntp-option enable
```
  - b. Specify the NTP server IP address and key ID. The key ID is a number between 1 and 65535. This is the key ID that you already set up on the NTP server.
 

```
sensor(config-hos-ena)# ntp-servers ip_address key-id key_ID
```

## Example

```
sensor(config-hos-ena)# ntp-servers 10.16.0.0 key-id 100
```

- c. Specify the key value NTP server. The key value is text (numeric or character). This is the key value that you already set up on the NTP server.

```
sensor(config-hos-ena)# ntp-keys key_ID md5-key key_value
```

## Example

```
sensor(config-hos-ena)# ntp-keys 100 md5-key attack
```

- d. Verify the NTP settings.

```
sensor(config-hos-ena)# show settings
enabled

ntp-keys (min: 1, max: 1, current: 1)

key-id: 100

md5-key: attack

ntp-servers (min: 1, max: 1, current: 1)

ip-address: 10.16.0.0
key-id: 100

sensor(config-hos-ena)#
```

- Step 6** Exit NTP configuration mode.

```
sensor(config-hos-ena)# exit
sensor(config-hos)# exit
Apply Changes?[yes]
```

- Step 7** Press **Enter** to apply the changes or enter **no** to discard them.

## Configuring SSH

This section describes SSH on the sensor, and contains the following topics:

- [Understanding SSH, page 4-47](#)
- [Adding Hosts to the SSH Known Hosts List, page 4-47](#)
- [Adding Authorized Keys, page 4-49](#)
- [Generating the RSA Server Host Key, page 4-50](#)

## Understanding SSH

SSH provides strong authentication and secure communications over channels that are not secure. SSH encrypts your connection to the sensor and provides a key so you can validate that you are connecting to the correct sensor. SSH also provides authenticated and encrypted access to other devices that the sensor connects to for blocking. The IPS supports a management through both SSHv1 and SSHv2.

SSH authenticates the hosts or networks using one or both of the following:

- Password
- User RSA public key



---

**Note** SSH never sends passwords in clear text.

---

SSH protects against the following:

- IP spoofing—A remote host sends out packets pretending to come from another trusted host.



---

**Note** SSH even protects against a spoofer on the local network who can pretend he is your router to the outside.

---

- IP source routing—A host pretends an IP packet comes from another trusted host.
- DNS spoofing—An attacker forges name server records.
- Interception of clear text passwords and other data by intermediate hosts.
- Manipulation of data by those in control of intermediate hosts.
- Attacks based on listening to X authentication data and spoofed connection to the X11 server.

## Adding Hosts to the SSH Known Hosts List

You must add hosts to the SSH known hosts list so that the sensor can recognize the hosts that it can communicate with through SSH. These hosts are SSH servers that the sensor needs to connect to for upgrades and file copying, and other hosts, such as Cisco routers, firewalls, and switches that the sensor will connect to for blocking.

For SSHv1, use the **ssh host-key ip-address [key-modulus-length public-exponent public-modulus]** command to add an entry to the known hosts list. If you do not know the values for the modulus, exponent, and length, the system displays the bubble babble for the requested IP address. You can then choose to add the key to the list. To modify a key for an IP address, the entry must be removed and recreated. Use the **no** form of the command to remove the entry.



### Caution

---

When you use the **ssh host-key ip-address** command, the SSH server at the specified IP address is contacted to obtain the required key over the network. The specified host must be accessible at the moment the command is issued. If the host is unreachable, you must use the full form of the command, **ssh host-key ip-address [key-modulus-length public-exponent public-modulus]**, to confirm the fingerprint of the key displayed to protect yourself from accepting the key of an attacker.

---

For SSHv2, use the **ssh host-key ip-address [public-key]** command to add an entry to the known hosts list.

The following commands apply:

- *ip address*—Specifies the IP address to add to the system.
- **rsa-key**—Specifies the RSA (SSHv2) key details.
  - **key**—Specifies the Base64 encoded public key.
- **rsa1-key**—Specifies the RSA1 (SSHv1) key details:
  - *key-modulus-length*—Specifies an ASCII decimal integer in the range[511, 2048].
  - *public-exponent*—Specifies an ASCII decimal integer in the range [3, 2<sup>32</sup>].
  - *public-modulus*—Specifies an ASCII decimal integer, *x*, such that (2<sup>(key-modulus-length-1)</sup> < *x* < (2<sup>(key-modulus-length)</sup>)).

### Adding a Host to the SSH Known Hosts List

To add a host to the SSH known hosts list, follow these steps:

---

**Step 1** Log in to the CLI using an account with administrator or operator privileges.

**Step 2** Enter configuration mode.

```
sensor# configure terminal
```

**Step 3** Add an entry to the known hosts list.

```
sensor(config)# ssh host-key 10.16.0.0
Bubble Babble is xucis-hehon-kizog-nedeg-zunom-kolyn-syzec-zasyk-symuf-rykum-sexyx
Would you like to add this to the known hosts table for this host?[yes]
```

The Bubble Babble appears. You are prompted to add it to the known hosts list.

If the host is not accessible when the command is issued, this message appears.

```
Error: getHostSshKey : socket connect failed [4,111]
```

**Step 4** Enter **yes** to have the fingerprint added to the known hosts list.

**Step 5** Verify that the host was added.

```
sensor(config)# exit
sensor# show ssh host-keys
10.89.146.110
```

**Step 6** View the key for a specific IP address.

```
sensor# show ssh host-keys 10.16.0.0
1024 35
139306213541835240385332922253968814685684523520064131997839905113640120217816869696708721
704631322844292073851730565044879082670677554157937058485203995572114631296604552161309712
60106861481274996959351374059833139315488498830230218292235335152653860589163651944997842
874583627883277460138506084043415861927
Bubble Babble: xebiz-vykyk-fekuh-rukuk-cabaz-paret-gosym-serum-korus-fypop-huxyx
sensor#
```

**Step 7** Remove an entry. The host is removed from the SSH known hosts list.

```
sensor(config)# no ssh host-key 10.16.0.0
```

**Step 8** Verify the host was removed. The IP address no longer appears in the list.

```
sensor(config)# exit
sensor# show ssh host-keys
```

## Adding Authorized Keys

Use the **ssh authorized-key** command to define public keys for a client allowed to use RSA1 or RSA2 authentication to log in to the local SSH server. The default is RSA2. You can configure the sensor to fall back to RSA1. To modify an authorized key, you must remove and recreate the entry. Use the **no** form of the command to remove the entry. Users can only create and remove their own keys.

The following commands apply:

- *id*—Specifies a 1 to 256-character string that uniquely identifies the authorized key. You can use numbers, “\_,” and “-,” but spaces and “?” are not acceptable.
  - **rsa-pubkey**—Specifies the RSA (SSHv2) key details.
  - *key*—Specifies the Base64 encoded public key.
- **rsa1-pubkey**—Specifies the RSA1 (SSHv1) key details:
  - *key-modulus-length*—Specifies an ASCII decimal integer in the range[511, 2048].
  - *public-exponent*—Specifies an ASCII decimal integer in the range [3, 2^32].
  - *public-modulus*—Specifies an ASCII decimal integer, *x*, such that  $(2^{(key-modulus-length-1)} < x < (2^{(key-modulus-length)}))$ .

Each user who can log in to the sensor has a list of authorized public keys. An SSH client with access to any of the corresponding RSA private keys can log in to the sensor as the user without entering a password.

For SSHv1, use an RSA key generation tool on the client where the private key is going to reside. Then, display the generated public key as a set of three numbers (modulus length, public exponent, public modulus) and enter those numbers as parameters for the **ssh authorized-key** command. For SSHv2, you just need the ID and the public key.



### Note

You configure your own list of SSH authorized keys. An administrator cannot manage the list of SSH authorized keys for other users on the sensor.



### Note

An SSH authorized key provides better security than passwords if the private key is adequately safeguarded. The best practice is to create the private key on the same host where it will be used and store it with a pass phrase on a local file system. To minimize password or pass phrase prompts, use a key agent.

### Adding a Key Entry

To add a key entry to the SSHv1 or SSHv2 authorized keys list for the current user, follow these steps:

**Step 1** Log in to the CLI.

**Step 2** Add a key to the authorized keys list for the current user.



**Note** You receive an error message if you try to add a key less than the 2048-bit key size and if the measured key length and input key length do not match.

For SSHv1:

```
sensor# configure terminal
sensor(config)# ssh authorized-key mhs rsa1-pubkey 512 34 877777777777
sensor(config)#
```

For SSHv2:

```
sensor# configure terminal
sensor(config)# ssh authorized-key phs rsa-pubkey AAAAAAAslkfjslkfjsjfs
sensor(config)#
```

**Step 3** Enter **yes** to add the key to the authorized key list.

**Step 4** Verify that the key was added.

```
sensor(config)# exit
sensor# show ssh authorized-keys
mhs
phs
sensor#
```

**Step 5** View the key for a specific ID.

```
sensor# show ssh authorized-keys mhs
512 34 877777777777
sensor#
```

**Step 6** Remove an entry from the list of SSH authorized keys.

```
sensor# configure terminal
sensor(config)# no ssh authorized-key mhs rsa1-pubkey
```

**Step 7** Verify the entry was removed.

```
sensor(config)# exit
sensor# show ssh authorized-keys
```

**Step 8** If you enter the former ID, you receive an error message.

```
sensor# show ssh authorized-keys mhs
Error: Requested id does not exist for the current user.
sensor#
```

## Generating the RSA Server Host Key

The server uses the SSHv1 or SSHv2 host key to prove its identity. Clients know they have contacted the correct server when they see a known key. The sensor generates an SSHv1 or SSHv2 host key the first time it starts up.

Use the **ssh generate-key** command to change the SSH server host key. The displayed fingerprint matches the one displayed in the remote SSH client in future connections with this sensor if the remote client is using SSH.



**Note**

The sensor only supports RSA keys. Peers that communicate with IPS need to support RSA keys; otherwise, the connection is not established.

To generate a new SSH server host key, follow these steps:

**Step 1** Log in to the CLI using an account with administrator privileges.

**Step 2** Generate the new server host key.

```
sensor# ssh generate-key
RSA1 Bubble Babble: xucor-gidyg-comym-zipib-pilyk-vucal-pekyd-hipuc-tuven-gigyr-fixyx
RSA Bubble Babble: xucot-sapaf-sufiz-duriv-rigud-kezol-tupif-buvih-zokap-sohoz-kixox
sensor#
```

**Caution**

The new key replaces the existing key, which requires you to update the known hosts tables on remote systems with the new host key so that future connections succeed. You can update the known hosts tables on remote systems using the **ssh host-key** command.

**Step 3** Display the current SSH server host key. The sensor displays the RSA1 (SSHv1) and the RSA (SSHv2) keys.

```
sensor# show ssh server-key
RSA1 Key: 2048 65537 21281522654207836691935756443555553045267729811163188158123
19221105282706971156372834395971442406711584211050336190060471149429624658777668
77772213935517164097668879617982452141055538222385354775671320871325156194364534
01964293900473562985740703815287014221909604447874565966052932190994239169399973
09451774745322103150629685832787333113245586019053499876586171642882215702812368
29552879443164044336379548314607826859130457872422419997317349512539506437629934
62410481738789963850532157078406908293224279442047344280835594921181229802312694
953678513524852911662327237984274983550808940893737598517019547234622411280871
RSA1 Bubble Babble: xucor-gidyg-comym-zipib-pilyk-vucal-pekyd-hipuc-tuven-gigyr-fixyx

RSA Key: AAAAB3NzaC1yc2EAAAADAQABAAQAC/zpG5nXqlAfc/aNzgQweipp9tjAqd1Xr5tuCJa+m
ccscEPHc25zUKLQG+8HkvieG2Pf11Y7mZk2POJICjXSV5izFczaFnZhBBUS+QOMH0nsk+S6F9Cujmz99
/OseRwQK16o2o2ds9otNfctAhVgex86SX5cWI0dyAYU1ZpwZCpK5sKxTtgZUkOEOSFPcHthf1DJvJfuj
1rWSJ/VAUHH4T5aomZ4m/is8jTp+nog5O5tDt6huqWYZt6MNTRoXTq0UgDgOlueoGM6Sqk0nCwUlsBG
TNaEoJbn0elRk+qNnSGo4FQYLVCryMZyi6GOxZVxwPST3IGuEHTunAw6aUnRl
RSA Bubble Babble: xucot-sapaf-sufiz-duriv-rigud-kezol-tupif-buvih-zokap-sohoz-kixox
sensor#
```

**For More Information**

For the procedure for updating the known hosts table, see [Adding Hosts to the SSH Known Hosts List](#), page 4-47.

## Configuring TLS

This section describes TLS on the sensor, and contains the following topics:

- [Understanding TLS](#), page 4-52
- [Adding TLS Trusted Hosts](#), page 4-53

- [Enabling Strict TLS Certificate Checks, page 4-54](#)
- [Adding and Updating TLS Trusted Root Certificates, page 4-55](#)
- [Displaying TLS Trusted Root Certificates, page 4-57](#)
- [Displaying and Generating the Server Certificate, page 4-58](#)

## Understanding TLS

The Cisco IPS contains a web server that is running the IDM. Management stations connect to this web server. Blocking forwarding sensors also connect to the web server of the master blocking sensor. To provide security, this web server uses an encryption protocol known as TLS, which is closely related to SSL protocol. When you enter a URL into the web browser that starts with `https://ip_address`, the web browser responds by using either TLS or SSL protocol to negotiate an encrypted session with the host.



### Caution

The web browser initially rejects the certificate presented by the IDM because it does not trust the certificate authority (CA).



### Note

The IDM is enabled by default to use TLS and SSL. We highly recommend that you use TLS and SSL.

The process of negotiating an encrypted session in TLS is called “handshaking,” because it involves a number of coordinated exchanges between client and server. The server sends its certificate to the client. The client performs the following three-part test on this certificate:

1. Is the issuer identified in the certificate trusted?

Every web browser ships with a list of trusted third-party CAs. If the issuer identified in the certificate is among the list of CAs trusted by your browser, the first test is passed.

2. Is the date within the range of dates during which the certificate is considered valid?

Each certificate contains a Validity field, which is a pair of dates. If the date falls within this range of dates, the second test is passed.

3. Does the common name of the subject identified in the certificate match the URL hostname?

The URL hostname is compared with the subject common name. If they match, the third test is passed.

When you direct your web browser to connect with the IDM, the certificate that is returned fails because the sensor issues its own certificate (the sensor is its own CA) and the sensor is not already in the list of CAs trusted by your browser.

When you receive an error message from your browser, you have three options:

- Disconnect from the site immediately.
- Accept the certificate for the remainder of the web browsing session.
- Add the issuer identified in the certificate to the list of trusted CAs of the web browser and trust the certificate until it expires.

The most convenient option is to permanently trust the issuer. However, before you add the issuer, use out-of-band methods to examine the fingerprint of the certificate. This prevents you from being victimized by an attacker posing as a sensor. Confirm that the fingerprint of the certificate appearing in your web browser is the same as the one on your sensor.

**Caution**

If you change the organization name or hostname of the sensor, a new certificate is generated the next time the sensor is rebooted. The next time your web browser connects to the IDM, you will receive the manual override dialog boxes. You must perform the certificate fingerprint validation again for Internet Explorer and Firefox.

## Adding TLS Trusted Hosts

In certain situations, the sensor uses TLS/SSL to protect a session it establishes with a remote web server. For these sessions to be secure from man-in-the-middle attacks you must establish trust of the TLS certificates of the remote web servers. A copy of the TLS certificate of each trusted remote host is stored in the trusted hosts list.

Use the **tls trusted-host ip-address ip-address [port port]** command to add a trusted host to the trusted hosts list. This command retrieves the TLS certificate from the specified host/port and displays its fingerprint. You can accept or reject the fingerprint based on information retrieved directly from the host you are requesting to add. The default port is 443.

Each certificate is stored with an identifier field (**id**). For the IP address and default port, the identifier field is **ipaddress**. For the IP address and specified port, the identifier field is **ipaddress:port**.

**Caution**

TLS at the specified IP address is contacted to obtain the required fingerprint over the network. The specified host must be accessible at the moment the command is issued. Use an alternate method to confirm the fingerprint to protect yourself from accepting a certificate of an attacker.

To add a trusted host to the trusted hosts list, follow these steps:

**Step 1** Log in to the CLI using an account with administrator or operator privileges.

**Step 2** Add the trusted host.

```
sensor# configure terminal
sensor(config)# tls trusted-host ip-address 10.16.0.0
Certificate SHA1 fingerprint is B1:6F:F5:DA:F3:7A:FB:FB:93:E9:2D:39:B9:99:08:D4:
47:02:F6:12
Would you like to add this to the trusted certificate table for this host?[yes]:
```

The SHA1 fingerprints appear. You are prompted to add the trusted host.

If the connection cannot be established, the transaction fails.

```
sensor(config)# tls trusted-host ip-address 10.89.146.110 port 8000
Error: getHostCertificate : socket connect failed [4,111]
```

**Step 3** Enter **yes** to accept the fingerprint. The host is added to the TLS trusted host list. The Certificate ID stored for the requested certificate is displayed when the command is successful.

```
Certificate ID: 10.89.146.110 successfully added to the TLS trusted host table.
sensor(config)#
```

**Step 4** Verify that the host was added.

```
sensor(config)# exit
sensor# show tls trusted-hosts
10.89.146.110
sensor#
```

**Step 5** View the fingerprint for a specific host.

```
sensor# show tls trusted-hosts 10.89.146.110
SHA1: B1:6F:F5:DA:F3:7A:FB:FB:93:E9:2D:39:B9:99:08:D4:47:02:F6:12
sensor#
```

**Step 6** Remove an entry from the trusted hosts list.

```
sensor# configure terminal
sensor(config)# no tls trusted-host 10.89.146.110
```

**Step 7** Verify the entry was removed from the trusted host list. The IP address no longer appears in the list.

```
sensor(config)# exit
sensor# show tls trusted-hosts
No entries
```

## Enabling Strict TLS Certificate Checks

Use the `strict-tls-server-validation {enable | disable}` command in service web server submode to enable the sensor to use strict TLS certificate validation for global correlation updates and automatic signature updates. The default is disabled. If the root CA validation fails, the TLS connection is not established to download the signature updates from the Cisco server or the global correlation updates from the update server.

To enable strict TLS server validation, follow these steps:

**Step 1** Log in to the sensor using an account with administrator privileges.

**Step 2** Enter web server submode.

```
sensor# configure terminal
sensor(config)# service web-server
```

**Step 3** Make sure TLS is enabled.

```
sensor(config-web)# enable-tls true
```

If you disable TLS, you receive this message:

```
Warning: TLS protocol support has been disabled. This change will not take effect until
the web server is re-started.
```

**Step 4** Enable strict TLS server validation.

```
sensor(config-web)# strict-tls-server-validation enable
```

**Step 5** Verify the web server changes.

```
sensor(config-web)# show settings
enable-tls: true <defaulted>
strict-tls-server-validation: enable default: disable
port: 443 <defaulted>
server-id: HTTP/1.1 compliant <defaulted>
configurable-service (min: 0, max: 99, current: 0)

websession-inactivity-timeout: 3600 <defaulted>
enable-websession-inactivity-timeout-logging: false <defaulted>
tls-client-ciphers-restriction: enable <defaulted>
```

```
sensor(config-web)#
```

**Step 6** Exit web server submode.

```
sensor(config-web)# exit
Apply Changes:[yes]:
```

**Step 7** Press **Enter** to apply the changes or enter **no** to discard them.



**Note** If you enable TLS settings, you must reset the sensor to make the web server uses the new settings.

## Adding and Updating TLS Trusted Root Certificates

Use the **tls trusted-root-certificate** command in global configuration mode to add a new certificate or to update an existing certificate to the trusted root certificates list. The protocols used for copying the TLS root certificate are SCP and HTTPs.

The following commands apply:

- *destination-url*—Specifies the location of the TLS trusted root certificate.
- **scp**:—Source or destination URL for the SCP network server. The syntax for this prefix is:

```
scp://[[username@]location][relativeDirectory]/filename
scp://[[username@]location][absoluteDirectory]/filename
```



**Note** If you use SCP protocol, you are prompted for a password and you must add the remote host to the SSH known hosts list.

- **https**:—Source URL for the Web server. The syntax for this prefix is:

```
https://[[username@]location][directory]/filename
```

To add a trusted root certificate to the trusted root certificates list, follow these steps:

**Step 1** Log in to the CLI using an account with administrator privileges.

**Step 2** Add the trusted root certificate.

```
sensor# configure terminal
sensor(config)# tls trusted-root-certificate scp:
User: jsmith
Server's IP Address: 173.39.51.249
Port[22]:
File name: /ws/jsmith-bgl/CertiPostRootCert.cer
Password: *****
SHA1 fingerprint of this certificate is
74:2c:df:15:94:04:9c:bf:17:a2:04:6c:c6:39:bb:38:88:e0:2e:33
Would you like to add this to the TLS trusted certificate store (yes/no)?[yes]:
```

- Step 3** Enter **yes** to accept the trusted root certificate. The certificate is added to the TLS trusted root certificates list.

```
Certificate with SHA1 fingerprint
74:2c:df:15:94:04:9c:bf:17:a2:04:6c:c6:39:bb:38:88:e0:2e:33 successfully added to the TLS
trusted root certificate store.
sensor(config)#
```

If the connection cannot be established, the transaction fails.

```
sensor(config)# tls trusted-root-certificate
Error: Unable to receive the certificate file from the specified URL - ssh: connect to
host 10.89.12.45 port 22: Connection timed out
```

- Step 4** Update the trusted root certificate.

```
sensor(config)# tls trusted-root-certificate scp:
User: jsmith
Server's IP Address: 173.39.51.249
Port[22]:
File name: /ws/jsmith-bgl/CertiPostRootCert.cer
Password: *****
SHA1 fingerprint of this certificate is
74:2c:df:15:94:04:9c:bf:17:a2:04:6c:c6:39:bb:38:88:e0:2e:33
This certificate is going to replace an existing certificate in the TLS trusted
certificate store.
Validity of the existing certificate is same as the one you are trying to add.
Are you sure you want to proceed (yes/no)?[yes]:
```

- Step 5** Enter **yes** to update the trusted root certificate. The certificate is updated in the TLS trusted root certificates list.

```
Certificate with SHA1 fingerprint
74:2c:df:15:94:04:9c:bf:17:a2:04:6c:c6:39:bb:38:88:e0:2e:33 already exists in the TLS
trusted root certificate store.
sensor(config)#
```

- Step 6** Verify that the trusted root certificate was added.

```
sensor(config)# exit
sensor# show tls trusted-root-certificates
```

```
TLS Certificate Name: GeoTrust Global CA
```

```
Issued To: c=US,o=GeoTrust Inc.,cn=GeoTrust Global CA
```

```
Issued By: c=US,o=GeoTrust Inc.,cn=GeoTrust Global CA
```

```
SHA1-fingerprint: de:28:f4:a4:ff:e5:b9:2f:a3:c5:03:d1:a3:49:a7:f9:96:2a:82:12
```

```
MD5-fingerprint: f7:75:ab:29:fb:51:4e:b7:77:5e:ff:05:3c:99:8e:f5
```

```
Expiration Date: Sat May 21 04:00:00 2022
```

```
sensor#
```

## Displaying TLS Trusted Root Certificates

Use the **show tls trusted-root-certificates** command in privileged EXEC mode to list the current TLS trusted root certificates on the sensor.

To display the TLS trusted root certificates on the sensor, follow these steps:

**Step 1** Log in to the CLI.

**Step 2** Display the TLS trusted root certificates:

```

sensor# show tls trusted-root-certificates
TLS Certificate Name: GeoTrust Global CA
Issued To: c=US,o=GeoTrust Inc.,cn=GeoTrust Global CA
Issued By: c=US,o=GeoTrust Inc.,cn=GeoTrust Global CA
SHA1-fingerprint: de:28:f4:a4:ff:e5:b9:2f:a3:c5:03:d1:a3:49:a7:f9:96:2a:82:12
MD5-fingerprint: f7:75:ab:29:fb:51:4e:b7:77:5e:ff:05:3c:99:8e:f5
Expiration Date: Sat May 21 04:00:00 2022

TLS Certificate Name: GeoTrust Primary Certification Authority
Issued To: c=US,o=GeoTrust Inc.,cn=GeoTrust Primary Certification Authority
Issued By: c=US,o=GeoTrust Inc.,cn=GeoTrust Primary Certification Authority
SHA1-fingerprint: 32:3c:11:8e:1b:f7:b8:b6:52:54:e2:e2:10:0d:d6:02:90:37:f0:96
MD5-fingerprint: 02:26:c3:01:5e:08:30:37:43:a9:d0:7d:cf:37:e6:bf
Expiration Date: Wed Jul 16 23:59:59 2036

TLS Certificate Name: GeoTrust Universal CA
Issued To: c=US,o=GeoTrust Inc.,cn=GeoTrust Universal CA
Issued By: c=US,o=GeoTrust Inc.,cn=GeoTrust Universal CA
SHA1-fingerprint: e6:21:f3:35:43:79:05:9a:4b:68:30:9d:8a:2f:74:22:15:87:ec:79
MD5-fingerprint: 92:65:58:8b:a2:1a:31:72:73:68:5c:b4:a5:7a:07:48
Expiration Date: Sun Mar 4 05:00:00 2029

TLS Certificate Name:
Issued To: c=US,o=VeriSign, Inc.,ou=Class 3 Public Primary Certification
Authority
Issued By: c=US,o=VeriSign, Inc.,ou=Class 3 Public Primary Certification
Authority
SHA1-fingerprint: a1:db:63:93:91:6f:17:e4:18:55:09:40:04:15:c7:02:40:b0:ae:6b
MD5-fingerprint: ef:5a:f1:33:ef:f1:cd:bb:51:02:ee:12:14:4b:96:c4
Expiration Date: Wed Aug 2 23:59:59 2028

TLS Certificate Name: Thawte Premium Server CA
Issued To: c=ZA,st=Western Cape,l=Cape Town,o=Thawte Consulting
cc,ou=Certification Services Division,cn=Thawte Premium Server
CA,e=premium-server@thawte.com
Issued By: c=ZA,st=Western Cape,l=Cape Town,o=Thawte Consulting
cc,ou=Certification Services Division,cn=Thawte Premium Server
CA,e=premium-server@thawte.com
SHA1-fingerprint: e0:ab:05:94:20:72:54:93:05:60:62:02:36:70:f7:cd:2e:fc:66:66
MD5-fingerprint: a6:6b:60:90:23:9b:3f:2d:bb:98:6f:d6:a7:19:0d:46
Expiration Date: Fri Jan 1 23:59:59 2021
sensor#

```

## Displaying and Generating the Server Certificate

A TLS certificate is generated when the sensor is first started. Use the **tls generate-key** command to generate a new server self-signed X.509 certificate. The IP address of the sensor is included in the certificate. If you change the sensor IP address, the sensor automatically generates a new certificate.

**Caution**

The new certificate replaces the existing certificate, which requires you to update the trusted hosts lists on remote systems with the new certificate so that future connections succeed. You can update the trusted hosts lists on remote IPS sensors using the **tls trusted-host** command. If the sensor is a master blocking sensor, you must update the trusted hosts lists on the remote sensors that are sending block requests to the master blocking sensor.

To generate a new TLS certificate, follow these steps:

**Step 1** Log in to the CLI using an account with administrator privileges.

**Step 2** Generate the new certificate.

```
sensor# tls generate-key
SHA1 fingerprint is 4A:2B:79:A0:82:8B:65:3A:83:B5:D9:50:C0:8E:F6:C6:B0:30:47:BB
sensor#
```

**Step 3** Verify that the key was generated.

```
sensor# show tls fingerprint
SHA1: 4A:2B:79:A0:82:8B:65:3A:83:B5:D9:50:C0:8E:F6:C6:B0:30:47:BB
sensor#
```

**For More Information**

For the procedure for updating the trusted hosts lists on remote sensors, see [Adding TLS Trusted Hosts, page 4-53](#).

## Installing the License Key

This section describes the IPS license key and how to install it. It contains the following topics:

- [Understanding the License Key, page 4-59](#)
- [Service Programs for IPS Products, page 4-59](#)
- [Obtaining and Installing the License Key, page 4-60](#)
- [Licensing the ASA 5500-X IPS SSP, page 4-62](#)
- [Uninstalling the License Key, page 4-62](#)



## Understanding the License Key

Although the sensor functions without the license key, you must have a license key to obtain signature updates and use the global correlation features. To obtain a license key, you must have the following:

- Cisco Service for IPS service contract—Contact your reseller, Cisco service or product sales to purchase a contract.
- Your IPS device serial number—To find the IPS device serial number in the IDM or the IME, for the IDM choose **Configuration > Sensor Management > Licensing**, and for the IME choose **Configuration > sensor\_name > Sensor Management > Licensing**, or in the CLI use the **show version** command.
- Valid Cisco.com username and password.

Trial license keys are also available. If you cannot get your sensor licensed because of problems with your contract, you can obtain a 60-day trial license that supports signature updates that require licensing.

You can obtain a license key from the Cisco.com licensing server, which is then delivered to the sensor. Or, you can update the license key from a license key provided in a local file. Go to <http://www.cisco.com/go/license> and click **IPS Signature Subscription Service** to apply for a license key.

You can view the status of the license key in these places:

- The IDM Home window Licensing section on the Health tab
- The IDM Licensing pane (**Configuration > Licensing**)
- The IME Home page in the Device Details section on the Licensing tab
- License Notice at CLI login

Whenever you start the IDM, the IME, or the CLI, you are informed of your license status—whether you have a trial, invalid, or expired license key. With no license key, an invalid license key, or an expired license key, you can continue to use the IDM, the IME, and the CLI, but you cannot download signature updates.

If you already have a valid license on the sensor, you can click **Download** on the License pane to download a copy of your license key to the computer that the IDM or the IME is running on and save it to a local file. You can then replace a lost or corrupted license, or reinstall your license after you have reimaged the sensor.

## Service Programs for IPS Products

You must have a Cisco Services for IPS service contract for any IPS product so that you can download a license key and obtain the latest IPS signature updates. If you have a direct relationship with Cisco Systems, contact your account manager or service account manager to purchase the Cisco Services for IPS service contract. If you do not have a direct relationship with Cisco Systems, you can purchase the service account from a one-tier or two-tier partner.

When you purchase the following IPS products you must also purchase a Cisco Services for IPS service contract:

- IPS 4345
- IPS 4345-DC
- IPS 4360
- IPS 4510

- IPS 4520

When you purchase an ASA 5500 series adaptive security appliance product that does not contain IPS, you must purchase a SMARTnet contract.

**Note**

SMARTnet provides operating system updates, access to Cisco.com, access to TAC, and hardware replacement NBD on site.

When you purchase an ASA 5500 series adaptive security appliance product that ships with an IPS module installed, or if you purchase one to add to your ASA 5500 series adaptive security appliance product, you must purchase the Cisco Services for IPS service contract.

**Note**

Cisco Services for IPS provides IPS signature updates, operating system updates, access to Cisco.com, access to TAC, and hardware replacement NBD on site.

For example, if you purchase an ASA 5585-X and then later want to add IPS and purchase an ASA-IPS10-K9, you must now purchase the Cisco Services for IPS service contract. After you have the Cisco Services for IPS service contract, you must also have your product serial number to apply for the license key.

**Caution**

If you ever send your product for RMA, the serial number changes. You must then get a new license key for the new serial number.

## Obtaining and Installing the License Key

**Note**

You cannot install an older license key over a newer license key.

Use the **copy source-url license\_file\_name license-key** command to copy the license key to your sensor.

The following commands apply:

- *source-url*—The location of the source file to be copied. It can be a URL or keyword.
- *destination-url*—The location of the destination file to be copied. It can be a URL or a keyword.
- **license-key**—The subscription license file.
- *license\_file\_name*—The name of the license file you receive.

The exact format of the source and destination URLs varies according to the file. Here are the valid types:

- **ftp**:—Source or destination URL for an FTP network server. The syntax for this prefix is:

```
ftp://[[username@]location][relativeDirectory]/filename
```

```
ftp://[[username@]location][absoluteDirectory]/filename
```

- **scp**:—Source or destination URL for the SCP network server. The syntax for this prefix is:

```
scp://[[username@]location][relativeDirectory]/filename
```

```
scp://[[username@]location][absoluteDirectory]/filename
```



**Note** If you use FTP or SCP protocol, you are prompted for a password. If you use SCP protocol, you must add the remote host to the SSH known hosts list.

- **http:**—Source URL for the Web server. The syntax for this prefix is:  
http://[[username@]location][directory]/filename
- **https:**—Source URL for the Web server. The syntax for this prefix is:  
https://[[username@]location][directory]/filename



**Note** If you use HTTPS protocol, the remote host must be a TLS trusted host.

### Installing the License Key

To install the license key, follow these steps:

**Step 1** Log in to [Cisco.com](http://Cisco.com).

**Step 2** Apply for the license key at this URL: [www.cisco.com/go/license](http://www.cisco.com/go/license).



**Note** In addition to a valid Cisco.com username and password, you must also have a Cisco Services for IPS service contract before you can apply for a license key.

**Step 3** Fill in the required fields. Your Cisco IPS Signature Subscription Service license key will be sent by email to the e-mail address you specified.



**Note** You must have the correct IPS device serial number and product identifier (PID) because the license key only functions on the device with that number.

**Step 4** Save the license key to a system that has a Web server, FTP server, or SCP server.

**Step 5** Log in to the CLI using an account with administrator privileges.

**Step 6** Copy the license key to the sensor.

```
sensor# copy scp://user@192.168.1.2/24://tftpboot/dev.lic license-key
Password: *****
```

**Step 7** Verify the sensor is licensed.

```
sensor# show version
Application Partition:

Cisco Intrusion Prevention System, Version 7.3(1)E4

Host:
 Realm Keys key1.0
Signature Definition:
 Signature Update S741.0 2013-09-10
 Threat Profile Version 2
OS Version: 2.6.29.1
Platform: IPS-4360
Serial Number: FGL1702401M
Licensed, expires: 21-Nov-2014 UTC
Sensor up-time is 22:09.
```

```

Using 14372M out of 15943M bytes of available memory (90% usage)
system is using 32.4M out of 160.0M bytes of available disk space (20% usage)
application-data is using 85.6M out of 376.4M bytes of available disk space (24% usage)
boot is using 63.1M out of 70.2M bytes of available disk space (95% usage)
application-log is using 494.0M out of 513.0M bytes of available disk space (96% usage)

```

```

MainApp C-2013_12_16_14_00_7_3_0_143 (Release) 2013-12-16T14:06:20-0600
Running
AnalysisEngine C-2013_12_16_14_00_7_3_0_143 (Release) 2013-12-16T14:06:20-0600
Running
CollaborationApp C-2013_12_16_14_00_7_3_0_143 (Release) 2013-12-16T14:06:20-0600
Running
CLI C-2013_12_16_14_00_7_3_0_143 (Release) 2013-12-16T14:06:20-0600

```

Upgrade History:

```
IPS-K9-7.3-1-E4 11:22:07 UTC Sat Jan 19 2013
```

Recovery Partition Version 1.1 - 7.3(1)E4

Host Certificate Valid from: 09-Oct-2014 to 09-Oct-2016

sensor#

---

## Licensing the ASA 5500-X IPS SSP

For the ASA 5500-X series adaptive security appliances with the IPS SSP, the ASA requires the IPS Module license. To view your current ASA licenses, in ASDM choose **Home > Device Dashboard > Device Information > Device License**. For more information about ASA licenses, refer to the licensing chapter in the configuration guide. After you obtain the ASA IPS Module license, you can obtain and install the IPS license key.

### For More Information

- For more information about getting started using the ASA 5500-X IPS SSP, refer to the [Cisco IPS Module on the ASA Quick Start Guide](#).
- For the procedures for obtaining and installing the IPS License key, see

## Uninstalling the License Key

Use the **erase license-key** command to uninstall the license key on your sensor. This allows you to delete an installed license key from a sensor without restarting the sensor or logging into the sensor using the service account.

To uninstall the license key, follow these steps:

---

**Step 1** Log in to the CLI using an account with administrator privileges.

**Step 2** Uninstall the license key on the sensor.

```
sensor# erase license-key
```

```
Warning: Executing this command will remove the license key installed on the sensor.
```

You must have a valid license key installed on the sensor to apply the Signature Updates and use the Global Correlation features.

```
Continue? []: yes
sensor#
```

**Step 3** Verify the sensor key has been uninstalled.

```
sensor# show version
Application Partition:

Cisco Intrusion Prevention System, Version 7.3(1)E4

Host:
 Realm Keys key1.0
Signature Definition:
 Signature Update S741.0 2013-09-10
 Threat Profile Version 2
OS Version: 2.6.29.1
Platform: IPS-4360
Serial Number: FGL1702401M
No license present
Sensor up-time is 22:09.
Using 14372M out of 15943M bytes of available memory (90% usage)
system is using 32.4M out of 160.0M bytes of available disk space (20% usage)
application-data is using 85.6M out of 376.4M bytes of available disk space (24% usage)
boot is using 63.1M out of 70.2M bytes of available disk space (95% usage)
application-log is using 494.0M out of 513.0M bytes of available disk space (96% usage)

MainApp C-2013_12_16_14_00_7_3_0_143 (Release) 2013-12-16T14:06:20-0600
Running
AnalysisEngine C-2013_12_16_14_00_7_3_0_143 (Release) 2013-12-16T14:06:20-0600
Running
CollaborationApp C-2013_12_16_14_00_7_3_0_143 (Release) 2013-12-16T14:06:20-0600
Running
CLI C-2013_12_16_14_00_7_3_0_143 (Release) 2013-12-16T14:06:20-0600

Upgrade History:

 IPS-K9-7.3-1-E4 11:22:07 UTC Sat Jan 19 2013

Recovery Partition Version 1.1 - 7.3(1)E4

Host Certificate Valid from: 09-Oct-2014 to 09-Oct-2016

sensor#
```





## Configuring Interfaces

---

This chapter describes how to configure interfaces on the sensor. You configured the interfaces when you initialized the sensor with the **setup** command, but if you need to change or add anything to your interface configuration, use the following procedures. For more information on configuring interfaces using the **setup** command, see [Chapter 3, “Initializing the Sensor.”](#)

This chapter contains the following sections:

- [Understanding Interfaces, page 5-2](#)
- [Configuring Physical Interfaces, page 5-11](#)
- [Configuring Promiscuous Mode, page 5-14](#)
- [Configuring Inline Interface Mode, page 5-16](#)
- [Configuring Inline VLAN Pair Mode, page 5-21](#)
- [Configuring VLAN Group Mode, page 5-26](#)
- [Configuring LACP for the 4500 Series Sensors, page 5-33](#)
- [Configuring Inline Bypass Mode, page 5-42](#)
- [Configuring Interface Notifications, page 5-44](#)
- [Configuring CDP Mode, page 5-45](#)
- [Displaying Interface Statistics, page 5-46](#)
- [Displaying Interface Traffic History, page 5-48](#)

## Interface Notes and Caveats

The following notes and caveats apply to configuring interfaces on the sensor:

- On appliances, all sensing interfaces are disabled by default. You must enable them to use them. On modules, the sensing interfaces are permanently enabled.
- There is only one sensing interface on the ASA IPS modules (ASA 5500-X IPS SSP and ASA 5585-X IPS SSP), so you cannot designate an alternate TCP reset interface.
- You can only assign a sensing interface as an alternate TCP reset interface. You cannot configure the management interface as an alternate TCP reset interface.
- You configure the ASA IPS modules (ASA 5500-X IPS SSP and ASA 5585-X IPS SSP) for promiscuous mode from the adaptive security appliance CLI and not from the Cisco IPS CLI.

- You can configure the ASA IPS modules (ASA 5500-X IPS SSP and ASA 5585-X IPS SSP) to operate inline even though they have only one sensing interface.
- The ASA IPS modules (ASA 5500-X IPS SSP and ASA 5585-X IPS SSP) do not support inline VLAN pairs.
- The ASA IPS modules (ASA 5500-X IPS SSP and ASA 5585-X IPS SSP) do not support VLAN groups mode.
- There are security consequences when you put the sensor in bypass mode. When bypass mode is on, the traffic bypasses the sensor and is not inspected; therefore, the sensor cannot prevent malicious attacks.
- As with signature updates, when the sensor applies a global correlation update, it may trigger bypass. Whether or not bypass is triggered depends on the traffic load of the sensor and the size of the signature/global correlation update. If bypass mode is turned off, an inline sensor stops passing traffic while the update is being applied.
- The ASA 5500-X IPS SSP and ASA 5585-X IPS SSP do not support bypass mode. The adaptive security appliance will either fail open, fail close, or fail over depending on the configuration of the adaptive security appliance and the type of activity being done on the IPS.
- The **show interface** command output for the IPS 4500 series sensors does not include the total undersize packets or total transmit FIFO overruns.
- When the IPS 4500 series sensors are configured in VLAN pairs, the **packet display** command does not work without the VLAN option if the **expression** keyword is also used.
- For the IPS 4500 series sensors, the maximum number of inline VLAN pairs you can create system wide is 150. On all other platforms, the limit is 255 per interface.
- On the IPS 4500 series sensors, no interface-related configurations are allowed when the SensorApp is down.

## Understanding Interfaces

This section describes the IPS interfaces and modes, and contains the following topics:

- [IPS Interfaces, page 5-2](#)
- [Command and Control Interface, page 5-3](#)
- [Sensing Interfaces, page 5-4](#)
- [TCP Reset Interfaces, page 5-4](#)
- [Interface Support, page 5-6](#)
- [Interface Configuration Restrictions, page 5-8](#)
- [Interface Configuration Sequence, page 5-10](#)

## IPS Interfaces

The sensor interfaces are named according to the maximum speed and physical location of the interface. The physical location consists of a port number and a slot number. All interfaces that are built-in on the sensor motherboard are in slot 0, and the interface card expansion slots are numbered beginning with



slot 1 for the bottom slot with the slot numbers increasing from bottom to top. Each physical interface can be divided into VLAN group subinterfaces, each of which consists of a group of VLANs on that interface.

There are three interface roles:

- Command and control
- Sensing
- Alternate TCP reset

There are restrictions on which roles you can assign to specific interfaces and some interfaces have multiple roles. You can configure any sensing interface to any other sensing interface as its TCP reset interface. The TCP reset interface can also serve as an IDS (promiscuous) sensing interface at the same time. The following restrictions apply:

- The TCP reset interface that is assigned to a sensing interface has no effect in inline interface or inline VLAN pair mode, because TCP resets are always sent on the sensing interfaces in those modes.
- There is only one sensing interface on the ASA IPS modules (ASA 5500-X IPS SSP and ASA 5585-X IPS SSP), so you cannot designate an alternate TCP reset interface.
- On the IPS 4510 and IPS 4520, no interface-related configurations are allowed when the SensorApp is down.

## Command and Control Interface

The command and control interface has an IP address and is used for configuring the sensor. It receives security and status events from the sensor and queries the sensor for statistics. The command and control interface is permanently enabled. It is permanently mapped to a specific physical interface, which depends on the specific model of sensor. You cannot use the command and control interface as either a sensing or alternate TCP reset interface.

Table 5-1 lists the command and control interfaces for each sensor.

**Table 5-1** *Command and Control Interfaces*

| Sensor                | Command and Control Interface |
|-----------------------|-------------------------------|
| ASA 5512-X IPS SSP    | Management 0/0                |
| ASA 5515-X IPS SSP    | Management 0/0                |
| ASA 5525-X IPS SSP    | Management 0/0                |
| ASA 5545-X IPS SSP    | Management 0/0                |
| ASA 5555-X IPS SSP    | Management 0/0                |
| ASA 5585-X IPS SSP-10 | Management 0/0                |
| ASA 5585-X IPS SSP-20 | Management 0/0                |
| ASA 5585-X IPS SSP-40 | Management 0/0                |
| ASA 5585-X IPS SSP-60 | Management 0/0                |
| IPS 4345              | Management 0/0                |
| IPS 4345-DC           | Management 0/0                |
| IPS 4360              | Management 0/0                |

**Table 5-1** *Command and Control Interfaces (continued)*

| Sensor   | Command and Control Interface |
|----------|-------------------------------|
| IPS 4510 | Management 0/0 <sup>1</sup>   |
| IPS 4520 | Management 0/0 <sup>1</sup>   |

1. The 4500 series sensors have two management ports, Management 0/0 and Management 0/1, but Management 0/1 is reserved for future use.

## Sensing Interfaces

Sensing interfaces are used by the sensor to analyze traffic for security violations. A sensor has one or more sensing interfaces depending on the sensor. Sensing interfaces can operate individually in promiscuous mode or you can pair them to create inline interfaces.



### Note

On appliances, all sensing interfaces are disabled by default. You must enable them to use them. On modules, the sensing interfaces are permanently enabled.

Some appliances support optional interface cards that add sensing interfaces to the sensor. You must insert or remove these optional cards while the sensor is powered off. The sensor detects the addition or removal of a supported interface card. If you remove an optional interface card, some of the interface configuration is deleted, such as the speed, duplex, description string, enabled/disabled state of the interface, and any inline interface pairings. These settings are restored to their default settings when the card is reinstalled. However, the assignment of promiscuous and inline interfaces to the Analysis Engine is not deleted from the Analysis Engine configuration, but is ignored until those cards are reinserted and you create the inline interface pairs again.

### For More Information

- For more information on supported interfaces, see [Interface Support, page 5-6](#).
- For more information on interface modes, see [Configuring Promiscuous Mode, page 5-14](#), [Configuring Inline Interface Mode, page 5-16](#), [Configuring Inline VLAN Pair Mode, page 5-21](#), [Configuring VLAN Group Mode, page 5-26](#), [Configuring Inline Bypass Mode, page 5-42](#).

## TCP Reset Interfaces

This section explains the TCP reset interfaces and when to use them. It contains the following topics:

- [Understanding Alternate TCP Reset Interfaces, page 5-4](#)
- [Designating the Alternate TCP Reset Interface, page 5-5](#)

## Understanding Alternate TCP Reset Interfaces

You can configure sensors to send TCP reset packets to try to reset a network connection between an attacker host and its intended target host. In some installations when the interface is operating in promiscuous mode, the sensor may not be able to send the TCP reset packets over the same sensing interface on which the attack was detected. In such cases, you can associate the sensing interface with an alternate TCP reset interface and any TCP resets that would otherwise be sent on the sensing interface when it is operating in promiscuous mode are instead sent out on the associated alternate TCP reset interface.

If a sensing interface is associated with an alternate TCP reset interface, that association applies when the sensor is configured for promiscuous mode but is ignored when the sensing interface is configured for inline mode. Any sensing interface can serve as the alternate TCP reset interface for another sensing interface

**Note**

There is only one sensing interface on the ASA IPS modules (ASA 5500-X IPS SSP and ASA 5585-X IPS SSP), so you cannot designate an alternate TCP reset interface.

Table 5-2 lists the alternate TCP reset interfaces.

**Table 5-2** *Alternate TCP Reset Interfaces*

| Sensor                | Alternate TCP Reset Interface |
|-----------------------|-------------------------------|
| ASA 5512-X IPS SSP    | None                          |
| ASA 5515-X IPS SSP    | None                          |
| ASA 5525-X IPS SSP    | None                          |
| ASA 5545-X IPS SSP    | None                          |
| ASA 5555-X IPS SSP    | None                          |
| ASA 5585-X IPS SSP-10 | None                          |
| ASA 5585-X IPS SSP-20 | None                          |
| ASA 5585-X IPS SSP-40 | None                          |
| ASA 5585-X IPS SSP-60 | None                          |
| IPS 4345              | Any sensing interface         |
| IPS 4345-DC           | Any sensing interface         |
| IPS 4360              | Any sensing interface         |
| IPS 4510              | Any sensing interface         |
| IPS 4520              | Any sensing interface         |
| IPS 4520-XL           | Any sensing interface         |

**For More Information**

For more information on choosing the alternate TCP interface, see [Designating the Alternate TCP Reset Interface, page 5-5](#).

## Designating the Alternate TCP Reset Interface

**Note**

There is only one sensing interface on the ASA IPS modules (ASA 5500-X IPS SSP and ASA 5585-X IPS SSP), so you cannot designate an alternate TCP reset interface.

You need to designate an alternate TCP reset interface in the following situations:

- When a switch is being monitored with either SPAN or VACL capture and the switch does not accept incoming packets on the SPAN or VACL capture port.

- When a switch is being monitored with either SPAN or VACL capture for multiple VLANs, and the switch does not accept incoming packets with 802.1q headers. The TCP resets need 802.1q headers to tell which VLAN the resets should be sent on.
- When a network tap is used for monitoring a connection. Taps do not permit incoming traffic from the sensor.

**Caution**

You can only assign a sensing interface as an alternate TCP reset interface. You cannot configure the management interface as an alternate TCP reset interface.

## Interface Support

Table 5-3 describes the interface support for appliances and modules running Cisco IPS.

**Table 5-3**      *Interface Support*

| Base Chassis          | Added Interface Cards | Interfaces Supporting Inline VLAN Pairs (Sensing Ports)                           | Combinations Supporting Inline Interface Pairs                                    | Interfaces Not Supporting Inline (Command and Control Port) |
|-----------------------|-----------------------|-----------------------------------------------------------------------------------|-----------------------------------------------------------------------------------|-------------------------------------------------------------|
| ASA 5512-X IPS SSP    | —                     | PortChannel 0/0 by security context instead of VLAN pair or inline interface pair | PortChannel 0/0 by security context instead of VLAN pair or inline interface pair | Management 0/0                                              |
| ASA 5515-X IPS SSP    | —                     | PortChannel 0/0 by security context instead of VLAN pair or inline interface pair | PortChannel 0/0 by security context instead of VLAN pair or inline interface pair | Management 0/0                                              |
| ASA 5525-X IPS SSP    | —                     | PortChannel 0/0 by security context instead of VLAN pair or inline interface pair | PortChannel 0/0 by security context instead of VLAN pair or inline interface pair | Management 0/0                                              |
| ASA 5545-X IPS SSP    | —                     | PortChannel 0/0 by security context instead of VLAN pair or inline interface pair | PortChannel 0/0 by security context instead of VLAN pair or inline interface pair | Management 0/0                                              |
| ASA 5555-X IPS SSP    | —                     | PortChannel 0/0 by security context instead of VLAN pair or inline interface pair | PortChannel 0/0 by security context instead of VLAN pair or inline interface pair | Management 0/0                                              |
| ASA 5585-X IPS SSP-10 | —                     | PortChannel 0/0 by security context instead of VLAN pair or inline interface pair | PortChannel 0/0 by security context instead of VLAN pair or inline interface pair | Management 0/0                                              |
| ASA 5585-X IPS SSP-20 | —                     | PortChannel 0/0 by security context instead of VLAN pair or inline interface pair | PortChannel 0/0 by security context instead of VLAN pair or inline interface pair | Management 0/0                                              |
| ASA 5585-X IPS SSP-40 | —                     | PortChannel 0/0 by security context instead of VLAN pair or inline interface pair | PortChannel 0/0 by security context instead of VLAN pair or inline interface pair | Management 0/0                                              |
| ASA 5585-X IPS SSP-60 | —                     | PortChannel 0/0 by security context instead of VLAN pair or inline interface pair | PortChannel 0/0 by security context instead of VLAN pair or inline interface pair | Management 0/0                                              |

Table 5-3 Interface Support (continued)

| Base Chassis | Added Interface Cards | Interfaces Supporting Inline VLAN Pairs (Sensing Ports)                                                                                                                              | Combinations Supporting Inline Interface Pairs | Interfaces Not Supporting Inline (Command and Control Port) |
|--------------|-----------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|------------------------------------------------|-------------------------------------------------------------|
| IPS 4345     | —                     | GigabitEthernet 0/0<br>GigabitEthernet 0/1<br>GigabitEthernet 0/2<br>GigabitEthernet 0/3<br>GigabitEthernet 0/4<br>GigabitEthernet 0/5<br>GigabitEthernet 0/6<br>GigabitEthernet 0/7 | All sensing ports can be paired together       | Management 0/0                                              |
| IPS 4345-DC  | —                     | GigabitEthernet 0/0<br>GigabitEthernet 0/1<br>GigabitEthernet 0/2<br>GigabitEthernet 0/3<br>GigabitEthernet 0/4<br>GigabitEthernet 0/5<br>GigabitEthernet 0/6<br>GigabitEthernet 0/7 | All sensing ports can be paired together       | Management 0/0                                              |
| IPS 4360     | —                     | GigabitEthernet 0/0<br>GigabitEthernet 0/1<br>GigabitEthernet 0/2<br>GigabitEthernet 0/3<br>GigabitEthernet 0/4<br>GigabitEthernet 0/5<br>GigabitEthernet 0/6<br>GigabitEthernet 0/7 | All sensing ports can be paired together       | Management 0/0                                              |

Table 5-3 Interface Support (continued)

| Base Chassis            | Added Interface Cards | Interfaces Supporting Inline VLAN Pairs (Sensing Ports)                                                                                                                                                                                        | Combinations Supporting Inline Interface Pairs | Interfaces Not Supporting Inline (Command and Control Port) |
|-------------------------|-----------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|------------------------------------------------|-------------------------------------------------------------|
| IPS 4510                | —                     | GigabitEthernet 0/0<br>GigabitEthernet 0/1<br>GigabitEthernet 0/2<br>GigabitEthernet 0/3<br>GigabitEthernet 0/4<br>GigabitEthernet 0/5<br>TenGigabitEthernet 0/6<br>TenGigabitEthernet 0/7<br>TenGigabitEthernet 0/8<br>TenGigabitEthernet 0/9 | All sensing ports can be paired together       | Management 0/0<br>Management 0/1 <sup>1</sup>               |
| IPS 4520<br>IPS 4520-XL | —                     | GigabitEthernet 0/0<br>GigabitEthernet 0/1<br>GigabitEthernet 0/2<br>GigabitEthernet 0/3<br>GigabitEthernet 0/4<br>GigabitEthernet 0/5<br>TenGigabitEthernet 0/6<br>TenGigabitEthernet 0/7<br>TenGigabitEthernet 0/8<br>TenGigabitEthernet 0/9 | All sensing ports can be paired together       | Management 0/0<br>Management 0/1 <sup>1</sup>               |

1. Reserved for future use.

## Interface Configuration Restrictions

The following restrictions apply to configuring interfaces on the sensor:

- Physical Interfaces
  - On the ASA IPS modules (ASA 5500-X IPS SSP and ASA 5585-X IPS SSP) all backplane interfaces have fixed speed, duplex, and state settings. These settings are protected in the default configuration on all backplane interfaces.
  - For nonbackplane FastEthernet interfaces the valid speed settings are 10 Mbps, 100 Mbps, and auto. Valid duplex settings are full, half, and auto.
  - For Gigabit copper interfaces (1000-TX on the IPS 4345, IPS 4360, IPS 4510, IPS 4520, and IPX 4520-XL), valid speed settings are 10 Mbps, 100 Mbps, 1000 Mbps, and auto. Valid duplex settings are full, half, and auto.

- For Gigabit (copper or fiber) interfaces, if the speed is configured for 1000 Mbps, the only valid duplex setting is auto.
- The command and control interface cannot also serve as a sensing interface.
- Inline Interface Pairs
  - Inline interface pairs can contain any combination of sensing interfaces regardless of the physical interface type (copper versus fiber), speed, or duplex settings of the interface. However, pairing interfaces of different media type, speeds, and duplex settings may not be fully tested or supported.
  - The command and control interface cannot be a member of an inline interface pair.
  - You cannot pair a physical interface with itself in an inline interface pair.
  - A physical interface can be a member of only one inline interface pair.
  - You can only configure bypass mode and create inline interface pairs on sensor platforms that support inline mode.
  - A physical interface cannot be a member of an inline interface pair unless the subinterface mode of the physical interface is **none**.
  - You can configure the ASA IPS modules (ASA 5500-X IPS SSP and ASA 5585-X IPS SSP) to operate inline even though they have only one sensing interface.
- Inline VLAN Pairs
  - You cannot pair a VLAN with itself.
  - You cannot use the default VLAN as one of the paired VLANs in an inline VLAN pair.
  - For a given sensing interface, a VLAN can be a member of only one inline VLAN pair. However, a given VLAN can be a member of an inline VLAN pair on more than one sensing interface.
  - The order in which you specify the VLANs in an inline VLAN pair is not significant.
  - A sensing interface in Inline VLAN Pair mode can have from 1 to 255 inline VLAN pairs.
  - You can enable LACP for inline VLAN pairs only on the 4500 series sensors.
  - The ASA IPS modules (ASA 5500-X IPS SSP and ASA 5585-X IPS SSP) do not support inline VLAN pairs.
  - For the IPS 4500 series sensors, the maximum number of inline VLAN pairs you can create system wide is 150. On all other platforms, the limit is 255 per interface.
- Alternate TCP Reset Interface
  - You can only assign the alternate TCP reset interface to a sensing interface. You cannot configure the command and control interface as an alternate TCP reset interface. The alternate TCP reset interface option is set to **none** as the default and is protected for all interfaces except the sensing interfaces.
  - You can assign the same physical interface as an alternate TCP reset interface for multiple sensing interfaces.
  - A physical interface can serve as both a sensing interface and an alternate TCP reset interface.
  - The command and control interface cannot serve as the alternate TCP reset interface for a sensing interface.
  - A sensing interface cannot serve as its own alternate TCP reset interface.
  - You can only configure interfaces that are capable of TCP resets as alternate TCP reset interfaces.

- There is only one sensing interface on the ASA IPS modules (ASA 5500-X IPS SSP and ASA 5585-X IPS SSP), so you cannot designate an alternate TCP reset interface.
- VLAN Groups
  - You can configure any single interface for promiscuous, inline interface pair, or inline VLAN pair mode, but no combination of these modes is allowed.
  - You cannot add a VLAN to more than one group on each interface.
  - You cannot add a VLAN group to multiple virtual sensors.
  - An interface can have no more than 255 user-defined VLAN groups.
  - When you pair a physical interface, you cannot subdivide it; you can subdivide the pair.
  - You can use a VLAN on multiple interfaces; however, you receive a warning for this configuration.
  - You can assign a virtual sensor to any combination of one or more physical interfaces and inline VLAN pairs, subdivided or not.
  - You can subdivide both physical and logical interfaces into VLAN groups.
  - The CLI, IDM, and IME prompt you to remove any dangling references. You can leave the dangling references and continue editing the configuration.
  - The CLI, IDM, and IME do not allow configuration changes in Analysis Engine that conflict with the interface configuration.
  - The CLI allows configuration changes in the interface configuration that cause conflicts in the Analysis Engine configuration. The IDM and IME do *not* allow changes in the interface configuration that cause conflicts in the Analysis Engine configuration.
  - The ASA IPS modules (ASA 5500-X IPS SSP and ASA 5585-X IPS SSP) do not support VLAN groups mode.
- Other Restrictions
  - A jumbo frame is an Ethernet packet that is larger than the standard maximum of 1518 bytes (including Layer 2 header and FCS). For IPS standalone appliances with 1 G and 10 G fixed or add-on interfaces, the maximum jumbo frame size is 9216 bytes. For integrated IPS sensors, such as the ASA 5500-X and ASA 5585-X series, refer to the following URL for information:  
[http://www.cisco.com/en/US/docs/security/asa/asa84/configuration/guide/interface\\_start.html#wp1328869](http://www.cisco.com/en/US/docs/security/asa/asa84/configuration/guide/interface_start.html#wp1328869)
  - On the IPS 4500 series, no interface-related configurations are allowed when the SensorApp is down.

#### For More Information

- For a list of supported sensor interfaces, see [Interface Support, page 5-6](#).
- For more information on alternate TCP reset, see [TCP Reset Interfaces, page 5-4](#).
- For more information on physical interfaces, see [Configuring Physical Interfaces, page 5-11](#).

## Interface Configuration Sequence

Follow these steps to configure interfaces on the sensor:

1. Configure the physical interface settings (speed, duplex, and so forth) and enable the interfaces.



2. Create or delete inline interfaces, inline VLAN subinterfaces, and VLAN groups, and set the inline bypass mode.
3. Assign the physical, subinterfaces, and inline interfaces to the virtual sensor.

#### For More Information

- For the procedure for configuring the physical interface settings, see [Configuring Physical Interfaces, page 5-11](#).
- For the procedures for creating and deleting different kinds of interfaces, see [Configuring Inline Interface Mode, page 5-16](#), [Configuring Inline VLAN Pair Mode, page 5-21](#), [Configuring VLAN Group Mode, page 5-26](#), and [Configuring Inline Bypass Mode, page 5-42](#).
- For the procedure for configuring virtual sensors, see [Adding, Editing, and Deleting Virtual Sensors, page 6-5](#).

## Configuring Physical Interfaces

Use the **physical-interfaces** *interface\_name* command in the service interface submode to configure promiscuous interfaces. The interface name is FastEthernet, GigabitEthernet, or PortChannel.



#### Note

You configure the ASA IPS modules (ASA 5500-X IPS SSP and ASA 5585-X IPS SSP) for promiscuous mode from the adaptive security appliance CLI and not from the Cisco IPS CLI.

The following commands apply:

- **admin-state {enabled | disabled}**—Specifies the administrative link state of the interface, whether the interface is enabled or disabled.



#### Note

On all backplane sensing interfaces on all modules, **admin-state** is set to enabled and is protected (you cannot change the setting). The **admin-state** has no effect (and is protected) on the command and control interface. It only affects sensing interfaces. The command and control interface does not need to be enabled because it cannot be monitored.

- **alt-tcp-reset-interface**—Sends TCP resets out an alternate interface when this interface is used for promiscuous monitoring and the reset action is triggered by a signature firing.



#### Note

You can only assign a sensing interface as an alternate TCP reset interface. You cannot configure the management interface as an alternate TCP reset interface.



#### Note

There is only one sensing interface on the ASA IPS modules (ASA 5500-X IPS SSP and ASA 5585-X IPS SSP), so you cannot designate an alternate TCP reset interface.

- *interface\_name*—Specifies the name of the interface on which TCP resets should be sent when this interface is used for promiscuous monitoring and the reset action is triggered by a signature firing. This setting is ignored when this interface is a member of an inline interface.
- **none**—Disables the use of an alternate TCP reset interface. TCP resets triggered by the reset action when in promiscuous mode will be sent out of this interface instead.

- **default**—Sets the value back to the system default setting.
- **description**—Specifies your description of the promiscuous interface.
- **duplex**—Specifies the duplex setting of the interface:
  - **auto**—Sets the interface to auto negotiate duplex.
  - **full**—Sets the interface to full duplex.
  - **half**—Sets the interface to half duplex.




---

**Note** The **duplex** option is protected on all modules.

---




---

**Note** For TenGigabit SFP+ ports, the permitted values are auto and full.

---

- **no**—Removes an entry or selection setting.
- **speed**—Specifies the speed setting of the interface:
  - **auto**—Sets the interface to auto negotiate speed.
  - **10**—Sets the interface to 10 MB (for TX interfaces only).
  - **100**—Sets the interface to 100 MB (for TX interfaces only).
  - **1000**—Sets the interface to 1 GB (for Gigabit interfaces only).




---

**Note** The **speed** option is protected on all modules.

---




---

**Note** For TenGigabit SFP+ ports with a 10 Gb connector, the permitted values are auto and 10000, and for TenGigabit SFP+ ports with a 1 Gb connector, the permitted value is auto.

---

### Configuring the Physical Interface Settings

To configure the physical interface settings for promiscuous mode on the sensor, follow these steps:

---

**Step 1** Log in to the CLI using an account with administrator privileges.

**Step 2** Enter interface submode.

```
sensor# configure terminal
sensor(config)# service interface
```

**Step 3** Display the list of available interfaces.

```
sensor(config-int)# physical-interfaces ?
GigabitEthernet0/0 GigabitEthernet0/0 physical interface.
GigabitEthernet0/1 GigabitEthernet0/1 physical interface.
GigabitEthernet0/2 GigabitEthernet0/2 physical interface.
GigabitEthernet0/3 GigabitEthernet0/3 physical interface.
Management0/0 Management0/0 physical interface.
sensor(config-int)# physical-interfaces
```

**Step 4** Specify the interface for promiscuous mode.

```
sensor(config-int)# physical-interfaces GigabitEthernet0/2
```

- Step 5** Enable the interface. You must assigned the interface to a virtual sensor and enable it before it can monitor traffic.
- ```
sensor(config-int-phy) # admin-state enabled
```
- Step 6** Add a description of this interface.
- ```
sensor(config-int-phy) # description INT1
```
- Step 7** Specify the duplex settings. This option is not available on the ASA IPS modules (ASA 5500-X IPS SSP and ASA 5585-X IPS SSP).
- ```
sensor(config-int-phy) # duplex full
```
- Step 8** Specify the speed. This option is not available on the ASA IPS modules (ASA 5500-X IPS SSP and ASA 5585-X IPS SSP).
- ```
sensor(config-int-phy) # speed 1000
```
- Step 9** Enable TCP resets for this interface if desired. This option is not available on the ASA IPS modules (ASA 5500-X IPS SSP and ASA 5585-X IPS SSP).
- ```
sensor(config-int-phy) # alt-tcp-reset-interface interface-name GigabitEthernet2/0
```
- Step 10** Repeat Steps 4 through 9 for any other interfaces you want to designate as promiscuous interfaces.
- Step 11** Verify the settings.



Note Make sure the `subinterface-type` is `none`, the default. You use the `subinterface-type` command to configure inline VLAN pairs.

```
sensor(config-int-phy) # show settings
<protected entry>
name: GigabitEthernet0/2
-----
media-type: tx <protected>
description: INT1 default:
admin-state: enabled default: disabled
duplex: full default: auto
speed: 1000 default: auto
alt-tcp-reset-interface
-----
interface-name: GigabitEthernet2/0
-----
subinterface-type
-----
none
-----
-----
sensor(config-int-phy) #
```

- Step 12** Remove TCP resets from an interface.
- ```
sensor(config-int-phy) # alt-tcp-reset-interface none
```
- Step 13** Verify the settings.

```
sensor(config-int-phy) # show settings
<protected entry>
name: GigabitEthernet0/0
```

```

media-type: tx <protected>
description: <defaulted>
admin-state: disabled <protected>
duplex: auto <defaulted>
speed: auto <defaulted>
alt-tcp-reset-interface

none

sensor(config-int-phy)#

```

**Step 14** Exit interface submode.

```

sensor(config-int-phy)# exit
sensor(config-int)# exit
Apply Changes?[yes]:

```

**Step 15** Press **Enter** to apply the changes or enter **no** to discard them.

#### For More Information

- For a list of possible interfaces for your sensor, see [Interface Support, page 5-6](#).
- For the procedure for sending traffic to the ASA 5500-X IPS SSP, see [Creating Virtual Sensors for the ASA 5500-X IPS SSP, page 18-3](#).
- For the procedure for sending traffic to the ASA 5585-X IPS SSP, see [Creating Virtual Sensors for the ASA 5585-X IPS SSP, page 19-4](#).
- For more information on the alternate TCP reset interface, see [Understanding Alternate TCP Reset Interfaces, page 5-4](#) and [Designating the Alternate TCP Reset Interface, page 5-5](#).
- For the procedure for configuring inline VLAN pairs, see [Configuring Inline VLAN Pairs, page 5-22](#).
- For the procedure for adding interfaces to virtual sensors, see [Adding, Editing, and Deleting Virtual Sensors, page 6-5](#).

## Configuring Promiscuous Mode

This section describes promiscuous mode on the sensor, and contains the following topics:

- [Understanding Promiscuous Mode, page 5-14](#)
- [Configuring Promiscuous Mode, page 5-15](#)
- [IPv6, Switches, and Lack of VACL Capture, page 5-15](#)

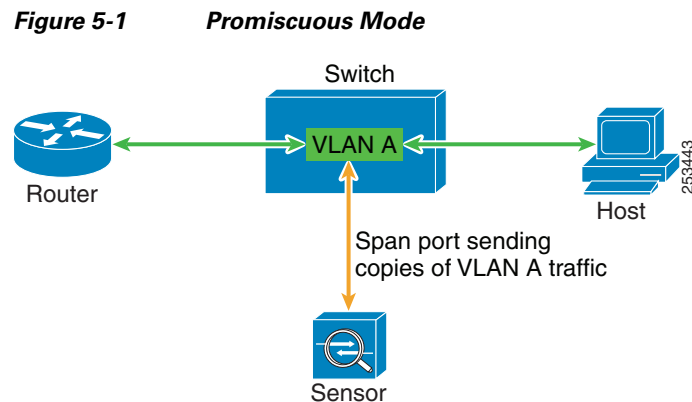
## Understanding Promiscuous Mode

In promiscuous mode, packets do not flow through the sensor. The sensor analyzes a copy of the monitored traffic rather than the actual forwarded packet. The advantage of operating in promiscuous mode is that the sensor does not affect the packet flow with the forwarded traffic. The disadvantage of

operating in promiscuous mode, however, is the sensor cannot stop malicious traffic from reaching its intended target for certain types of attacks, such as atomic attacks (single-packet attacks). The response actions implemented by promiscuous sensor devices are post-event responses and often require assistance from other networking devices, for example, routers and firewalls, to respond to an attack. While such response actions can prevent some classes of attacks, in atomic attacks the single packet has the chance of reaching the target system before the promiscuous-based sensor can apply an ACL modification on a managed device (such as a firewall, switch, or router).

By default, all sensing interfaces are in promiscuous mode. To change an interface from inline interface mode to promiscuous mode, delete any inline interface that contains that interface and delete any inline VLAN pair subinterfaces of that interface from the interface configuration.

Figure 5-1 illustrates promiscuous mode:



## Configuring Promiscuous Mode

By default, all sensing interfaces are in promiscuous mode. To change an interface from inline mode to promiscuous mode, delete the inline interface that contains that interface from the interface configuration.

## IPv6, Switches, and Lack of VACL Capture

VACLs on Catalyst switches do not have IPv6 support. The most common method for copying traffic to a sensor configured in promiscuous mode is to use VACL capture. If you want to have IPv6 support, you can use SPAN ports.

However, you can only configure up to two monitor sessions on a switch unless you use the following configuration:

- Monitor session
- Multiple trunks to one or more sensors
- Restrict per trunk port which VLANs are allowed to perform monitoring of many VLANs to more than two different sensors or virtual sensors within one IPS

The following configuration uses one SPAN session to send all of the traffic on any of the specified VLANs to all of the specified ports. Each port configuration only allows a particular VLAN or VLANs to pass. Thus you can send data from different VLANs to different sensors or virtual sensors all with one SPAN configuration line:

```
clear trunk 4/1-4 1-4094
set trunk 4/1 on dot1q 930
set trunk 4/2 on dot1q 932
set trunk 4/3 on dot1q 960
set trunk 4/4 on dot1q 962
set span 930, 932, 960, 962 4/1-4 both
```

**Note**

The SPAN/Monitor configuration is valuable when you want to assign different IPS policies per VLAN or when you have more bandwidth to monitor than one interface can handle.

## Configuring Inline Interface Mode

This section describes inline mode on the sensor, and contains the following topics:

- [Understanding Inline Interface Mode, page 5-16](#)
- [Configuring Inline Interface Pairs, page 5-17](#)

## Understanding Inline Interface Mode

Operating in inline interface pair mode puts the IPS directly into the traffic flow and affects packet-forwarding rates making them slower by adding latency. This allows the sensor to stop attacks by dropping malicious traffic before it reaches the intended target, thus providing a protective service. Not only is the inline device processing information on Layers 3 and 4, but it is also analyzing the contents and payload of the packets for more sophisticated embedded attacks (Layers 3 to 7). This deeper analysis lets the system identify and stop and/or block attacks that would normally pass through a traditional firewall device.

In inline interface pair mode, a packet comes in through the first interface of the pair on the sensor and out the second interface of the pair. The packet is sent to the second interface of the pair unless that packet is being denied or modified by a signature.

**Note**

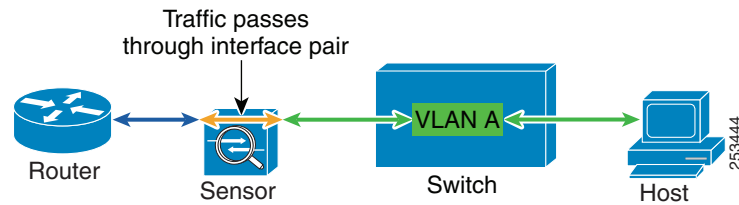
You can configure the ASA IPS modules (ASA 5500-X IPS SSP and ASA 5585-X IPS SSP) to operate inline even though they have only one sensing interface.

**Note**

If the paired interfaces are connected to the same switch, you should configure them on the switch as access ports with different access VLANs for the two ports. Otherwise, traffic does not flow through the inline interface.

Figure 5-2 illustrates inline interface pair mode:

**Figure 5-2 Inline Interface Pair Mode**



## Configuring Inline Interface Pairs

Use the **inline-interfaces** *name* command in the service interface submode to create inline interface pairs.



### Note

You can configure the ASA IPS modules (ASA 5500-X IPS SSP and ASA 5585-X IPS SSP) to operate inline even though they have only one sensing interface.

The following commands apply:

- **inline-interfaces** *name*—Specifies the name of the logical inline interface pair.
- **default**—Sets the value back to the system default setting.
- **description**—Specifies your description of the inline interface pair.
- **interface1** *interface\_name*—Specifies the first interface in the inline interface pair.
- **interface2** *interface\_name*—Specifies the second interface in the inline interface pair.
- **no**—Removes an entry or selection setting.
- **admin-state** {**enabled** | **disabled**}—Specifies the administrative link state of the interface, whether the interface is enabled or disabled.



### Note

On all backplane sensing interfaces on all modules, **admin-state** is set to enabled and is protected (you cannot change the setting). The **admin-state** has no effect (and is protected) on the command and control interface. It only affects sensing interfaces. The command and control interface does not need to be enabled because it cannot be monitored.

### Creating Inline Interface Pairs

To create inline interface pairs, follow these steps:

**Step 1** Log in to the CLI using an account with administrator privileges.

**Step 2** Enter interface submode.

```
sensor# configure terminal
sensor(config)# service interface
sensor(config-int)#
```

- Step 3** Verify that the subinterface mode is “none” for both of the physical interfaces you are pairing in the inline interface.

```
sensor(config-int)# show settings
physical-interfaces (min: 0, max: 999999999, current: 2)

<protected entry>
name: GigabitEthernet0/0 <defaulted>

media-type: tx <protected>
description: <defaulted>
admin-state: disabled <protected>
duplex: auto <defaulted>
speed: auto <defaulted>
alt-tcp-reset-interface

none

subinterface-type

none

```

- Step 4** Name the inline pair.

```
sensor(config-int)# inline-interfaces PAIR1
```

- Step 5** Display the available interfaces.

```
sensor(config-int)# interface1 ?
GigabitEthernet0/0 GigabitEthernet0/0 physical interface.
GigabitEthernet0/1 GigabitEthernet0/1 physical interface.
GigabitEthernet0/2 GigabitEthernet0/2 physical interface.
GigabitEthernet0/3 GigabitEthernet0/3 physical interface.
Management0/0 Management0/0 physical interface.
```

- Step 6** Configure two interfaces into a pair. You must assign the interface to a virtual sensor and enable it before it can monitor traffic (see Step 10).

```
sensor(config-int-inl)# interface1 GigabitEthernet0/0
sensor(config-int-inl)# interface2 GigabitEthernet0/1
```

- Step 7** Add a description of the interface pair.

```
sensor(config-int-inl)# description PAIR1 Gig0/0 and Gig0/1
```

- Step 8** Repeat Steps 4 through 7 for any other interfaces that you want to configure into inline interface pairs.

- Step 9** Verify the settings.

```
sensor(config-int-inl)# show settings
name: PAIR1

description: PAIR1 Gig0/0 & Gig0/1 default:
interface1: GigabitEthernet0/0
interface2: GigabitEthernet0/1

```

- Step 10** Enable the interfaces assigned to the interface pair.

```
sensor(config-int)# exit
```



```

sensor(config-int)# physical-interfaces GigabitEthernet0/0
sensor(config-int-phy)# admin-state enabled
sensor(config-int-phy)# exit
sensor(config-int)# physical-interfaces GigabitEthernet0/1
sensor(config-int-phy)# admin-state enabled
sensor(config-int-phy)# exit
sensor(config-int)#

```

**Step 11** Verify that the interfaces are enabled.

```

sensor(config-int)# show settings
 physical-interfaces (min: 0, max: 999999999, current: 5)

 <protected entry>
 name: GigabitEthernet0/0

 media-type: tx <protected>
 description: <defaulted>
 admin-state: enabled default: disabled
 duplex: auto <defaulted>
 speed: auto <defaulted>
 default-vlan: 0 <defaulted>
 alt-tcp-reset-interface

 none

 subinterface-type

 none

 <protected entry>
 name: GigabitEthernet0/1

 media-type: tx <protected>
 description: <defaulted>
 admin-state: enabled default: disabled
 duplex: auto <defaulted>
 speed: auto <defaulted>
 default-vlan: 0 <defaulted>
 alt-tcp-reset-interface

 none

 subinterface-type

 none

 <protected entry>
 name: GigabitEthernet0/2 <defaulted>

 media-type: tx <protected>
 description: <defaulted>
 admin-state: disabled <defaulted>
 duplex: auto <defaulted>

```

```

speed: auto <defaulted>
default-vlan: 0 <defaulted>
alt-tcp-reset-interface

none

subinterface-type

none

<protected entry>
name: GigabitEthernet0/3 <defaulted>

media-type: tx <protected>
--MORE--

```

- Step 12** Delete an inline interface pair and return the interfaces to promiscuous mode. You must also delete the inline interface pair from the virtual sensor to which it is assigned.

```
sensor(config-int)# no inline-interfaces PAIR1
```

- Step 13** Verify the inline interface pair has been deleted.

```

sensor(config-int)# show settings

command-control: Management0/0 <protected>
inline-interfaces (min: 0, max: 999999999, current: 0)

bypass-mode: auto <defaulted>
interface-notifications

```

- Step 14** Exit interface configuration submenu.

```

sensor(config-int)# exit
Apply Changes?[yes]:

```

- Step 15** Press **Enter** to apply the changes or enter **no** to discard them.

#### For More Information

- For the procedure for configuring inline interface mode for the ASA 5500-X IPS SSP, see [Assigning Virtual Sensors to Adaptive Security Appliance Contexts, page 18-6](#).
- For the procedure for configuring inline interface mode for the ASA 5585-X IPS SSP, see [Assigning Virtual Sensors to Adaptive Security Appliance Contexts, page 19-7](#).

For the procedure for assigning inline interface pairs to a virtual sensor, or deleting the inline interface pair from the virtual sensor to which it is assigned, see [Adding, Editing, and Deleting Virtual Sensors, page 6-5](#).

## Configuring Inline VLAN Pair Mode

This section describes inline VLAN pair mode and how to configure inline VLAN pairs. It contains the following topics:

- [Understanding Inline VLAN Pair Mode, page 5-21](#)
- [Configuring Inline VLAN Pairs, page 5-22](#)

## Understanding Inline VLAN Pair Mode



**Note**

The ASA IPS modules (ASA 5500-X IPS SSP and ASA 5585-X IPS SSP) do not support inline VLAN pairs.



**Note**

For the IPS 4500 series sensors, the maximum number of inline VLAN pairs you can create system wide is 150. On all other platforms, the limit is 255 per interface.

You can associate VLANs in pairs on a physical interface. This is known as inline VLAN pair mode. Packets received on one of the paired VLANs are analyzed and then forwarded to the other VLAN in the pair.

Inline VLAN pair mode is an active sensing mode where a sensing interface acts as an 802.1q trunk port, and the sensor performs VLAN bridging between pairs of VLANs on the trunk. The sensor inspects the traffic it receives on each VLAN in each pair, and can either forward the packets on the other VLAN in the pair, or drop the packet if an intrusion attempt is detected. You can configure an IPS sensor to simultaneously bridge up to 255 VLAN pairs on each sensing interface. The sensor replaces the VLAN ID field in the 802.1q header of each received packet with the ID of the egress VLAN on which the sensor forwards the packet. The sensor drops all packets received on any VLANs that are not assigned to inline VLAN pairs.

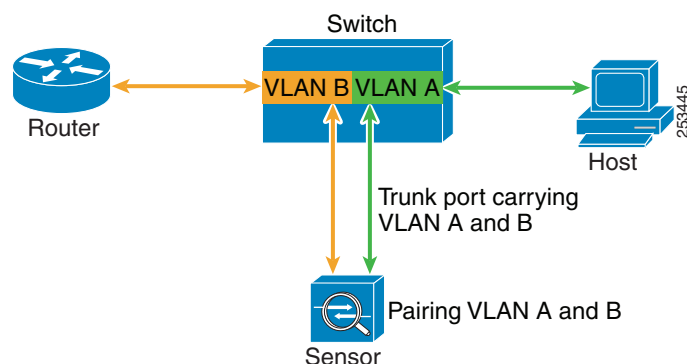


**Note**

You cannot use the default VLAN as one of the paired VLANs in an inline VLAN pair.

[Figure 5-3](#) illustrates inline VLAN pair mode:

**Figure 5-3** *Inline VLAN Pair Mode*



## Configuring Inline VLAN Pairs

Use the **physical-interfaces** *interface\_name* command in the service interface submode to configure inline VLAN pairs. The interface name is FastEthernet or GigabitEthernet.

The following commands apply:

- **admin-state {enabled | disabled}**—Specifies the administrative link state of the interface, whether the interface is enabled or disabled.




---

**Note** On all backplane sensing interfaces on all modules, **admin-state** is set to enabled and is protected (you cannot change the setting). The **admin-state** has no effect (and is protected) on the command and control interface. It only affects sensing interfaces. The command and control interface does not need to be enabled because it cannot be monitored.

---

- **default**—Sets the value back to the system default setting.
- **description**—Specifies the description of the interface.
- **duplex**—Specifies the duplex setting of the interface:
  - **auto**—Sets the interface to auto negotiate duplex.
  - **full**—Sets the interface to full duplex.
  - **half**—Sets the interface to half duplex.




---

**Note** The **duplex** option is protected on all modules.

---

- **no**—Removes an entry or selection setting.
- **speed**—Specifies the speed setting of the interface:
  - **auto**—Sets the interface to auto negotiate speed.
  - **10**—Sets the interface to 10 MB (for TX interfaces only).
  - **100**—Sets the interface to 100 MB (for TX interfaces only).
  - **1000**—Sets the interface to 1 GB (for Gigabit interfaces only).




---

**Note** The **speed** option is protected on all modules.

---

- **subinterface-type**—Specifies that the interface is a subinterface and what type of subinterface is defined.
  - **inline-vlan-pair**—Lets you define the subinterface as an inline VLAN pair.
  - **none**—No subinterfaces defined.
- **subinterface name**—Defines the subinterface as an inline VLAN pair:
  - **vlan1**—Specifies the first VLAN in the inline VLAN pair.
  - **vlan2**—Specifies the second VLAN in the inline VLAN pair.

### Configuring Inline VLAN Pairs

To configure the inline VLAN pair settings on the sensor, follow these steps:

**Step 1** Log in to the CLI using an account with administrator privileges.

**Step 2** Enter interface submode.

```
sensor# configure terminal
sensor(config)# service interface
sensor(config-int)#
```

**Step 3** Verify if any inline interfaces exist (the subinterface type should read “none” if no inline interfaces have been configured).

```
sensor(config-int)# show settings
physical-interfaces (min: 0, max: 999999999, current: 5)

<protected entry>
name: GigabitEthernet0/0 <defaulted>

media-type: tx <protected>
description: <defaulted>
admin-state: disabled <defaulted>
duplex: auto <defaulted>
speed: auto <defaulted>
alt-tcp-reset-interface

none

subinterface-type

none

<protected entry>
name: GigabitEthernet0/1 <defaulted>

media-type: tx <protected>
description: <defaulted>
admin-state: disabled <defaulted>
duplex: auto <defaulted>
speed: auto <defaulted>
alt-tcp-reset-interface

none

subinterface-type

none

<protected entry>
name: GigabitEthernet0/2 <defaulted>

media-type: tx <protected>
```

```

description: <defaulted>
admin-state: disabled <defaulted>
duplex: auto <defaulted>
speed: auto <defaulted>
alt-tcp-reset-interface

 none

subinterface-type

 none

<protected entry>
name: GigabitEthernet0/3 <defaulted>

media-type: tx <protected>
description: <defaulted>
admin-state: disabled <defaulted>
duplex: auto <defaulted>
speed: auto <defaulted>
alt-tcp-reset-interface

 none

subinterface-type

 none

<protected entry>
name: Management0/0 <defaulted>

media-type: tx <protected>
description: <defaulted>
admin-state: disabled <protected>
duplex: auto <defaulted>
speed: auto <defaulted>
alt-tcp-reset-interface

 none

subinterface-type

 none

command-control: Management0/0 <protected>
inline-interfaces (min: 0, max: 999999999, current: 0)


```

```

bypass-mode: auto <defaulted>
interface-notifications

missed-percentage-threshold: 0 percent <defaulted>
notification-interval: 30 seconds <defaulted>
idle-interface-delay: 30 seconds <defaulted>

sensor(config-int)#

```

- Step 4** If there are inline interfaces that are using this physical interface, remove them. You must also delete the inline interface from the virtual sensor to which it is assigned.

```
sensor(config-int)# no inline-interfaces interface_name
```

- Step 5** Display the list of available interfaces.

```

sensor(config-int)# physical-interfaces ?
GigabitEthernet0/0 GigabitEthernet0/0 physical interface.
GigabitEthernet0/1 GigabitEthernet0/1 physical interface.
GigabitEthernet0/2 GigabitEthernet0/2 physical interface.
GigabitEthernet0/3 GigabitEthernet0/3 physical interface.
Management0/0 Management0/0 physical interface.
sensor(config-int)# physical-interfaces

```

- Step 6** Designate an interface.

```
sensor(config-int)# physical-interfaces GigabitEthernet0/2
```

- Step 7** Enable the interface. You must assign the interface to a virtual sensor and enable it before it can monitor traffic.

```
sensor(config-int-phy)# admin-state enabled
```

- Step 8** Add a description of this interface.

```
sensor(config-int-phy)# description INT1
```

- Step 9** Configure the duplex settings. This option is not available on the ASA IPS modules (ASA 5500-X IPS SSP or ASA 5585-X IPS SSP).

```
sensor(config-int-phy)# duplex full
```

- Step 10** Configure the speed. This option is not available on the ASA IPS modules (ASA 5500-X IPS SSP or ASA 5585-X IPS SSP).

```
sensor(config-int-phy)# speed 1000
```

- Step 11** Set up the inline VLAN pair.

```

sensor(config-int-phy)# subinterface-type inline-vlan-pair
sensor(config-int-phy-inl)# subinterface 1
sensor(config-int-phy-inl-sub)# vlan1 52
sensor(config-int-phy-inl-sub)# vlan2 53

```

- Step 12** Add a description for the inline VLAN pair.

```
sensor(config-int-phy-inl-sub)# description INT1 vlans 52 and 53
```

- Step 13** Verify the inline VLAN pair settings.

```

sensor(config-int-phy-inl-sub)# show settings
subinterface-number: 1

description: INT1 vlans 52 and 53 default:
vlan1: 52
vlan2: 53

```

```

sensor(config-int-phy-inl-sub) #
```

**Step 14** To delete VLAN pairs:

- a. Delete one VLAN pair.

```
sensor(config-int-phy-inl-sub) # exit
sensor(config-int-phy-inl) # no subinterface 1
```

If this VLAN pair is the last one on the sensor, you receive the following error message:

```
Error: This "subinterface-type" contains less than the required number of
"subinterface" entries. Please add entry(s) to reach the minimum required entries or
select a different "subinterface-type".
```

Go to Step b to remove the last VLAN pair.

- b. Delete all VLAN pairs.

```
sensor(config-int-phy-inl-sub) # exit
sensor(config-int-phy-inl) # exit
sensor(config-int-phy) # subinterface-type none
```

**Step 15** Exit interface submenu. You must also delete the interface from the virtual sensor to which it is assigned.

```
sensor(config-int-phy-inl-sub) # exit
sensor(config-int-phy-inl) # exit
sensor(config-int-phy) # exit
sensor(config-int) # exit
Apply Changes:[yes]:
```

**Step 16** Press **Enter** to apply the changes or enter **no** to discard them.

#### For More Information

For the procedure for assigning inline interface pairs to a virtual sensor, or deleting the inline interface pair from the virtual sensor to which it is assigned, see [Adding, Editing, and Deleting Virtual Sensors](#), page 6-5.

## Configuring VLAN Group Mode

This section describes VLAN Group mode and how to configure VLAN groups. It contains the following topics:

- [Understanding VLAN Group Mode](#), page 5-26
- [Deploying VLAN Groups](#), page 5-27
- [Configuring VLAN Groups](#), page 5-28

## Understanding VLAN Group Mode



#### Note

The ASA IPS modules (ASA 5500-X IPS SSP and ASA 5585-X IPS SSP) do not support VLAN groups mode.



You can divide each physical interface or inline interface into VLAN group subinterfaces, each of which consists of a group of VLANs on that interface. Analysis Engine supports multiple virtual sensors, each of which can monitor one or more of these interfaces. This lets you apply multiple policies to the same sensor. The advantage is that now you can use a sensor with only a few interfaces as if it had many interfaces.

**Note**

---

You cannot divide physical interfaces that are in inline VLAN pairs into VLAN groups.

---

VLAN group subinterfaces associate a set of VLANs with a physical or inline interface. No VLAN can be a member of more than one VLAN group subinterface. Each VLAN group subinterface is identified by a number between 1 and 255. Subinterface 0 is a reserved subinterface number used to represent the entire unvirtualized physical or logical interface. You cannot create, delete, or modify subinterface 0 and no statistics are reported for it.

An unassigned VLAN group is maintained that contains all VLANs that are not specifically assigned to another VLAN group. You cannot directly specify the VLANs that are in the unassigned group. When a VLAN is added to or deleted from another VLAN group subinterface, the unassigned group is updated.

Packets in the native VLAN of an 802.1q trunk do not normally have 802.1q encapsulation headers to identify the VLAN number to which the packets belong. A default VLAN variable is associated with each physical interface and you should set this variable to the VLAN number of the native VLAN or to 0. The value 0 indicates that the native VLAN is either unknown or you do not care if it is specified. If the default VLAN setting is 0, the following occurs:

- Any alerts triggered by packets without 802.1q encapsulation have a VLAN value of 0 reported in the alert.
- Non-802.1q encapsulated traffic is associated with the unassigned VLAN group and it is not possible to assign the native VLAN to any other VLAN group.

**Note**

---

You can configure a port on a switch as either an access port or a trunk port. On an access port, all traffic in a single VLAN is called the access VLAN. On a trunk port, multiple VLANs can be carried over the port, and each packet has a special header attached called the 802.1q header that contains the VLAN ID. This header is commonly referred as the VLAN tag. However, a trunk port has a special VLAN called the native VLAN. Packets in the native VLAN do not have the 802.1q headers attached.

---

## Deploying VLAN Groups

Because a VLAN group of an inline pair does not translate the VLAN ID, an inline paired interface must exist between two switches to use VLAN groups on a logical interface. For an appliance, you can connect the two pairs to the same switch, make them access ports, and then set the access VLANs for the two ports differently. In this configuration, the sensor connects between two VLANs, because each of the two ports is in access mode and carries only one VLAN. In this case the two ports must be in different VLANs, and the sensor bridges the two VLANs, monitoring any traffic that flows between the two VLANs.

You can also connect appliances between two switches. There are two variations. In the first variation, the two ports are configured as access ports, so they carry a single VLAN. In this way, the sensor bridges a single VLAN between the two switches.

In the second variation, the two ports are configured as trunk ports, so they can carry multiple VLANs. In this configuration, the sensor bridges multiple VLANs between the two switches. Because multiple VLANs are carried over the inline interface pair, the VLANs can be divided into groups and each group can be assigned to a virtual sensor.

## Configuring VLAN Groups

Use the **physical-interfaces** *interface\_name* command in the service interface submode to configure inline VLAN groups. The interface name is FastEthernet or GigabitEthernet.

The following commands apply:

- **admin-state {enabled | disabled}**—Specifies the administrative link state of the interface, whether the interface is enabled or disabled.




---

**Note** On all backplane sensing interfaces on all modules, **admin-state** is set to enabled and is protected (you cannot change the setting). The **admin-state** has no effect (and is protected) on the command and control interface. It only affects sensing interfaces. The command and control interface does not need to be enabled because it cannot be monitored.

---

- **default**—Sets the value back to the system default setting.
- **description**—Specifies the description of the interface.
- **duplex**—Specifies the duplex setting of the interface:
  - **auto**—Sets the interface to auto negotiate duplex.
  - **full**—Sets the interface to full duplex.
  - **half**—Sets the interface to half duplex.




---

**Note** The **duplex** option is protected on all modules.

---

- **no**—Removes an entry or selection setting.
- **speed**—Specifies the speed setting of the interface:
  - **auto**—Sets the interface to auto negotiate speed.
  - **10**—Sets the interface to 10 MB (for TX interfaces only).
  - **100**—Sets the interface to 100 MB (for TX interfaces only).
  - **1000**—Sets the interface to 1 GB (for Gigabit interfaces only).




---

**Note** The **speed** option is protected on all modules.

---

- **subinterface-type**—Specifies that the interface is a subinterface and what type of subinterface is defined.
  - **vlan-group**—Lets you define the subinterface as a VLAN group.
  - **none**—Specifies that no subinterfaces are defined.
- **subinterface name**—Defines the subinterface as a VLAN group:

- **vlan** {**range** | **unassigned**}—Specifies the set of VLANs in the VLAN group. The value for **range** is 1 to 4095 in a comma-separated pattern of individual VLAN IDs or ranges: 1,5-8,10-15. There are no spaces between the entries.

### Configuring Inline VLAN Groups

To configure the inline VLAN group settings on the sensor, follow these steps:

**Step 1** Log in to the CLI using an account with administrator privileges.

**Step 2** Enter interface submode.

```
sensor# configure terminal
sensor(config)# service interface
sensor(config-int)#
```

**Step 3** Verify if any inline interfaces exist (the subinterface type should read “none” if no inline interfaces have been configured).

```
sensor(config-int)# show settings
physical-interfaces (min: 0, max: 999999999, current: 5)

<protected entry>
name: GigabitEthernet0/0 <defaulted>

media-type: tx <protected>
description: <defaulted>
admin-state: disabled <defaulted>
duplex: auto <defaulted>
speed: auto <defaulted>
alt-tcp-reset-interface

none

subinterface-type

none

<protected entry>
name: GigabitEthernet0/1 <defaulted>

media-type: tx <protected>
description: <defaulted>
admin-state: disabled <defaulted>
duplex: auto <defaulted>
speed: auto <defaulted>
alt-tcp-reset-interface

none

subinterface-type

none


```

```

<protected entry>
name: GigabitEthernet0/2 <defaulted>

 media-type: tx <protected>
 description: <defaulted>
 admin-state: disabled <defaulted>
 duplex: auto <defaulted>
 speed: auto <defaulted>
 alt-tcp-reset-interface

 none

 subinterface-type

 none

<protected entry>
name: GigabitEthernet0/3 <defaulted>

 media-type: tx <protected>
 description: <defaulted>
 admin-state: disabled <defaulted>
 duplex: auto <defaulted>
 speed: auto <defaulted>
 alt-tcp-reset-interface

 none

 subinterface-type

 none

<protected entry>
name: Management0/0 <defaulted>

 media-type: tx <protected>
 description: <defaulted>
 admin-state: disabled <protected>
 duplex: auto <defaulted>
 speed: auto <defaulted>
 alt-tcp-reset-interface

 none

 subinterface-type

 none


```

```

command-control: Management0/0 <protected>
inline-interfaces (min: 0, max: 999999999, current: 0)

bypass-mode: auto <defaulted>
interface-notifications

missed-percentage-threshold: 0 percent <defaulted>
notification-interval: 30 seconds <defaulted>
idle-interface-delay: 30 seconds <defaulted>

sensor(config-int)#

```

- Step 4** If there are inline interfaces that are using this physical interface, remove them.

```
sensor(config-int)# no inline-interfaces interface_name
```

- Step 5** Display the list of available interfaces.

```

sensor(config-int)# physical-interfaces ?
GigabitEthernet0/0 GigabitEthernet0/0 physical interface.
GigabitEthernet0/1 GigabitEthernet0/1 physical interface.
GigabitEthernet0/2 GigabitEthernet0/2 physical interface.
GigabitEthernet0/3 GigabitEthernet0/3 physical interface.
Management0/0 Management0/0 physical interface.
sensor(config-int)# physical-interfaces

```

- Step 6** Specify an interface.

```
sensor(config-int)# physical-interfaces GigabitEthernet0/2
```

- Step 7** Enable the interface. You must also assign the interface to a virtual sensor and enable it before it can monitor traffic.

```
sensor(config-int-phy)# admin-state enabled
```

- Step 8** Add a description of this interface.

```
sensor(config-int-phy)# description INT1
```

- Step 9** Specify the duplex settings. This option is not available on the ASA IPS modules (ASA 5500-X IPS SSP or ASA 5585-X IPS SSP).

```
sensor(config-int-phy)# duplex full
```

- Step 10** Specify the speed. This option is not available on the ASA IPS modules (ASA 5500-X IPS SSP or ASA 5585-X IPS SSP).

```
sensor(config-int-phy)# speed 1000
```

- Step 11** Set up the VLAN group.

```

sensor(config-int-phy)# subinterface-type vlan-group
sensor(config-int-phy-vla)# subinterface 1

```

- Step 12** Assign the VLANs to this group:

- a. Assign specific VLANs.

```

sensor(config-int-phy-vla-sub)# vlans range 1,5-8,10-15
sensor(config-int-phy-vla-sub)#

```

- b. Verify the settings.

```
sensor(config-int-phy-vla-sub)# show settings
```

```

subinterface-number: 1

description: <defaulted>
vlans

range: 1,5-8,10-15

sensor(config-int-phy-vla-sub) #

```

c. Configure unassigned VLANs.

```

sensor(config-int-phy-vla-sub) # vlans unassigned
sensor(config-int-phy-vla-sub) #

```

d. Verify the settings.

```

sensor(config-int-phy-vla-sub) # show settings
subinterface-number: 1

description: <defaulted>
vlans

unassigned

sensor(config-int-phy-vla-sub) #

```




---

**Note** Assigning the unassigned VLANs to a separate virtual sensor allows you to specify a policy for all VLANs that you have not specifically assigned to other groups. For example, you can group your important internal VLANs in one group and apply a stringent security policy to that group. You can group the other less important unassigned VLANs into another group, and apply the default security policy to that group, so that only very serious alerts are reported.

---

**Step 13** Add a description for the VLAN group.

```

sensor(config-int-phy-inl-sub) # description INT1 vlans 52 and 53

```

**Step 14** Verify the VLAN group settings.

```

sensor(config-int-phy-vla-sub) # show settings
subinterface-number: 1

description: GROUP1 default:
vlans

unassigned

sensor(config-int-phy-vla-sub) #

```

**Step 15** Delete VLAN groups:

a. Delete one VLAN group.

```

sensor(config-int-phy-vla-sub) # exit
sensor(config-int-phy-vla) # no subinterface 1

```

If this VLAN group is the last one on the sensor, you receive an error message.

```
Error: This "subinterface-type" contains less than the required number of
"subinterface" entries. Please add entry(s) to reach the minimum required entries or
select a different "subinterface-type".
```

Go to Step b to remove the last VLAN group.

- b. Delete all VLAN groups. You must also delete the VLAN group from the virtual sensor to which it is assigned.

```
sensor(config-int-phy-vla-sub)# exit
sensor(config-int-phy-vla)# exit
sensor(config-int-phy)# subinterface-type none
```

- Step 16** Exit interface submode.

```
sensor(config-int-phy-vla-sub)# exit
sensor(config-int-phy-vla)# exit
sensor(config-int-phy)# exit
sensor(config-int)# exit
Apply Changes?[yes]:
```

- Step 17** Press **Enter** to apply the changes or enter **no** to discard them.

#### For More Information

For the procedure for assigning inline interface pairs to a virtual sensor, or deleting the inline interface pair from the virtual sensor to which it is assigned, see [Adding, Editing, and Deleting Virtual Sensors](#), page 6-5.

## Configuring LACP for the 4500 Series Sensors

This section describes how to configure LACP for the 4500 series sensors, and contains the following topics:

- [Understanding ECLB Using LACP](#), page 5-33
- [LACP Restrictions](#), page 5-35
- [Understanding Failover/Fallback](#), page 5-36
- [LACP Link States](#), page 5-36
- [Configuring LACP](#), page 5-36
- [Displaying LACP](#), page 5-38

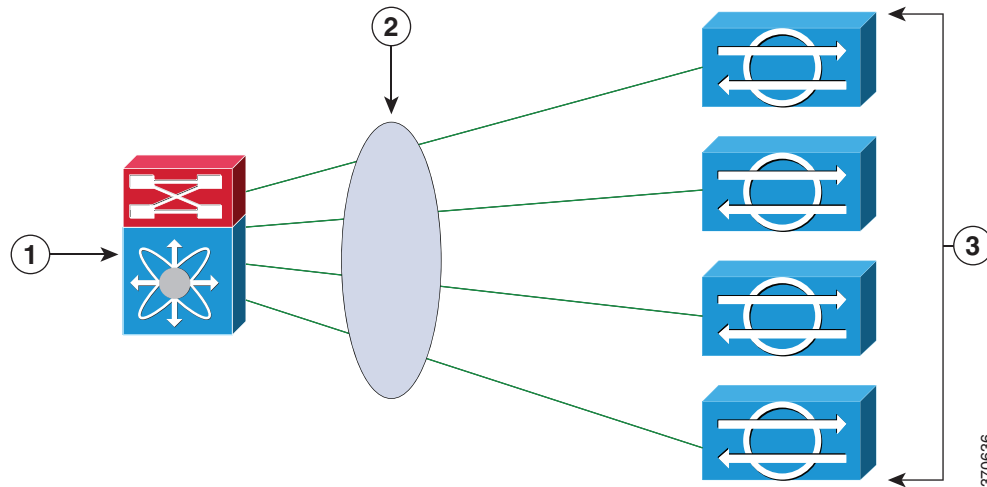
## Understanding ECLB Using LACP

Link Aggregation Control Protocol (LACP) support has been added to the IPS 4500 series sensors to meet the scalability and high availability requirements of the data center. The capability of network switches is leveraged to aggregate multiple links in a single port channel. For example, using EtherChannel, LACP can scale aggregated IPS throughput to a maximum of up to 80 Gbps with 16

IPS 4520 in an LACP group. LACP with IPS interoperates with Catalyst 6K and Nexus 7K switches in data center environments. ECLB using LACP in the Nexus 7K supports up to 16 IPS devices; with the Catalyst 6K it supports 8 IPS devices.

Figure 5-4 shows a port channel configuration that consists of a switch that is connected to four IPS sensors to achieve scalability and increase bandwidth.

**Figure 5-4** IPS Port Channel Configuration with a Switch



1	Switch S1	2	Port Channel
3	IPS		

A port channel is created on the switch and the four physical interfaces that are connected to the IPS sensors are added to the port channel. The port channel supports load balancing to distribute the load among the members of the port channel. IPS sensors in an LACP port channel act as independent appliances. We recommend that you have the same configuration on all IPS sensors participating in the LACP port channel. You must configure each of the IPS interfaces in the LACP port channel in inline VLAN pair mode.

LACP is a point-to-point protocol and has two modes of operation—active and passive. Both are supported by the IPS. In active mode, Link Aggregation Control Protocol Data Units (LACPDU)s are periodically sent to actively probe for LACP support on the device on the other side. If the device on the other side responds to the LACP packets, an LACP connection is initiated. If LACP mode is configured as active, the connection is initiated as soon as the physical link is up.

In passive mode, the device does not actively send any LACP packets to probe the LACP device on the other end. It waits for the other side to probe and initiate an LACP connection. This can work only if the device on the other side supports LACP and is configured in active mode.

By default, LACP mode in IPS is disabled. Once LACP is enabled in the IPS, you must have it configured and operational in the switch, otherwise the LACP port will be either suspended or independent based on the switch side configuration.



**Note**

We do not recommend that you enable LACP unless you are sure of the configuration on the other side.

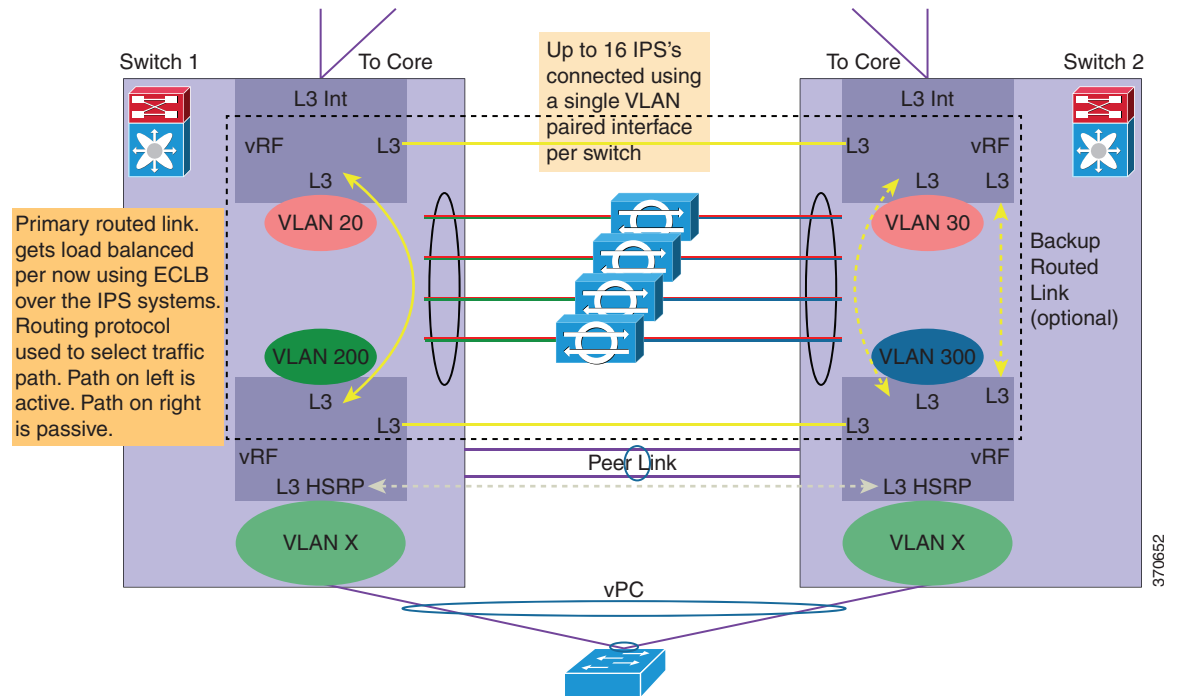




We do not recommend that you configure UDLD with LACP.

Figure 5-5 displays the recommended IPS LACP deployment in a data center environment:

**Figure 5-5 Recommended IPS LACP Configuration**



## LACP Restrictions



We do not recommend that you configure UDLD with LACP.

Pay attention to the following when configuring LACP on the sensor:

- The IPS 4520 is the only platform that supports the dual configuration. You can add another 4520 module to an existing 4520 or you can order the 4520-XL with two modules already installed. Mixing 4510s and 4520s is not a valid configuration.
- Make sure that in your ECLB setup, you do not have multiple links within an EtherChannel going to the same IPS device, because this can lead to a load distribution imbalance.
- When a group of IPS devices participate in an LACP ether channel as one single device, the devices should all have the same system ID. The default system ID ensures this. However, if you have a requirement to have two different IPSes to have different system IDs in order to be able to distinguish them in the show LACP neighbor output, you can configure the LACP system priority, which influences the system ID.

- Make sure that the IPS interface that is part of the same LACP port channel is configured with the same VLAN pair.
- Make sure that bypass mode is off so that IPS can failover and fallback during failure conditions.
- Make sure the TCP session tracking mode is assigned to the virtual sensor, which is the default.

## Understanding Failover/Fallback

The IPS has enhanced the support for seamless failover/fallback of TCP sessions from one sensor to another (nonconnection-based traffic, such as UDP and ICMP already had seamless support). The IPS determines that a gap in the state of the sessions being monitored may have been caused by failover/bypass/link flaps. It intelligently updates its state machine to restart the inspection of the sessions and ensures that the flows do not get dropped.

## LACP Link States

The LACP link state represents whether the link can forward traffic or not and does not represent the actual link state of the physical port. LACP has two link states:

- Up—When the interface is up and the LACP state is either bundled or independent.
- Down (LACP suspended)—When the LACP configuration is mismatched on both ends, which means the LACP state is suspended and the switch does not allow traffic.

The following sequence of event leads to a link being up:

1. At least two sensors are configured with the same channel ID.
2. LACP is configured on both the switch and the sensor.
3. The sensor participating in the same port channel must have the same system priority and channel ID.
4. Interfaces participating in LACP must all have the same duplex and speed configuration.
5. The physical interface is up.

The following sequence of events leads to a link going down.

1. LACP is down.
2. The SensorApp is down.
3. The physical interface is down.

## Configuring LACP



### Note

Make sure that you have LACP configured on a Cisco Nexus 7K or Catalyst 6K switch before configuring LACP on the sensor.

Use these **lACP** commands in service interface submode to configure LACP on the sensor.

The following parameters apply:

- **lACP-node-id** *number*—Adds an LACP node identifier to the sensor. This ID uniquely identifies the node in the LACP group. The range is 1 to 16. The default is 1.

- **lACP-system-priority** *number*—Adds an LACP system priority to the sensor. Make sure the system priority is the same number across all of the nodes in the port channel. The range is 1 to 65535. The default is 32768.
- **physical-interfaces** *interface\_name*—Creates a port channel on a physical interface:
  - **subinterface-type inline-vlan-pair**—Creates an inline VLAN pair.
  - **lACP**—Enters LACP configuration:
    - channel-id** *number*—Creates a port channel identifier. The range is 1 to 255. The default is 1.
    - lACP mode** (**active** | **passive** | **off**)—Enables/disables LACP mode on a physical interface. The default is disabled.

LACP has two modes of operation: active and passive. Because LACP is a point-to-point protocol, in active mode, LACPDU s are periodically sent to actively probe for LACP support on the device other side. If the device on the other side responds to the LACP packets, an LACP connection is initiated. If LACP mode is configured active, this is done as soon as the physical link is up.

In passive mode, the device does not actively send any LACP packets to probe LACP devices on the other end. It waits for the other side to probe and initiate an LACP connection. This can work only if the device on the other side supports LACP and is configured in active mode.

### Configuring LACP and Inline VLAN Pairs on the Sensor

To configure LACP and inline VLAN pairs on the sensor, follow these steps:

- 
- Step 1** Log in to the CLI using an account with administrator privileges.
- Step 2** Enter interface submode.
- ```
IPS-4510# configure terminal
IPS-4510 (config)# service interface
IPS-4510 (config-int)#
```
- Step 3** Assign the LACP node identification to the system. The default is 1.
- ```
IPS-4510 (config-int)# lACP-node-id 4
IPS-4510 (config-int)#
```
- Step 4** Assign the system priority to the system. The default is 32768.
- ```
IPS-4510 (config-int)# lACP-system-priority 25545
IPS-4510 (config-int)#
```
- Step 5** Configure inline VLAN pairs on the sensor.
- ```
IPS-4510 (config-int)# physical-interfaces GigabitEthernet0/0
IPS-4510 (config-int-phy)# admin-state enabled
IPS-4510 (config-int-phy)# subinterface-type inline-vlan-pair
IPS-4510 (config-int-phy-inl)# subinterface 10
IPS-4510 (config-int-phy-inl-sub)# vlan1 3333
IPS-4510 (config-int-phy-inl-sub)# vlan2 3332
IPS-4510 (config-int-phy-inl-sub)# exit
IPS-4510 (config-int-phy-inl)#
```
- Step 6** Add the port channel to a physical interface and enable LACP. The default is disabled.
- ```
IPS-4510 (config-int-phy-inl)# lACP
IPS-4510 (config-int-phy-inl-lACP)# mode active
IPS-4510 (config-int-phy-inl-lACP)# channel-id 10
IPS-4510 (config-int-phy-inl-lACP)# exit
IPS-4510 (config-int-phy-inl)#
```

Step 7 Verify the LACP settings.

```

IPS-4510(config-int-phy-in1)# show settings
inline-vlan-pair
-----
subinterface (min: 1, max: 255, current: 1)
-----
subinterface-number: 1
-----
description: <defaulted>
vlan1: 640
vlan2: 641
-----
lacp
-----
channel-id: 1 <defaulted>
mode: active default: off
-----

```

Step 8 Exit interface submenu.

```

sensor(config-int-phy-in1)# exit
sensor(config-int-phy-in1)# exit
sensor(config-int-phy)# exit
sensor(config-int)# exit
Apply Changes:[yes]:

```

Step 9 Press **Enter** to apply the changes or enter **no** to discard them.

Displaying LACP

Use the **show lacp (neighbors | internals)** command in EXEC mode to display traffic statistics, system identifiers, and neighbor details.

**Note**

The **show interface** and **show interfaces brief** also show the LACP status (susp, indep, and up).

The following parameters apply:

- **neighbors**—Displays system details of the neighbors.
- **internals**—Displays system details of the internals.

The output has the following fields:

- **Interface**—Displays the interface name.
- **Flags**—Displays the flags associated with this interface:
 - **A**—The device is in active mode.
 - **F**—The device is sending fast Link Aggregation Control Protocol Data Units (LACPDU).
 - **S**—The device is sending slow LACPDU.
 - **P**—The device is in passive mode.

- State—Displays the following states:
 - Independent—LACP is configured at the local end and LACP is not configured on a partner/other end, for example, no LACP partner PDU is received.
 - Bundled—LACP is configured on both ends and able to create a bundle successfully; this means the configuration is valid and able to create a bundle.
 - Suspended—LACP is configured on both ends, but the received partner information is invalid' this means there is an invalid configuration on the partner, such as half duplex on the link.
- Port Priority—Displays the port priority in interfaces that have LACP enabled. The range is 1 to 65535 with the higher number signifying a lower priority. The default is 32768. The port priority is only displayed with the port is in active or passive mode.
- Admin Key—Displays the administrative key, which is a 16-bit number used by LACP to manage aggregation. For the IPS, the channel ID is used as the administrative key. All members of the port channel have the same administrative key assigned by the system as the channel ID.
- Oper Key—Displays the operational key, which is a 16-bit number assigned to an interface signifying that it can aggregate with all of the other interfaces that are assigned the same operational key. The operational key matches the administrative key.
- Port Number—Displays the port number, which is a 16-bit number used as the port aggregation priority. The IPS generates a unique port number by concatenating the LACP node identification number with the port number.
- Port State—Displays the state variables of the local/partner port encoded as individual bits within a single octet:
 - LACP_Activity—Encoded in bit 0. This flag indicates the activity control value with regard to this link. Active LACP is encoded as a 1; passive LACP is encoded as a 0.
 - LACP_Timeout—Encoded in bit 1. This flag indicates the timeout control value with regard to this link. Short timeout is encoded as a 1; long timeout is encoded as a 0.
 - Aggregation—Encoded in bit 2. If TRUE (encoded as a 1), this flag indicates that the system considers this link to be suitable for aggregation; this means., it is a potential candidate for aggregation. If FALSE (encoded as a 0), the link is considered to be Individual; this means, this link can be operated only as an individual link.
 - Synchronization—Encoded in bit 3. If TRUE (encoded as a 1), the system considers this link to be IN_SYNC; this means, it has been allocated to the correct Link Aggregation Group, the group has been associated with a compatible Aggregator, and the identity of the Link Aggregation Group is consistent with the system ID and operational key information transmitted. If FALSE (encoded as a 0), then this link is currently OUT_OF_SYNC; this means, it is not in the right aggregation.
 - Collecting—Encoded in bit 4. TRUE (encoded as a 1) means collection of incoming frames on this link is definitely enabled; for example, collection is currently enabled and is not expected to be disabled in the absence of administrative changes or changes in received protocol information. Its value is otherwise FALSE (encoded as a 0).
 - Distributing—Encoded in bit 5. FALSE (encoded as a 0) means distribution of outgoing frames on this link is definitely disabled; for example, distribution is currently disabled and is not expected to be enabled in the absence of administrative changes or changes in received protocol information. Its value is otherwise TRUE (encoded as a 1).
 - Defaulted—Encoded in bit 6. If TRUE (encoded as a 1), this flag indicates that the Actor's receive machine is using defaulted operational partner information, administratively configured for the partner. If FALSE (encoded as a 0), the operational partner information in use has been received in a LACPDU.

- Expired—Encoded in bit 7. If TRUE (encoded as a 1), this flag indicates that the actor's receive machine is in the EXPIRED state; if FALSE (encoded as a 0), this flag indicates that the actor's receive machine is not in the EXPIRED state.



Note The received values of defaulted and expired state are not used by LACP; however, knowing their values can be useful when diagnosing protocol problems.

- System Priority—Displays the system-wide priority setting that is assigned to this interface. It is a 16-bit value with a range of 1 to 65535 and a default of 32768. In most cases, we recommend that you use the default.
- System Mac—Displays the hard-coded System MAC address, which is used across the IPS interfaces to establish the port channel across the IPS.

To display LACP information, follow these steps:

Step 1 Log in to the CLI.

Step 2 Display the LACP information for neighbors.

```
IPS-4520# show lacp neighbors
Flags: S - Device is sending Slow LACPDUs F - Device is sending Fast LACPDUs
      A - Device is in Active mode       P - Device is in Passive mode

channel group 1 neighbors

Partner
Port                               System ID                Port Number              State                    Flags
GigabitEthernet0/0                0x8000,0-19-a9-0-2e-c0  0x922                    bndl                     FP

Partner
Port                               System ID                Port Number              State                    Flags
GigabitEthernet0/0                0x8000,0-19-a9-0-2e-c0  0x922                    bndl                     FP

Partner
Port                               System ID                Port Number              State                    Flags
GigabitEthernet0/0                0x8000,0-19-a9-0-2e-c0  0x922                    bndl                     FP

Partner
Port                               System ID                Port Number              State                    Flags
GigabitEthernet0/0                0x8000,0-19-a9-0-2e-c0  0x922                    bndl                     FP

State                               Port Priority            Oper Key                 Port
32768                               0x32                    0x3e                     0x3e

IPS-4520#
```

Step 3 Show the LACP information for internals.

```
IPS-4520# show lacp internals
Flags: S - Device is sending Slow LACPDUs F - Device is sending Fast LACPDUs
      A - Device is in Active mode       P - Device is in Passive mode

Node Information:
-----
Node Identification Number: 1
System Priority: 32768

Global Event Queue Information:
-----
No of Events: 0

Interfaces Information:
-----
Interface Name: GigabitEthernet0/0
-----
ChannelId: 1
Port State: bndl
Port Mode: LACP/Active
```

```

Counter Information:
-----
TX CNT: 15277
RX CNT: 553
MARKER TX CNT: 0
MARKER RX CNT: 0
MARKER ResTX CNT: 0
MARKER ResRX CNT: 0
Internal Mem Stats:
-----
RX Alloc: 553
RX Free: 553
TX Alloc: 15277
TX Free: 15277
Link UP/DOWN counters:
-----
Link UP events: 1
Link DOWN events: 0

Local Information:
-----
Flags: SA
Lacp State: bndl
Port Priority: 32768
Oper Key: 0x1
Port Num: 0x105
Port State: 0x3d

Partner Information:
-----
Flags: FP
Lacp State: bndl
Port Priority: 32768
Oper Key: 0x32
Port Num: 0x922
Port State: 0x3e
State Machine Information:
-----
TxSM: tx_p
RxSM: curr
PeriodicSM: fast
MuxSM: col_dis
TxState: TRANSMIT_PDU(tt_expired)-> WAITED(ntt)-> TRANSMIT_PDU(tt_expired)->
WAITED(ntt)-> TRANSMIT_PDU(tt_expired)-> WAITED(ntt)-> TRANSMIT_PDU(tt_expired)->
WAITED(ntt)-> TRANSMIT_PDU(tt_expired)-> WAITED(ntt)
RxState: CURRENT(recv_lacpdu)-> CURRENT(recv_lacpdu)-> CURRENT(recv_lacpdu)->
CURRENT(recv_lacpdu)-> CURRENT(recv_lacpdu)-> CURRENT(recv_lacpdu)->
CURRENT(recv_lacpdu)-> CURRENT(recv_lacpdu)-> CURRENT(recv_lacpdu)-> CURRENT(recv_lacpdu)
PtxState: FAST_PERIODIC(pt_expired)-> PERIODIC_TX(short_timeout)->
FAST_PERIODIC(pt_expired)-> PERIODIC_TX(short_timeout)-> FAST_PERIODIC(pt_expired)->
PERIODIC_TX(short_timeout)-> FAST_PERIODIC(pt_expired)-> PERIODIC_TX(short_timeout)->
FAST_PERIODIC(pt_expired)-> PERIODIC_TX(short_timeout)
MuxState: COLLECTING_DISTRIBUTING(in_sync)-> COLLECTING_DISTRIBUTING(in_sync)->
COLLECTING_DISTRIBUTING(in_sync)-> COLLECTING_DISTRIBUTING(in_sync)->
COLLECTING_DISTRIBUTING(in_sync)-> COLLECTING_DISTRIBUTING(in_sync)->
COLLECTING_DISTRIBUTING(in_sync)-> COLLECTING_DISTRIBUTING(in_sync)->
COLLECTING_DISTRIBUTING(in_sync)-> COLLECTING_DISTRIBUTING(in_sync)
LAG Information:
-----
State: partner_up

```

```
LAG ID:
(8000,02-49-50-53-04-05,0001,8000,0105), (8000,00-19-a9-00-2e-c0,0032,8000,0922)
IPS-4520#
```

Configuring Inline Bypass Mode

This section describes inline bypass mode for sensors configured as inline interface and inline VLAN pairs, and contains the following topics:

- [Understanding Inline Bypass Mode, page 5-42](#)
- [Configuring Inline Bypass Mode, page 5-43](#)

Understanding Inline Bypass Mode



Note

The ASA 5500-X IPS SSP and ASA 5585-X IPS SSP do not support bypass mode. The adaptive security appliance will either fail open, fail close, or fail over depending on the configuration of the adaptive security appliance and the type of activity being done on the IPS.



Caution

There are security consequences when you put the sensor in bypass mode. When bypass mode is on, the traffic bypasses the sensor and is not inspected; therefore, the sensor cannot prevent malicious attacks.



Caution

As with signature updates, when the sensor applies a global correlation update, it may trigger bypass. Whether or not bypass is triggered depends on the traffic load of the sensor and the size of the signature/global correlation update. If bypass mode is turned off, an inline sensor stops passing traffic while the update is being applied.

You can use inline bypass as a diagnostic tool and a failover protection mechanism. Normally, the sensor Analysis Engine performs packet analysis. When inline bypass is activated, the Analysis Engine is bypassed, allowing traffic to flow through the inline interfaces and inline VLAN pairs without inspection. Inline bypass ensures that packets continue to flow through the sensor when the sensor processes are temporarily stopped for upgrades or when the sensor monitoring processes fail. There are three modes: on, off, and automatic. By default, bypass mode is set to automatic.

The inline bypass functionality is implemented in software, so it only functions when the operating system is running. If the sensor is powered off or shut down, inline bypass does not work—traffic does not flow through the sensor.

For IPS 4510 and IPS 4520, when the SensorApp is not running or if bypass mode is on, the following occurs:

- The output from the **packet capture/display** command does not show any packets.
- The **show interface** and **show interface *interface_name*** commands do not show VLAN statistics.

Configuring Inline Bypass Mode

Use the **bypass-mode** command in the service interface submode to configure bypass mode. The following commands apply:

- **off**—Turns off inline bypassing. Packet inspection is performed on inline data traffic. However, inline traffic is interrupted if the Analysis Engine is stopped.
- **on**—Turns on inline bypassing. No packet inspection is performed on the traffic. Inline traffic continues to flow even if the Analysis Engine is stopped.
- **auto**—Turns on automatic bypassing. The sensor automatically begins bypassing inline packet inspection if the Analysis Engine stops processing packets. This prevents data interruption on inline interfaces. This is the default.

Configuring Bypass Mode

To configure bypass mode, follow these steps:

Step 1 Log in to the CLI using an account with administrator privileges.

Step 2 Enter interface submode.

```
sensor# configure terminal
sensor(config)# service interface
```

Step 3 Configure bypass mode.

```
sensor(config-int)# bypass-mode off
```

Step 4 Verify the settings.

```
sensor(config-int)# show settings
-----
bypass-mode: off default: auto
interface-notifications
-----
missed-percentage-threshold: 0 percent <defaulted>
notification-interval: 30 seconds <defaulted>
idle-interface-delay: 30 seconds <defaulted>
-----
sensor(config-int)#
```

Step 5 Exit interface submode.

```
sensor(config-int)# exit
Apply Changes:[yes]:
```

Step 6 Press **Enter** to apply the changes or enter **no** to discard them.

For More Information

For more information on inline bypass mode, see [Configuring Inline Bypass Mode, page 5-42](#).

Configuring Interface Notifications

You can configure the sensor to monitor the flow of packets across an interface and send notification if that flow changes (starts/stops) during a specified interval. You can configure the missed packet threshold within a specific notification interval and also configure the interface idle delay before a status event is reported.

Use the **interface-notifications** command in the service interface submode to configure traffic notifications.

The following parameters apply:

- **default**—Sets the value back to the system default setting.
- **idle-interface-delay**—Specifies the number of seconds an interface must be idle before sending a notification. The valid range is 5 to 3600. The default is 30 seconds.
- **missed-percentage-threshold**—Specifies the percentage of packets that must be missed during a specified interval before notification will be sent. The valid range is 0 to 100. The default is 0.
- **notification-interval**—Specifies the interval to check for missed packet percentage. The valid range is 5 to 3600. The default is 30 seconds

Configuring Interface Notifications

To configure the interface notification settings, follow these steps:

Step 1 Log in to the CLI using an account with administrator privileges.

Step 2 Enter global configuration mode.

```
sensor# configure terminal
```

Step 3 Enter interface submode.

```
sensor(config)# service interface
```

Step 4 Enter interface notifications submode.

```
sensor(config-int)# interface-notifications
```

Step 5 Specify the idle interface delay.

```
sensor(config-int-int)# idle-interface-delay 60
```

Step 6 Specify the missed percentage threshold.

```
sensor(config-int-int)# missed-percentage-threshold 1
```

Step 7 Specify the notification interval.

```
sensor(config-int-int)# notification-interval 60
```

Step 8 Verify the settings.

```
sensor(config-int-int)# show settings
interface-notifications
-----
missed-percentage-threshold: 1 percent default: 0
notification-interval: 60 seconds default: 30
idle-interface-delay: 60 seconds default: 30
-----
sensor(config-int-int)#
```

Step 9 Exit interface notifications submode.

```
sensor(config-int-int)# exit
sensor(config-int)# exit
Apply Changes:[yes]:
```

Step 10 Press **Enter** to apply the changes or enter **no** to discard them.

Configuring CDP Mode



Note

The ASA 5500-X IPS SSP and ASA 5585-X IPS SSP do not support CDP mode.

You can configure the sensor to enable or disable the forwarding of CDP packets. This action applies globally to all interfaces.

Cisco Discovery Protocol is a media- and protocol-independent device-discovery protocol that runs on all Cisco-manufactured equipment, including routers, access servers, bridges, and switches. Using CDP, a device can advertise its existence to other devices and receive information about other devices on the same LAN or on the remote side of a WAN. CDP runs on all media that support SNAP, including LANs, Frame Relay, and ATM media.

User the **cdp-mode** command in service interface mode to have the sensor either forward or drop CDP packets.

The following parameter applies:

- **cdp-mode {forward-cdp-packets | drop-cdp-packets}**—Configures the sensor to either forward CDP packets or drop CDP packets. The default is drop-cdp-packets.

Enabling CDP Mode

To configure CDP mode, follow these steps:

Step 1 Log in to the CLI using an account with administrator privileges.

Step 2 Enter interface submode.

```
sensor# configure terminal
sensor(config)# service interface
```

Step 3 Enable CDP mode.

```
sensor(config-int)# cdp-mode forward-cdp-packets
```

Step 4 Verify the settings.

```
sensor(config-int)# show settings
-----
bypass-mode: auto <defaulted>
interface-notifications
-----
missed-percentage-threshold: 0 percent <defaulted>
notification-interval: 30 seconds <defaulted>
idle-interface-delay: 30 seconds <defaulted>
-----
```

```
cdp-mode: forward-cdp-packets default: drop-cdp-packets
sensor (config-int)#
```

Displaying Interface Statistics



Note

The **show interface** command output for the IPS 4510 and IPS 4520 does not include the total undersize packets or total transmit FIFO overruns.



Note

When the IPS 4510 and IPS 4520 are in bypass mode, VLAN statistics in the **show interface** and **packet display/capture** command output do not show any packets.

The jumbo packet count in the **show interface** command output from the lines `Total Jumbo Packets Received` and `Total Jumbo Packets Transmitted` for ASA IPS modules may be larger than expected due to some packets that were almost jumbo size on the wire being counted as jumbo size by the IPS. This miscount is a result of header bytes added to the packet by the ASA before the packet is transmitted to the IPS. For IPv4, 58 bytes of header data are added. For IPv6, 78 bytes of header data are added. The ASA removes the added IPS header before the packet leaves the ASA.

Use the **show interfaces [clear | brief]** command in EXEC mode to display statistics for all system interfaces. Use the **show interfaces {FastEthernet | GigabitEthernet | Management | PortChannel} [slot/port]** command to display statistics for specific interfaces.

The following commands apply:

- **clear**—(Optional) Clears the diagnostics.
- **brief**—(Optional) Displays a summary of the usability status information for each interface.
- **FastEthernet**—Displays statistics for FastEthernet interfaces.
- **GigabitEthernet**—Displays statistics for GigabitEthernet interfaces.
- **Management**—Displays statistics for Management interfaces.



Note

Only platforms with external ports marked *Management* support this keyword.

- **PortChannel**—Displays statistics for PortChannel interfaces
- *slot/port*—Displays statistics for the specific slot/port of the interface.

Displaying Interface Statistics

To display interface statistics, follow these steps:

- Step 1** Log in to the CLI.
- Step 2** Display statistics for all interfaces.

```
sensor# show interfaces
Interface Statistics
  Total Packets Received = 0
  Total Bytes Received = 0
```

```

Missed Packet Percentage = 0
Current Bypass Mode = Auto_off
MAC statistics from interface GigabitEthernet0/0
Statistics From Subinterface 12
  Vlans in this group = 12
  Total Packets Received On This Vlan Group = 0
  Total Bytes Received On This Vlan Group = 0
  Total Packets Transmitted On This Vlan Group = 0
  Total Bytes Transmitted On This Vlan Group = 0
Statistics From Subinterface 16
  Vlans in this group = 10
  Total Packets Received On This Vlan Group = 0
  Total Bytes Received On This Vlan Group = 0
  Total Packets Transmitted On This Vlan Group = 0
  Total Bytes Transmitted On This Vlan Group = 0
Statistics From Subinterface 25
  Vlans in this group = 11
  Total Packets Received On This Vlan Group = 0
  Total Bytes Received On This Vlan Group = 0
  Total Packets Transmitted On This Vlan Group = 0
  Total Bytes Transmitted On This Vlan Group = 0
--MORE--

```

Step 3 Show a brief summary of the interfaces. The * indicates that the interface is the command and control interface.

```

sensor# show interfaces brief
CC  Interface           Sensing State  Link  Inline Mode  Pair Status
    GigabitEthernet0/0 Disabled       Down  Unpaired     N/A
*   Management0/0      Disabled       Up    Unpaired     N/A
    GigabitEthernet0/1 Disabled       Down  Unpaired     N/A
    GigabitEthernet0/2 Disabled       Down  Unpaired     N/A
    GigabitEthernet0/3 Disabled       Down  Unpaired     N/A
sensor#

```

Step 4 Display the statistics for a specific interface.

```

sensor# show interfaces Management0/0
MAC statistics from interface Management0/0
  Interface function = Command-control interface
  Description =
  Media Type = TX
  Default Vlan = 0
  Link Status = Up
  Link Speed = Auto_100
  Link Duplex = Auto_Full
  Total Packets Received = 4305909
  Total Bytes Received = 280475712
  Total Multicast Packets Received = 0
  Total Receive Errors = 0
  Total Receive FIFO Overruns = 0
  Total Packets Transmitted = 973627
  Total Bytes Transmitted = 437632618
  Total Transmit Errors = 0
  Total Transmit FIFO Overruns = 0
sensor#

```

Step 5 Clear the statistics.

```

sensor# show interfaces clear
Interface Statistics
  Total Packets Received = 0
  Total Bytes Received = 0
  Missed Packet Percentage = 0

```

```

Current Bypass Mode = Auto_off
MAC statistics from interface GigabitEthernet0/0
  Statistics From Subinterface 12
    Vlans in this group = 12
    Total Packets Received On This Vlan Group = 0
    Total Bytes Received On This Vlan Group = 0
    Total Packets Transmitted On This Vlan Group = 0
    Total Bytes Transmitted On This Vlan Group = 0
  Statistics From Subinterface 16
    Vlans in this group = 10
    Total Packets Received On This Vlan Group = 0
    Total Bytes Received On This Vlan Group = 0
    Total Packets Transmitted On This Vlan Group = 0
    Total Bytes Transmitted On This Vlan Group = 0
  Statistics From Subinterface 25
    Vlans in this group = 11
    Total Packets Received On This Vlan Group = 0
    Total Bytes Received On This Vlan Group = 0
    Total Packets Transmitted On This Vlan Group = 0
    Total Bytes Transmitted On This Vlan Group = 0
--MORE--

```

For More Information

For information on slot and port numbers and which platforms have a Management port, refer to [Cisco Intrusion Prevention System Appliances and Modules Installation Guide for IPS 7.3](#).

Displaying Interface Traffic History

Use the **show interfaces-history [traffic-by-hour | traffic-by-minute]** command in EXEC mode to display historical interfaces statistics for all system interfaces. The historical information for each interface is maintained for three days with 60 seconds granularity. Use the **show interfaces-history {FastEthernet | GigabitEthernet | Management | PortChannel} [traffic-by-hour | traffic-by-minute]** command to display statistics for specific interfaces.

**Note**

You must have health monitoring enabled to support the historic interface function.

Each record has the following details:

- Total packets received
- Total bytes received
- FIFO overruns
- Receive errors
- Received Mbps
- Missed packet percentage
- Average load
- Peak load

**Note**

Historical data for each interface for the past 72 hours is also included in the **show tech-support** command.

The following commands apply:

- **traffic-by-hour**—Displays interface traffic history by the hour.
- **traffic-by-minute**—Displays interface traffic history by the minute.
- **past**—Displays historical interface traffic information.
- **HH:MM**—Specifies the amount of time to go back in the past to begin the traffic display. The range for HH is 0 to 72. The range for MM is 0 to 59. The minimum value is 00:01 and the maximum value is 72:00.
- **FastEthernet**—Displays statistics for FastEthernet interfaces.
- **GigabitEthernet**—Displays statistics for GigabitEthernet interfaces.
- **Management**—Displays statistics for Management interfaces.

**Note**

Only platforms with external ports marked *Management* support this keyword.

- **PortChannel**—Displays statistics for PortChannel interfaces.

Displaying Historical Interface Statistics

To display interface traffic history, follow these steps:

Step 1 Log in to the CLI.

Step 2 Display the interface traffic history by the hour.

```
sensor# show interfaces-history traffic-by-hour past 02:15
GigabitEthernet0/0
Time                Packets Received  Bytes Received  Mbps  MPP  FIFO  Overruns  Receive Errors  Avg Load  Peak Load
11:30:31 UTC Tue Mar 05 2013  0                0                0     0    0     0          0                0         0
10:27:32 UTC Tue Mar 05 2013  0                0                0     0    0     0          0                0         0

GigabitEthernet0/1
Time                Packets Received  Bytes Received  Mbps  MPP  FIFO  Overruns  Receive Errors  Avg Load  Peak Load
11:30:31 UTC Tue Mar 05 2013  0                0                0     0    0     0          0                0         0
10:27:32 UTC Tue Mar 05 2013  0                0                0     0    0     0          0                0         0

GigabitEthernet0/2
Time                Packets Received  Bytes Received  Mbps  MPP  FIFO  Overruns  Receive Errors  Avg Load  Peak Load
11:30:31 UTC Tue Mar 05 2013  0                0                0     0    0     0          0                0         0
10:27:32 UTC Tue Mar 05 2013  0                0                0     0    0     0          0                0         0

GigabitEthernet0/3
Time                Packets Received  Bytes Received  Mbps  MPP  FIFO  Overruns  Receive Errors  Avg Load  Peak Load
11:30:31 UTC Tue Mar 05 2013  0                0                0     0    0     0          0                0         0
10:27:32 UTC Tue Mar 05 2013  0                0                0     0    0     0          0                0         0

Management0/0
Time                Packets Received  Bytes Received  Mbps  MPP  FIFO  Overruns  Receive Errors  Avg Load  Peak Load
11:30:31 UTC Tue Mar 05 2013  31071600         3240924703      0     0    0     0          0                0         0
10:27:32 UTC Tue Mar 05 2013  30859941         3216904786      0     0    0     0          0                0         0

--MORE--
```

Step 3 Display the interface traffic history by the minute.

```
sensor# show interfaces-history traffic-by-minute past 00:45
GigabitEthernet0/0
```

■ Displaying Interface Traffic History

```

Time
Load
12:27:49 UTC Tue Mar 05 2013 0 0 0 0 0 0 0 0 0 0
12:26:45 UTC Tue Mar 05 2013 0 0 0 0 0 0 0 0 0 0
12:25:48 UTC Tue Mar 05 2013 0 0 0 0 0 0 0 0 0 0
12:24:42 UTC Tue Mar 05 2013 0 0 0 0 0 0 0 0 0 0
12:23:37 UTC Tue Mar 05 2013 0 0 0 0 0 0 0 0 0 0
12:22:30 UTC Tue Mar 05 2013 0 0 0 0 0 0 0 0 0 0
12:21:31 UTC Tue Mar 05 2013 0 0 0 0 0 0 0 0 0 0
12:20:29 UTC Tue Mar 05 2013 0 0 0 0 0 0 0 0 0 0
12:19:25 UTC Tue Mar 05 2013 0 0 0 0 0 0 0 0 0 0
12:18:18 UTC Tue Mar 05 2013 0 0 0 0 0 0 0 0 0 0
12:17:12 UTC Tue Mar 05 2013 0 0 0 0 0 0 0 0 0 0
12:16:07 UTC Tue Mar 05 2013 0 0 0 0 0 0 0 0 0 0
12:15:00 UTC Tue Mar 05 2013 0 0 0 0 0 0 0 0 0 0
12:13:54 UTC Tue Mar 05 2013 0 0 0 0 0 0 0 0 0 0
12:12:49 UTC Tue Mar 05 2013 0 0 0 0 0 0 0 0 0 0
12:11:43 UTC Tue Mar 05 2013 0 0 0 0 0 0 0 0 0 0
12:10:36 UTC Tue Mar 05 2013 0 0 0 0 0 0 0 0 0 0
12:09:30 UTC Tue Mar 05 2013 0 0 0 0 0 0 0 0 0 0
12:08:24 UTC Tue Mar 05 2013 0 0 0 0 0 0 0 0 0 0
12:07:25 UTC Tue Mar 05 2013 0 0 0 0 0 0 0 0 0 0
12:06:23 UTC Tue Mar 05 2013 0 0 0 0 0 0 0 0 0 0
12:05:25 UTC Tue Mar 05 2013 0 0 0 0 0 0 0 0 0 0
sensor#

```

Step 4 Display the interface traffic history for a specific interface.

```

sensor# show interfaces-history GigabitEthernet0/0 traffic-by-minute past 00:05
GigabitEthernet0/0
Time
13:34:38 UTC Thu Mar 07 2013 0 0 0 0 0 0 0 0 0 0
13:33:35 UTC Thu Mar 07 2013 0 0 0 0 0 0 0 0 0 0
13:32:32 UTC Thu Mar 07 2013 0 0 0 0 0 0 0 0 0 0
13:31:27 UTC Thu Mar 07 2013 0 0 0 0 0 0 0 0 0 0
13:30:25 UTC Thu Mar 07 2013 0 0 0 0 0 0 0 0 0 0
sensor#

```

For More Information

For information on enabling health monitoring, see [Configuring Health Status Information, page 17-14](#).



Configuring Virtual Sensors

This chapter explains the function of the Analysis Engine and how to create, edit, and delete virtual sensors. It also explains how to assign interfaces to a virtual sensor. It contains the following sections:

- [Virtual Sensor Notes and Caveats, page 6-1](#)
- [Understanding the Analysis Engine, page 6-2](#)
- [Understanding Virtual Sensors, page 6-2](#)
- [Advantages and Restrictions of Virtualization, page 6-2](#)
- [Inline TCP Session Tracking Mode, page 6-3](#)
- [Normalization and Inline TCP Evasion Protection Mode, page 6-4](#)
- [HTTP Advanced Decoding, page 6-4](#)
- [Adding, Editing, and Deleting Virtual Sensors, page 6-5](#)
- [Configuring Global Variables, page 6-12](#)

Virtual Sensor Notes and Caveats

The following notes and caveats apply to configuring the virtual sensor:

- The Cisco IPS does not support more than four virtual sensors. You cannot delete the default virtual sensor vs0.
- The ASA IPS modules (ASA 5500-X IPS SSP and ASA 5585-X IPS SSP) do not support the inline TCP session tracking mode.
- For the ASA IPS modules (ASA 5500-X IPS SSP and ASA 5585-X IPS SSP), normalization is performed by the adaptive security appliance and not the IPS.
- Anomaly detection is disabled by default. You must enable it to configure or apply an anomaly detection policy. Enabling anomaly detection results in a decrease in performance.

Understanding the Analysis Engine

The Analysis Engine performs packet analysis and alert detection. It monitors traffic that flows through specified interfaces.

You create virtual sensors in the Analysis Engine. Each virtual sensor has a unique name with a list of interfaces, inline interface pairs, inline VLAN pairs, and VLAN groups associated with it. To avoid definition ordering issues, no conflicts or overlaps are allowed in assignments. You assign interfaces, inline interface pairs, inline VLAN pairs, and VLAN groups to a specific virtual sensor so that no packet is processed by more than one virtual sensor. Each virtual sensor is also associated with a specifically named signature definition, event action rules, and anomaly detection configuration. Packets from interfaces, inline interface pairs, inline VLAN pairs, and VLAN groups that are not assigned to any virtual sensor are disposed of according to the inline bypass configuration.

**Note**

The Cisco IPS does not support more than four virtual sensors. You cannot delete the default virtual sensor vs0.

Understanding Virtual Sensors

The sensor can receive data inputs from one or many monitored data streams. These monitored data streams can either be physical interface ports or virtual interface ports. For example, a single sensor can monitor traffic from in front of the firewall, from behind the firewall, or from in front of and behind the firewall concurrently. And a single sensor can monitor one or more data streams. In this situation a single sensor policy or configuration is applied to all monitored data streams.

A virtual sensor is a collection of data that is defined by a set of configuration policies. The virtual sensor is applied to a set of packets as defined by interface component.

A virtual sensor can monitor multiple segments, and you can apply a different policy or configuration for each virtual sensor within a single physical sensor. You can set up a different policy per monitored segment under analysis. You can also apply the same policy instance, for example, sig0, rules0, or ad0, to different virtual sensors. You can assign interfaces, inline interface pairs, inline VLAN pairs, and VLAN groups to a virtual sensor.

**Note**

The default virtual sensor is vs0. You cannot delete the default virtual sensor. The interface list, the anomaly detection operational mode, the inline TCP session tracking mode, and the virtual sensor description are the only configuration features you can change for the default virtual sensor. You cannot change the signature definition, event action rules, or anomaly detection policies.

Advantages and Restrictions of Virtualization

Virtualization has the following advantages:

- You can apply different configurations to different sets of traffic.
- You can monitor two networks with overlapping IP spaces with one sensor.
- You can monitor both inside and outside of a firewall or NAT device.

Virtualization has the following restrictions:

- You must assign both sides of asymmetric traffic to the same virtual sensor.
- Using VACL capture or SPAN (promiscuous monitoring) is inconsistent with regard to VLAN tagging, which causes problems with VLAN groups.
 - When using Cisco IOS software, a VACL capture port or a SPAN target does not always receive tagged packets even if it is configured for trunking.
 - When using the MSFC, fast path switching of learned routes changes the behavior of VACL captures and SPAN.
- Persistent store is limited.

Virtualization has the following traffic capture requirements:

- The virtual sensor must receive traffic that has 802.1q headers (other than traffic on the native VLAN of the capture port).
- The sensor must see both directions of traffic in the same VLAN group in the same virtual sensor for any given sensor.

The following sensors support virtualization:

- ASA 5500-X IPS SSP
- ASA 5585-X IPS SSP
- IPS 4345
- IPS 4345-DC
- IPS 4360
- IPS 4510
- IPS 4520
- IPS 4520-XL

Inline TCP Session Tracking Mode



Note

The ASA IPS modules (ASA 5500-X IPS SSP and ASA 5585-X IPS SSP) do not support the inline TCP session tracking mode.

When you choose to modify packets inline, if the packets from a stream are seen twice by the Normalizer engine, it cannot properly track the stream state and often the stream is dropped. This situation occurs most often when a stream is routed through multiple VLANs or interfaces that are being monitored by the IPS. A further complication in this situation is the necessity of allowing asymmetric traffic to merge for proper tracking of streams when the traffic for either direction is received from different VLANs or interfaces. To deal with this situation, you can set the mode so that streams are perceived as unique if they are received on separate interfaces and/or VLANs (or the subinterface for VLAN pairs).

The following inline TCP session tracking modes apply:

- Interface and VLAN—All packets with the same session key (AaBb) in the same VLAN (or inline VLAN pair) and on the same interface belong to the same session. Packets with the same key but on different VLANs are tracked separately.

- VLAN Only—All packets with the same session key (AaBb) in the same VLAN (or inline VLAN pair) regardless of the interface belong to the same session. Packets with the same key but on different VLANs are tracked separately.
- Virtual Sensor—All packets with the same session key (AaBb) within a virtual sensor belong to the same session. This is the default and almost always the best option to choose.

Normalization and Inline TCP Evasion Protection Mode



Note

For the ASA IPS modules (ASA 5500-X IPS SSP and ASA 5585-X IPS SSP), normalization is performed by the adaptive security appliance and not the IPS.

Normalization only applies when the sensor is operating in inline mode. The default is strict evasion protection, which is full enforcement of TCP state and sequence tracking. The Normalizer enforces duplicate packets, changed packets, out-of-order packets, and so forth, which helps prevent attackers from evading the IPS.

Asymmetric mode disables most of the Normalizer checks. Use asymmetric mode only when the entire stream cannot be inspected, because in this situation, attackers can now evade the IPS.

HTTP Advanced Decoding

HTTP advanced decoding facilitates analysis of encoded HTTP return web traffic by using on-the-fly decoding. Changes to HTTP advanced decoding take effect immediately and only affect the new traffic flows.

Restrictions

The following restrictions apply when you enable HTTP advanced decoding:

- Although HTTP advanced decoding does not fire any new signatures, drop packets, or modify traffic, it allows existing signatures to match on content that was previously not detectable because of encodings.
- HTTP advanced decoding only acts on return web response traffic.



Caution

Enabling HTTP advanced decoding severely impacts system performance.



Note

Because HTTP advanced decoding requires the Regex card and the String XL engine, it is available only to those platforms that have them. HTTP advanced decoding is supported on the ASA 5585-X IPS SSP, ASA 5525-X IPS SSP, ASA 5545-X IPS SSP, ASA 5555-X IPS SSP, IPS 4345, IPS 4360, IPS 4510, IPS 4520, and IPS 4520-XL.

Adding, Editing, and Deleting Virtual Sensors

This section describes how to add, edit, and delete virtual sensors, and contains the following topics:

- [Adding Virtual Sensors, page 6-5](#)
- [Editing and Deleting Virtual Sensors, page 6-9](#)

Adding Virtual Sensors

Use the **virtual-sensor** *name* command in service analysis engine submode to create a virtual sensor. You can create up to four virtual sensors. You assign policies (anomaly detection, event action rules, and signature definition) to the virtual sensor. Then you assign interfaces (promiscuous, inline interface pairs, inline VLAN pairs, and VLAN groups) to the virtual sensor. You must configure the inline interface pairs and VLAN pairs before you can assign them to a virtual sensor.



Note

Anomaly detection is disabled by default. You must enable it to configure or apply an anomaly detection policy. Enabling anomaly detection results in a decrease in performance.

The following commands apply:

- **http-advanced-decoding {true | false}**—Enables deeper inspection of HTTP traffic. The default is disabled. Valid for IPS 7.1(5)E4 and later.



Caution

Enabling HTTP advanced decoding severely impacts system performance.



Note

HTTP advanced decoding is supported on the ASA 5585-X IPS SSP, ASA 5525-X IPS SSP, ASA 5545-X IPS SSP, ASA 5555-X IPS SSP, IPS 4345, IPS 4360, IPS 4510, IPS 4520, and IPS 4520-XL.

- **anomaly-detection**—Specifies the anomaly detection parameters:
 - **anomaly-detection-name** *name*—Specifies the name of the anomaly detection policy.
 - **operational-mode**—Specifies the anomaly detection mode (**inactive**, **learn**, **detect**).
- **description**—Description of the virtual sensor.
- **event-action-rules**—Specifies the name of the event action rules policy.
- **inline-TCP-evasion-protection-mode**—Lets you choose which type of normalization you need for traffic inspection:
 - **asymmetric** —Specifies that the sensor can only see one direction of bidirectional traffic flow. Asymmetric mode protection relaxes the evasion protection at the TCP layer.



Note

Asymmetric mode lets the sensor synchronize state with the flow and maintain inspection for those engines that do not require both directions. Asymmetric mode lowers security because full protection requires both sides of traffic to be seen.

- **strict**—Specifies that if a packet is missed for any reason, all packets after the missed packet are not processed. Strict evasion protection provides full enforcement of TCP state and sequence tracking.



Note Any out-of-order packets or missed packets can produce Normalizer engine signatures 1300 or 1330 firings, which try to correct the situation, but can result in denied connections.



Note For the ASA IPS modules (ASA 5500-X IPS SSP and ASA 5585-X IPS SSP), normalization is performed by the adaptive security appliance and not the IPS.

- **inline-TCP-session-tracking-mode**—Enables an advanced method used to identify duplicate TCP sessions in inline traffic. The default is virtual sensor, which is almost always the best choice.
 - **virtual-sensor** —Specifies that all packets with the same session key (AaBb) within a virtual sensor belong to the same session.
 - **interface-and-vlan**—Specifies that all packets with the same session key (AaBb) in the same VLAN (or inline VLAN pair) and on the same interface belong to the same session. Packets with the same key but on different VLANs or interfaces are tracked independently.
 - **vlan-only**—Specifies that all packets with the same session key (AaBb) in the same VLAN (or inline VLAN pair) regardless of the interface belong to the same session. Packets with the same key but on different VLANs are tracked independently.



Note The ASA IPS modules (ASA 5500-X IPS SSP and ASA 5585-X IPS SSP) do not support the inline TCP session tracking mode.

- **signature-definition**—Specifies the name of the signature definition policy.
- **logical-interfaces**—Specifies the name of the logical interfaces (inline interface pairs).
- **physical-interfaces**—Specifies the name of the physical interfaces (promiscuous, inline VLAN pairs, and VLAN groups):
 - **subinterface-number**—Specifies the physical subinterface number. If the subinterface-type is none, the value of 0 indicates the entire interface is assigned in promiscuous mode.
- **no**—Removes an entry or selection.

Adding a Virtual Sensor

To add a virtual sensor, follow these steps:

Step 1 Log in to the CLI using an account with administrator privileges.

Step 2 Enter service analysis mode.

```
sensor# configure terminal
sensor(config)# service analysis-engine
sensor(config-ana)#
```

Step 3 Add a virtual sensor.

```
sensor(config-ana)# virtual-sensor vs1
sensor(config-ana-vir)#
```

Step 4 Add a description for this virtual sensor.

```
sensor(config-ana-vir)# description virtual sensor 1
```

Step 5 Assign an anomaly detection policy and operational mode to this virtual sensor.

```
sensor(config-ana-vir)# anomaly-detection
sensor(config-ana-vir-ano)# anomaly-detection-name ad1
sensor(config-ana-vir-ano)# operational-mode learn
```

Step 6 Assign an event action rules policy to this virtual sensor.

```
sensor(config-ana-vir-ano)# exit
sensor(config-ana-vir)# event-action-rules rules1
```

Step 7 Assign a signature definition policy to this virtual sensor.

```
sensor(config-ana-vir)# signature-definition sig1
```

Step 8 Enable HTTP advanced decoding.

```
sensor(config-ana-vir)# http-advanced-decoding true
```



Caution

Enabling HTTP advanced decoding severely impacts system performance.

Step 9 Assign the inline TCP session tracking mode. The default is virtual sensor mode, which is almost always the best option to choose.

```
sensor(config-ana-vir)# inline-TCP-session-tracking-mode virtual-sensor
```

Step 10 Assign the inline TCP evasion protection mode. The default is strict mode, which is almost always the best option to choose.

```
sensor(config-ana-vir)# inline-TCP-evasion-protection-mode strict
```

Step 11 Enable HTTP advanced decoding.

```
sensor(config-ana-vir)# http-advanced-decoding true
```

Step 12 Display the list of available interfaces.

```
sensor(config-ana-vir)# physical-interface ?
GigabitEthernet0/0    GigabitEthernet0/0 physical interface.
GigabitEthernet0/1    GigabitEthernet0/1 physical interface.
GigabitEthernet2/0    GigabitEthernet0/2 physical interface.
GigabitEthernet2/1    GigabitEthernet0/3 physical interface.
sensor(config-ana-vir)# physical-interface

sensor(config-ana-vir)# logical-interface ?
<none available>
```

Step 13 Assign the promiscuous mode interfaces you want to add to this virtual sensor. Repeat this step for all the promiscuous interfaces that you want to assign to this virtual sensor.

```
sensor(config-ana-vir)# physical-interface GigabitEthernet0/3
```

Step 14 Assign the inline interface pairs you want to add to this virtual sensor. You must have already paired the interfaces.

```
sensor(config-ana-vir)# logical-interface inline_interface_pair_name
```

- Step 15** Assign the subinterfaces of the inline VLAN pairs or groups you want to add to this virtual sensor. You must have already subdivided any interfaces into VLAN pairs or groups.

```
sensor(config-ana-vir)# physical-interface GigabitEthernet2/0 subinterface-number
subinterface_number
```

- Step 16** Verify the virtual sensor settings.

```
sensor(config-ana-vir)# show settings
name: vs1
-----
description: virtual sensor 1 default:
signature-definition: sig1 default: sig0
event-action-rules: rules1 default: rules0
anomaly-detection
-----
anomaly-detection-name: ad1 default: ad0
operational-mode: learn default: detect
-----
physical-interface (min: 0, max: 999999999, current: 2)
-----
name: GigabitEthernet0/3
subinterface-number: 0 <defaulted>
-----
inline-TCP-session-tracking-mode: virtual-sensor default: virtual-sensor
-----
logical-interface (min: 0, max: 999999999, current: 0)
-----
-----
sensor(config-ana-vir)#
```

- Step 17** Exit analysis engine mode.

```
sensor(config-ana-vir)# exit
sensor(config-ana)# exit
sensor(config)#
Apply Changes:[yes]:
```

- Step 18** Press **Enter** to apply the changes or enter **no** to discard them.

For More Information

- For the procedure for creating virtual sensors on the ASA 5500-X IPS SSP, see [Creating Virtual Sensors for the ASA 5500-X IPS SSP, page 18-3](#).
- For the procedure for creating virtual sensors on the ASA 5585-X IPS SSP, see [Creating Virtual Sensors for the ASA 5585-X IPS SSP, page 19-4](#).
- For more information on creating and configuring anomaly detection policies, see [Working With Anomaly Detection Policies, page 9-8](#).
- For more information on creating and configuring event action rules policies, see [Working With Event Action Rules Policies, page 8-8](#).
- For more information on creating and configuring signature definition policies, see [Working With Signature Definition Policies, page 7-2](#).
- For more information about normalization, see [Normalization and Inline TCP Evasion Protection Mode, page 6-4](#).
- For more information about inline TCP session tracking mode, see [Inline TCP Session Tracking Mode, page 6-3](#).

- For the procedure for pairing inline interfaces, see [Configuring Inline Interface Pairs, page 5-17](#). Repeat Step 11 for all the inline interface pairs that you want to assign to this virtual sensor.
- For the procedure for pairing and grouping inline VLANs, see [Configuring Inline VLAN Pairs, page 5-22](#) and [Configuring VLAN Groups, page 5-28](#). Repeat Step 12 for all inline VLAN pairs or VLAN groups that you want to assign to this virtual sensor.
- For the procedure for enabling anomaly detection, see [Enabling Anomaly Detection, page 9-8](#).

Editing and Deleting Virtual Sensors

You can edit the following parameters of a virtual sensor:

- Signature definition policy
- Event action rules policy
- Anomaly detection policy



Note

Anomaly detection is disabled by default. You must enable it to configure or apply an anomaly detection policy. Enabling anomaly detection results in a decrease in performance.

- Anomaly detection operational mode
- Inline TCP session tracking mode



Note

The ASA IPS modules (ASA 5500-X IPS SSP and ASA 5585-X IPS SSP) do not support the inline TCP session tracking mode.

- Description
- Interfaces assigned

Editing or Deleting a Virtual Sensor

To edit or delete a virtual sensor, follow these steps:

Step 1 Log in to the CLI using an account with administrator privileges.

Step 2 Enter analysis engine mode.

```
sensor# configure terminal
sensor(config)# service analysis-engine
sensor(config-ana)#
```

Step 3 Edit the virtual sensor, vs1.

```
sensor(config-ana)# virtual-sensor vs1
sensor(config-ana-vir)#
```

Step 4 Edit the description of this virtual sensor.

```
sensor(config-ana-vir)# description virtual sensor A
```

Step 5 Change the anomaly detection policy and operational mode assigned to this virtual sensor.

```
sensor(config-ana-vir)# anomaly-detection
sensor(config-ana-vir-ano)# anomaly-detection-name ad0
sensor(config-ana-vir-ano)# operational-mode learn
```

Step 6 Change the event action rules policy assigned to this virtual sensor.

```
sensor(config-ana-vir-ano)# exit
sensor(config-ana-vir)# event-action-rules rules0
```

Step 7 Change the signature definition policy assigned to this virtual sensor.

```
sensor(config-ana-vir)# signature-definition sig0
```

Step 8 Change the inline TCP session tracking mode. The default is virtual sensor mode, which is almost always the best option to choose.

```
sensor(config-ana-vir)# inline-TCP-session-tracking-mode interface-and-vlan
```

Step 9 Display the list of available interfaces.

```
sensor(config-ana-vir)# physical-interface ?
GigabitEthernet0/0    GigabitEthernet0/0 physical interface.
GigabitEthernet0/1    GigabitEthernet0/1 physical interface.
GigabitEthernet2/0    GigabitEthernet0/2 physical interface.
GigabitEthernet2/1    GigabitEthernet0/3 physical interface.
sensor(config-ana-vir)# physical-interface

sensor(config-ana-vir)# logical-interface ?
<none available>
```

Step 10 Change the promiscuous mode interfaces assigned to this virtual sensor.

```
sensor(config-ana-vir)# physical-interface GigabitEthernet0/2
```

Step 11 Change the inline interface pairs assigned to this virtual sensor. You must have already paired the interfaces.

```
sensor(config-ana-vir)# logical-interface inline_interface_pair_name
```

Step 12 Change the subinterface with the inline VLAN pairs or groups assigned to this virtual sensor. You must have already subdivided any interfaces into VLAN pairs or groups.

```
sensor(config-ana-vir)# physical-interface GigabitEthernet2/0 subinterface-number
subinterface_number
```

Step 13 Verify the edited virtual sensor settings.

```
ssensor(config-ana-vir)# show settings
name: vs1
-----
description: virtual sensor 1 default:
signature-definition: sig1 default: sig0
event-action-rules: rules1 default: rules0
anomaly-detection
-----
anomaly-detection-name: ad1 default: ad0
operational-mode: learn default: detect
-----
physical-interface (min: 0, max: 999999999, current: 2)
-----
name: GigabitEthernet0/3
subinterface-number: 0 <defaulted>
-----
inline-TCP-session-tracking-mode: interface-and-vlan default: virtual-sensor
```

```
-----
logical-interface (min: 0, max: 999999999, current: 0)
-----
-----

```

```
sensor(config-ana-vir)#
```

Step 14 Delete a virtual sensor.

```
sensor(config-ana-vir)# exit
sensor(config-ana)# no virtual-sensor vs1
```

Step 15 Verify the deleted virtual sensor. Only the default virtual sensor, vs0, is present.

```
sensor(config-ana)# show settings
global-parameters
-----
ip-logging
-----
max-open-iplog-files: 20 <defaulted>
-----
virtual-sensor (min: 1, max: 255, current: 2)
-----
<protected entry>
name: vs0 <defaulted>
-----
description: default virtual sensor <defaulted>
signature-definition: sig0 <protected>
event-action-rules: rules0 <protected>
anomaly-detection
-----
anomaly-detection-name: ad0 <protected>
operational-mode: detect <defaulted>
-----
physical-interface (min: 0, max: 999999999, current: 0)
-----
logical-interface (min: 0, max: 999999999, current: 0)
-----
sensor(config-ana)#
```

Step 16 Exit analysis engine mode.

```
sensor(config-ana)# exit
sensor(config)#
Apply Changes:[yes]:
```

Step 17 Press **Enter** to apply the changes or enter **no** to discard them.

For More Information

- For more information on creating and configuring anomaly detection policies, see [Working With Anomaly Detection Policies, page 9-8](#).
- For more information on creating and configuring event action rules policies, see [Working With Event Action Rules Policies, page 8-8](#).
- For more information on creating and configuring signature definition policies, see [Working With Signature Definition Policies, page 7-2](#).

- For the procedure for pairing inline interfaces, see [Configuring Inline Interface Pairs, page 5-17](#). Repeat Step 11 for all the inline interface pairs that you want to assign to this virtual sensor.
- For the procedure for pairing and grouping inline VLANs, see [Configuring Inline VLAN Pairs, page 5-22](#) and [Configuring VLAN Groups, page 5-28](#). Repeat Step 12 for all inline VLAN pairs or VLAN groups that you want to assign to this virtual sensor.
- For the procedure for enabling anomaly detection, see [Enabling Anomaly Detection, page 9-8](#).

Configuring Global Variables

Use the **global-parameters** command in service analysis engine submode to create global variables, such as IP logging, service activity, and specifying the flow depth. Flow depth is used for String, Multi-String, Service HTTP, and State engines. It does not apply to the XL String engine and the platforms that support it.



Note

The IPS 4345, IPS 4360, IPS 4510, IPS 4520, IPS 4520-XL, ASA 5525-X IPS SSP, ASA 5545-X IPS SSP, ASA 5555-X IPS SSP, and ASA 5585-X IPS SSP support the String XL engines and the Regex accelerator card.

The following commands apply:

- **ip-logging**—Enables global IP logging parameters.
 - **max-open-iplog-files**—Specifies the maximum number of concurrently open log files. The range is 20 to 100. The default is 20.
- **serviceActivity**—Lets you gather information about service activities for diagnostic purposes. The details are more granular and have port level details.



Note

Enabling service activity impacts system performance. Enable service activity collection temporarily for diagnostic purposes only. You must reboot the sensor after you enable service activity for the change to take affect.

- **enable-serviceactivity [1 | 0]**—Set to 1 to enable, set to 0 to disable. The default is disabled.
- **serviceActivityLimit limit**—Sets the limit for how many services you want to enable. The valid range is from 10 to 65536. The default is 15.
- **specify-flow-depth**—Lets you specify the inspection depth of the flow. Flow depth is the number of bytes inspected in a flow. The new value applies for new flows only. The valid range is from 0 to 429496296. The default is 0, which is infinite. Creating a Global Variable

To create a global variable, follow these steps:

Step 1 Log in to the CLI using an account with administrator privileges.

Step 2 Enter service analysis mode.

```
sensor# configure terminal
sensor(config)# service analysis-engine
sensor(config-ana)#
```

Step 3 Create the variable for the maximum number of open IP logs.

```
sensor(config-ana)# global-parameters
```

```

sensor(config-ana-glo)# ip-logging
sensor(config-ana-glo-ip)# max-open-iplog-files 50
sensor(config-ana-glo-ip)# exit
sensor(config-ana-glo)#

```

Step 4 Create the variable for service activity.

```

sensor(config-ana-glo)# serviceActivity
sensor(config-ana-glo-ser)# enable-serviceactivity 1
sensor(config-ana-glo-ser)# serviceActivityLimit 25
sensor(config-ana-glo-ser)# exit
sensor(config-ana-glo)#

```

Step 5 Create the flow depth variable.

```

sensor(config-ana-glo)# specify-flow-depth 500
sensor(config-ana-glo)# exit
sensor(config-ana)#

```

Step 6 Verify the global variable settings.

```

sensor(config-ana-glo)# show settings
global-parameters
-----
specify-flow-depth: 500 default: 0
serviceActivity
-----
enable-serviceactivity: 1 default: 0
serviceActivityLimit: 25 default: 15
-----
ip-logging
-----
max-open-iplog-files: 50 default: 20
-----
sensor(config-ana-glo)#

```

Step 7 Exit analysis engine mode.

```

sensor(config-ana)# exit
sensor(config)#
Apply Changes:[yes]:

```

Step 8 Press **Enter** to apply the changes or enter **no** to discard them.

For More Information

For detailed information about String, Multi-String, Service HTTP, and State engines, see [Appendix B, “Signature Engines.”](#)



Defining Signatures

This chapter describes how to define and create signatures. It contains the following sections:

- [Signature Definition Notes and Caveats, page 7-1](#)
- [Understanding Policies, page 7-2](#)
- [Working With Signature Definition Policies, page 7-2](#)
- [Understanding Signatures, page 7-3](#)
- [Configuring Threat Profiles, page 7-4](#)
- [Configuring Signature Variables, page 7-9](#)
- [Configuring Signatures, page 7-11](#)
- [Creating Custom Signatures, page 7-46](#)

Signature Definition Notes and Caveats

The following notes and caveats apply to defining signatures:

- You must preface signature variables with a dollar (\$) sign to indicate that you are using a variable rather than a string.
- We recommend that you do NOT change the promiscuous delta setting for a signature.
- The parameters **tcp-3-way-handshake-required** and **tcp-reassembly-mode** only impact sensors inspecting traffic in promiscuous mode, not inline mode. To configure asymmetric options for sensors inspecting inline traffic, use the **inline-TCP-evasion-protection-mode** parameter.
- A custom signature can affect the performance of your sensor. Test the custom signature against a baseline sensor performance for your network to determine the overall impact of the signature.
- If a signature has both threat profile tuning and custom tuning, you cannot return the signature to its default configuration using the **default signatures sig-id sub sig-id** command unless the threat profile is removed from the signature instance. Note that the **default signatures** command still works and removes all custom tunings but preserves the threat profile tunings.

Understanding Policies

You can create multiple security policies and apply them to individual virtual sensors. A security policy is made up of a signature definition policy, an event action rules policy, and an anomaly detection policy. Cisco IPS contains a default signature definition policy called sig0, a default event action rules policy called rules0, and a default anomaly detection policy called ad0. You can assign the default policies to a virtual sensor or you can create new policies. The use of multiple security policies lets you create security policies based on different requirements and then apply these customized policies per VLAN or physical interface.

Working With Signature Definition Policies

Use the **service signature-definition** *name* command in service signature definition mode to create a signature definition policy. The values of this signature definition policy are the same as the default signature definition policy, sig0, until you edit them.

Or you can use the **copy signature-definition** *source_destination* command in privileged EXEC mode to make a copy of an existing policy and then edit the values of the new policy as needed.

Use the **list signature-definition-configurations** command in privileged EXEC mode to list the signature definition policies.

Use the **no service signature-definition** *name* command in global configuration mode to delete a signature definition policy. Use the **default service signature-definition** *name* command in global configuration mode to reset the signature definition policy to factory settings.



Note

If a signature has both threat profile tuning and custom tuning, you cannot return the signature to its default configuration using the **default** command unless the threat profile is removed from the signature instance.

Creating, Copying, Editing, and Deleting Signature Definition Policies

To create, copy, edit, and delete signature definition policies, follow these steps:

Step 1 Log in to the CLI using an account with administrator privileges.

Step 2 Create a signature definition policy.

```
sensor# configure terminal
sensor(config)# service signature-definition MySig
Editing new instance MySig.
sensor(config-sig)# exit
Apply Changes?[yes]: yes
sensor(config)# exit
```

Step 3 Or copy an existing signature definition policy to a new signature definition policy.

```
sensor# copy signature-definition sig0 sig1
sensor#
```



Note

You receive an error if the policy already exists or if there is not enough space available for the new policy.

Step 4 Accept the default signature definition policy values or edit the following parameters:

- a. Add signature definition variables.
- b. Configure the general signature options.

Step 5 Display a list of signature definition policies on the sensor.

```
sensor# list signature-definition-configurations
Signature Definition
  Instance   Size   Virtual Sensor   Threat Profile
  sig0       255   vs0               Edge
  temp       707   N/A               NONE
  MySig      255   N/A               Web_Applications
  sig1       141   vs1               Edge
sensor#
```

Step 6 Delete a signature definition policy.

```
sensor# configure terminal
sensor(config)# no service signature-definition MySig
sensor(config)# exit
sensor#
```



Note You cannot delete the default signature definition policy, sig0.

Step 7 Confirm the signature definition policy has been deleted.

```
sensor# list signature-definition-configurations
Signature Definition
  Instance   Size   Virtual Sensor   Threat Profile
  sig0       255   vs0               Edge
  temp       707   N/A               NONE
  sig1       141   vs1               Edge
sensor#
```

Step 8 Reset a signature definition policy to factory settings.

```
sensor# configure terminal
sensor(config)# default service signature-definition sig1
sensor(config)#
```

For More Information

- For the procedure for adding signature variables, see [Configuring Signature Variables, page 7-9](#).
- For the procedure for configuring the general settings, see [Configuring Signatures, page 7-11](#).

Understanding Signatures

Attacks or other misuses of network resources can be defined as network intrusions. Sensors that use a signature-based technology can detect network intrusions. A signature is a set of rules that your sensor uses to detect typical intrusive activity, such as DoS attacks. As sensors scan network packets, they use signatures to detect known attacks and respond with actions that you define.

The sensor compares the list of signatures with network activity. When a match is found, the sensor takes an action, such as logging the event or sending an alert. Sensors let you modify existing signatures and define new ones.

Signature-based intrusion detection can produce false positives because certain normal network activity can be misinterpreted as malicious activity. For example, some network applications or operating systems may send out numerous ICMP messages, which a signature-based detection system might interpret as an attempt by an attacker to map out a network segment. You can minimize false positives by tuning your signatures.

To configure a sensor to monitor network traffic for a particular signature, you must enable the signature. By default, the most critical signatures are enabled when you install the signature update. When an attack is detected that matches an enabled signature, the sensor generates an alert, which is stored in the Event Store of the sensor. The alerts, as well as other events, may be retrieved from the Event Store by web-based clients. By default the sensor logs all Informational alerts or higher.

Some signatures have subsignatures, that is, the signature is divided into subcategories. When you configure a subsignature, changes made to the parameters of one subsignature apply only to that subsignature. For example, if you edit signature 3050 subsignature 1 and change the severity, the severity change applies to only subsignature 1 and not to 3050 2, 3050 3, and 3050 4.

The Cisco IPS contains over 10,000 built-in default signatures. You cannot rename or delete signatures from the list of built-in signatures, but you can retire signatures to remove them from the sensing engine. You can later activate retired signatures; however, this process requires the sensing engines to rebuild their configuration, which takes time and could delay the processing of traffic. You can tune built-in signatures by adjusting several signature parameters. Built-in signatures that have been modified are called tuned signatures.

**Note**

We recommend that you retire any signatures that you are not using. This improves sensor performance.

You can create signatures, which are called custom signatures. Custom signature IDs begin at 60000. You can configure them for several things, such as matching of strings on UDP connections, tracking of network floods, and scans. Each signature is created using a signature engine specifically designed for the type of traffic being monitored.

Configuring Threat Profiles

This section describes threat profiles, how to display, apply, and remove them, and contains the following topics:

- [Understanding Threat Profiles, page 7-4](#)
- [Displaying, Applying, and Removing Threat Profiles, page 7-5](#)
- [Displaying a Signature Tuned with a Threat Profile, page 7-8](#)
- [Displaying Threat Profile Versions, page 7-8](#)

Understanding Threat Profiles

**Note**

Signature threat profiles are supported on the IPS 4345, IPS 4360, IPS 4510, IPS 4520, IPS 4520-XL, ASA 5525-X IPS SSP, ASA 5545-X IPS SSP, ASA 5555-X IPS SSP, and ASA 5585-X IPS SSP.

A signature threat profile is a predefined signature template that includes customized tunings. These tunings adjust the signature coverage and response actions to enable the sensor to make better choices in various deployment and threat scenarios. You can apply a signature threat profile to one or more signature policies.

You can dynamically upgrade threat profiles through signature upgrades. You can see a description of the template when you select it. Threat profiles may tune several signatures and when these signature policies are assigned to virtual sensors, depending on the signatures turned ON in the threat profile, there may be increased usage of the resources of the sensor. Furthermore, based on the traffic pattern of your network, these signatures may be further tuned.

Once you apply a signature template to a signature policy, you can make modifications to the signature policy, such as retiring a signature to eliminate a false positive. Your changes are NOT overwritten during signature updates or sensor software upgrades.

The following threat profiles are part of the signature upgrades:

- Supervisory Control and Data Acquisition (SCADA)—In addition to signatures in the default set, the SCADA signature template includes specialized signatures for general SCADA protocol detections and specific identifiers that address tools and environments common to most device controlled environments. Use this template if the Cisco IPS is primarily used for protecting Industrial Control Systems (ICS).

**Caution**

You must purchase a SCADA signature license to use the SCADA threat profile. For more information refer to the following URL:

http://www.cisco.com/en/US/prod/collateral/vpndevc/ps5729/ps5713/ps4077/ips_industrial_control_protection.pdf

- Edge—In addition to signatures in the default set, the Internet Edge signature template includes additional signatures that provide broader protection for desktop operating systems, web browsers, web technologies, and common desktop applications. Use this template if the Cisco IPS is primarily used for securing an Internet connection.
- Web Applications—In addition to signatures in the default set, the Web Applications signature template includes additional signatures that provide broader protection for web servers, web development tools and frameworks, content management systems, load balancers, and databases. Use this template if the Cisco IPS is primarily used for protecting web server farms.
- Data Center—In addition to signatures in the default set, the Data Center signature template includes additional signatures that provide broader protection for server operating systems, web servers, application servers, databases, content management systems, messaging servers, and virtualization systems. Use this template if the Cisco IPS is primarily used for protecting data centers.

Displaying, Applying, and Removing Threat Profiles

Use the **service threat-profile** *signature_instance* command in service threat profile mode to configure threat profiles. The threat profiles adjust the signature coverage and response actions of the virtual sensor to better match deployment and threat scenarios.

**Note**

Only one threat profile per signature instance is allowed. Make sure that you have already created a signature instance using the **service signature-definition** command before applying the threat profile.

The following commands apply:

- **list threat-profiles**—Displays the list of threat profiles available on the sensor:
 - SCADA—Default signature set plus SCADA-specific signatures, that is, signatures for protecting Industrial Control Systems (ICS).



Caution

You must purchase a SCADA signature license to use the SCADA threat profile. For more information refer to the following URL:

http://www.cisco.com/en/US/prod/collateral/vpndevc/ps5729/ps5713/ps4077/ips_industrial_control_protection.pdf

- Edge—Default signature set plus signatures that help in securing an Internet connection.
- Web_Applications—Default signature set plus signatures for protecting web server farms.
- Data_Center—Default signature set plus signatures for protecting data centers.
-
- **list signature-definition-configurations**—Displays the threat profiles applied on the virtual sensor. ‘None’ is displayed if there is no threat profile applied or if the threat profile is removed from the signature instance by the **no threat-profile** command.
- **service threat-profile signature_instance**—Assigns a threat profile to a virtual sensor signature instance:
 - **threat-profile threat_profile**—Assigns Data_Center, Edge, SCADA, or Web_Applications to the signature instance.



Note

Make sure you have created the signature instance you want to apply the threat profile to beforehand in service signature definition mode; otherwise, you receive an error message.

Displaying, Applying, and Removing Threat Profiles

To display, apply, and remove threat profiles, follow these steps:

- Step 1** Log in to the CLI using an account with administrator or operator privileges.
- Step 2** List the threat profiles available on the sensor.

```
sensor# list threat-profiles
Available Threat Profiles:
```

```
Data_Center          In addition to signatures in the default set, Data Center
signature template includes additional signatures that provide broader protection for
server operating systems, web servers, application servers, databases, content management
systems, messaging servers and virtualization systems. This template should be used if
Cisco IPS is primarily used for protecting Data Centers.
```

```
Edge                In addition to signatures in the default set, Internet Edge
signature template includes additional signatures that provide broader protection for
desktop operating systems, web browsers, web technologies and common desktop applications.
This template should be used if Cisco IPS is primarily used for securing an Internet
connection.
```

```
SCADA               In addition to signatures in the default set, SCADA signature
template includes specialized signatures for general SCADA protocol detections and
specific identifiers that address tools and environments common to most device controlled
```

environments. This template should be used if Cisco IPS is primarily used for protecting Industrial Control Systems. You are entitled to use these signatures only if you have purchased an IPS SCADA Signature License.

Web_Applications In addition to signatures in the default set, Web Applications signature template includes additional signatures that provide broader protection for web servers, web development tools and frameworks, content management systems, load balancers and databases. This template should be used if Cisco IPS is primarily used for protecting web server farms.

```
sensor#
```

Step 3 Display the signature templates applied to various virtual sensors.

```
sensor# list signature-definition-configurations
Signature Definition
  Instance   Size   Virtual Sensor   Threat Profile
  sig0       321   vs0              Edge
  sig1       141   N/A              Edge
  sig2       141   N/A              Web_Applications
  sig3       141   N/A              NONE
  sig4       141   N/A              Web_Applications
  sig5       259   N/A              Data_Center
sensor#
```

Step 4 Enter threat profile mode.

```
sensor# configure terminal
sensor(config)# service threat-profile sig0
sensor(config-thr)#
```

Step 5 Apply a threat profile.

```
sensor(config-sig-thr)# threat-profile DATA_CENTER
sensor(config-sig-thr)#
```

Step 6 Verify that the threat profile was added.

```
sensor(config-thr)# show settings
  threat-profile: Data_Center default: None
sensor(config-thr)#
```

Step 7 Remove the threat profile.

```
sensor(config-sig-thr)# no threat-profile
sensor(config-thr)#
```

Step 8 Verify that the threat profile was removed.

```
sensor(config-thr)# show settings
  threat-profile: default: None
sensor(config-thr)#
```

Step 9 Exit threat profile mode.

```
sensor(config-sig-thr)# exit
Apply Changes:[yes]:
```

Step 10 Press **Enter** to apply the changes or enter **no** to discard them.

Displaying a Signature Tuned with a Threat Profile

Use the **show settings** command in service signature definition signatures submode to display whether a signature has been tuned with a threat profile.



Note

If a signature has both threat profile tuning and custom tuning, you cannot return the signature to its default configuration using the **default signatures sig-id sub sig-id** command unless the threat profile is removed from the signature instance. Note that the **default signatures** command still works and removes all custom tunings but preserves the threat profile tunings.

To view a threat-profile tuned signature, follow these steps:

Step 1 Log in to the CLI using an account with administrator privileges.

Step 2 Check to see if a signature has been tuned with a threat profile.

```
sensor(config)# service signature-definition sig1
sensor(config-sig)# signatures 3708 0
sensor(config-sig-sig)# status
sensor(config-sig-sig-sta)# show settings
-----
status
-----
enabled: true <threatProfile> default: false
retired: false <threatProfile> default: true
obsoletes (min: 0, max: 65535, current: 0)
-----
-----
sensor(config-sig-sig-sta)#
```

Step 3 Exit signatures submode.

```
sensor(config-sig-sig-sta)# exit
sensor(config-sig-sig)# exit
sensor(config-sig)#
Apply Changes:[yes]:
```

Step 4 Press **Enter** to apply the changes or enter **no** to discard them.

Displaying Threat Profile Versions

Use the **show version** command in global configuration mode to display the software version on the sensor, the license key status, signature update, and threat profile version.

To display the software versions on the sensor, follow these steps:

Step 1 Log in to the CLI.

Step 2 Verify the software versions on the sensor.

```
sensor# show version
Application Partition:

Cisco Intrusion Prevention System, Version 7.3(1)E4
```

```

Host:
  Realm Keys                key1.0
Signature Definition:
  Signature Update          S741.0      2013-09-10
  Threat Profile Version    2
OS Version:                2.6.29.1
Platform:                  IPS-4360
Serial Number:             FGL1702401M
Licensed, expires:        21-Nov-2014 UTC
Sensor up-time is 22:09.
Using 14372M out of 15943M bytes of available memory (90% usage)
system is using 32.4M out of 160.0M bytes of available disk space (20% usage)
application-data is using 85.6M out of 376.4M bytes of available disk space (24% usage)
boot is using 63.1M out of 70.2M bytes of available disk space (95% usage)
application-log is using 494.0M out of 513.0M bytes of available disk space (96% usage)

MainApp                    C-2013_12_16_14_00_7_3_0_143  (Release)  2013-12-16T14:06:20-0600
Running
AnalysisEngine            C-2013_12_16_14_00_7_3_0_143  (Release)  2013-12-16T14:06:20-0600
Running
CollaborationApp         C-2013_12_16_14_00_7_3_0_143  (Release)  2013-12-16T14:06:20-0600
Running
CLI                      C-2013_12_16_14_00_7_3_0_143  (Release)  2013-12-16T14:06:20-0600

Upgrade History:

  IPS-K9-7.3-1-E4   11:22:07 UTC Sat Jan 19 2013

Recovery Partition Version 1.1 - 7.3(1)E4

Host Certificate Valid from: 09-Oct-2014 to 09-Oct-2016

sensor#

```

Configuring Signature Variables

This section describes signature variables, and contains the following topics:

- [Understanding Signature Variables, page 7-9](#)
- [Creating Signature Variables, page 7-10](#)

Understanding Signature Variables

When you want to use the same value within multiple signatures, use a variable. When you change the value of a variable, that variable is updated in all signatures in which it appears. This saves you from having to change the variable repeatedly as you configure signatures.



Note

You must preface signature variables with a dollar (\$) sign to indicate that you are using a variable rather than a string.

Some variables cannot be deleted because they are necessary to the signature system. If a variable is protected, you cannot select it to edit it. You receive an error message if you try to delete protected variables. You can edit only one variable at a time.

Creating Signature Variables

Use the **variables** command in the signature definition submode to create signature variables.

The following commands apply:

- **variable_name**—Identifies the name assigned to this variable. A valid name can only contain numbers or letters. You can also use a hyphen (-) or underscore (_).
- **ip-addr-range**—Specifies the system-defined variable for grouping IP addresses. The valid values are: A.B.C.D-A.B.C.D[,A.B.C.D-A.B.C.D]
- **web-ports**—Specifies the system-defined variable for ports to look for HTTP traffic. To designate multiple port numbers for a single variable, place a comma between the entries. For example, 80, 3128, 8000, 8010, 8080, 8888, 24326.

Adding, Editing, and Deleting Signature Variables

To add, edit, and delete signature variables, follow these steps:

Step 1 Log in to the CLI using an account with administrator or operator privileges.

Step 2 Enter signature definition submode.

```
sensor# configure terminal
sensor(config)# service signature-definition sig1
```

Step 3 Create a signature variable for a group of IP addresses.

```
sensor(config-sig)# variables IPADD ip-addr-range 10.1.1.1-10.1.1.24
```

Step 4 Edit the signature variable for web ports. WEBPORTS has a predefined set of ports where web servers are running, but you can edit the value. This variable affects all signatures that have web ports. The default is 80, 3128, 8000, 8010, 8080, 8888, 24326.

```
sensor(config-sig)# variables WEBPORTS web-ports 80,3128,8000
```

Step 5 Verify the changes.

```
sensor(config-sig)# show settings
  variables (min: 0, max: 256, current: 2)
  -----
  variable-name: IPADD
  -----
  ip-addr-range: 10.1.1.1-10.1.1.24
  -----
  <protected entry>
  variable-name: WEBPORTS
  -----
  web-ports: 80,3128,8000 default: 80-80,3128-3128,8000-8000,8010-8010,80
  80-8080,8888-8888,24326-24326
  -----
```

Step 6 Delete a variable.

```
sensor(config-sig)# no variables IPADD
```


Step 7 Verify the variable has been deleted.

```
sensor(config-sig)# show settings
  variables (min: 0, max: 256, current: 1)
  -----
  <protected entry>
  variable-name: WEBPORTS
  -----
  web-ports: 80,3128,8000 default: 80-80,3128-3128,8000-8000,8010-8010,80
80-8080,8888-8888,24326-24326
  -----
```

Step 8 Exit signature definition submenu.

```
sensor(config-sig)# exit
Apply Changes?[yes]:
```

Step 9 Press **Enter** to apply the changes or enter **no** to discard them.

Configuring Signatures

This section describes how to configure signature parameters, and contains the following topics:

- [Signature Definition Options, page 7-11](#)
- [Configuring Alert Frequency, page 7-12](#)
- [Configuring Alert Severity, page 7-14](#)
- [Configuring the Event Counter, page 7-15](#)
- [Configuring Signature Fidelity Rating, page 7-17](#)
- [Configuring the Status of Signatures, page 7-18](#)
- [Configuring the Vulnerable OSEs for a Signature, page 7-19](#)
- [Assigning Actions to Signatures, page 7-20](#)
- [Configuring AIC Signatures, page 7-22](#)
- [Configuring IP Fragment Reassembly, page 7-34](#)
- [Configuring TCP Stream Reassembly, page 7-37](#)
- [Configuring IP Logging, page 7-45](#)

Signature Definition Options

The following commands apply to configuring the general parameters of a specific signature:

- **alert-frequency**—Sets the summary options for grouping alerts.
- **alert-severity**—Sets the severity of the alert.
- **engine**—Specifies the signature engine. You can assign actions when you are in the engine submenu.
- **event-counter**—Sets the event count.
- **promisc-delta**—Specifies the delta value used to determine the seriousness of the alert.

**Caution**

We recommend that you do NOT change the promiscuous delta setting for a signature.

Promiscuous delta lowers the risk rating of certain alerts in promiscuous mode. Because the sensor does not know the attributes of the target system and in promiscuous mode cannot deny packets, it is useful to lower the prioritization of promiscuous alerts (based on the lower risk rating) so the administrator can focus on investigating higher risk rating alerts.

In inline mode, the sensor can deny the offending packets and they never reach the target host, so it does not matter if the target was vulnerable. The attack was not allowed on the network and so we do not subtract from the risk rating value.

Signatures that are not service, OS, or application specific have 0 for the promiscuous delta. If the signature is specific to an OS, service, or application, it has a promiscuous delta of 5, 10, or 15 calculated from 5 points for each category.

- **sig-description**—Your description of the signature.
- **sig-fidelity-rating**—Specifies the rating of the fidelity of signature.
- **status**—Sets the status of the signature to enabled or retired.
- **vulnerable-os**—Specifies the list of OS types that are vulnerable to this attack signature.

For More Information

- For the procedure for configuring alert frequency, see [Configuring Alert Frequency, page 7-12](#).
- For more information about signature engines, see [Appendix B, “Signature Engines.”](#)
- For the procedure for assigning actions, see [Assigning Actions to Signatures, page 7-20](#).
- For the procedure for configuring event counts, see [Configuring the Event Counter, page 7-15](#).
- For the procedure for configuring the signature fidelity rating, see [Configuring Signature Fidelity Rating, page 7-17](#).
- For the procedure for enabling and disabling signatures, see [Configuring the Status of Signatures, page 7-18](#).
- For the procedure for configuring vulnerable OSEs, see [Configuring the Vulnerable OSEs for a Signature, page 7-19](#).

Configuring Alert Frequency

Use the **alert-frequency** command in signature definition submode to configure the alert frequency for a signature. The **alert-frequency** command specifies how often the sensor alerts you when this signature is firing.

The following commands apply:

- **sig_id**—Identifies the unique numerical value assigned to this signature. This value lets the sensor identify a particular signature. The value is 1000 to 65000.
- **subsig_id**—Identifies the unique numerical value assigned to this subsignature. A subsignature ID is used to identify a more granular version of a broad signature. The value is 0 to 255.
- **summary-mode**—Specifies the way you want the sensor to group the alerts:
 - **fire-all**—Fires an alert on all events.
 - **fire-once**—Fires an alert only once.

- **global-summarize**—Summarizes an alert so that it only fires once regardless of how many attackers or victims.
- **summarize**—Summarize all the alerts.
- **specify-summary-threshold {yes | no}**—Enables summary threshold mode:
 - **summary-threshold**—Specifies the minimum number of hits the sensor must receive before sending a summary alert for this signature. The value is 0 to 65535.
 - **summary-interval**—Specifies the time in seconds used in each summary alert. The value is 1 to 1000.
- **summary-key**—Specifies the storage type on which to summarize this signature:
 - **Axxx**—Attacker address.
 - **Axxb**—Attacker address and victim port.
 - **AxBx**—Attacker and victim addresses.
 - **AaBb**—Attacker and victim addresses and ports.
 - **xxBx**—Victim address.
- **specify-global-summary-threshold {yes | no}**—(Optional) Enables global summary threshold mode:
 - **global-summary-threshold**—Specifies the threshold number of events to take alert in to global summary. The value is 1 to 65535.

Configuring Alert Frequency

To configure the alert frequency parameters of a signature, follow these steps:

Step 1 Log in to the CLI using an account with administrator or operator privileges.

Step 2 Enter signature definition submode.

```
sensor# configure terminal
sensor(config)# service signature-definition sig1
```

Step 3 Specify the signature you want to configure.

```
sensor(config-sig)# signatures 9000 0
```

Step 4 Enter alert frequency submode.

```
sensor(config-sig-sig)# alert-frequency
```

Step 5 Specify the alert frequency of this signature:

- a. Configure the summary mode to, for example, fire once.

```
sensor(config-sig-sig-ale)# summary-mode fire-once
sensor(config-sig-sig-ale-fir)# specify-global-summary-threshold yes
sensor(config-sig-sig-ale-fir-yes)# global-summary-threshold 3000
sensor(config-sig-sig-ale-fir-yes)# summary-interval 5000
```

- b. Specify the summary key.

```
sensor(config-sig-sig-ale-fir-yes)# exit
sensor(config-sig-sig-ale-fir)# summary-key AxBx
```

- c. Verify the settings.

```
sensor(config-sig-sig-ale-fir)# show settings
fire-once
```

```

-----
summary-key: AxBx default: Axxx
specify-global-summary-threshold
-----
yes
-----
global-summary-threshold: 3000 default: 120
summary-interval: 5000 default: 15
-----
-----
sensor(config-sig-sig-ale-fir)#

```

Step 6 Exit alert-frequency submode.

```

sensor(config-sig-sig-ale-fir)# exit
sensor(config-sig-sig-ale)# exit
sensor(config-sig-sig)# exit
sensor(config-sig)# exit
Apply Changes?[yes]:

```

Step 7 Press **Enter** to apply the changes or enter **no** to discard them.

Configuring Alert Severity

Use the **alert-severity** command in signature definition submode to configure the severity of a signature.

The following commands apply:

- *sig_id*—Identifies the unique numerical value assigned to this signature. This value lets the sensor identify a particular signature. The value is 1000 to 65000.
- *subsig_id*—Identifies the unique numerical value assigned to this subsignature. A subsignature ID is used to identify a more granular version of a broad signature. The value is 0 to 255.
- **alert-severity**—Specifies the severity of the alert:
 - **high**—Dangerous alert.
 - **medium**—Medium level alert (default).
 - **low**—Low level alert.
 - **informational**—Informational alert.

Configuring Alert Severity

To configure the alert severity, follow these steps:

Step 1 Log in to the CLI using an account with administrator or operator privileges.

Step 2 Enter signature definition submode.

```

sensor# configure terminal
sensor(config)# service signature-definition sig1

```

Step 3 Specify the signature you want to configure.

```

sensor(config-sig)# signatures 9000 0

```

Step 4 Assign the alert severity.

```
sensor(config-sig-sig)# alert-severity medium
```

Step 5 Verify the settings.

```
sensor(config-sig-sig)# show settings
<protected entry>
sig-id: 9000
subsig-id: 0
-----
alert-severity: medium default: medium
sig-fidelity-rating: 75 <defaulted>
promisc-delta: 0 <defaulted>
sig-description
-----
sig-name: Back Door Probe (TCP 12345) <defaulted>
sig-string-info: SYN to TCP 12345 <defaulted>
sig-comment: <defaulted>
alert-traits: 0 <defaulted>
release: 40 <defaulted>
-----
vulnerable-os: general-os <defaulted>
engine
-----
atomic-ip
-----
event-action: produce-alert <defaulted>
fragment-status: any <defaulted>
specify-l4-protocol
-----
--MORE--
```

Step 6 Exit signatures submode.

```
sensor(config-sig-sig)# exit
sensor(config-sig)# exit
Apply Changes:[yes]:
```

Step 7 Press **Enter** to apply the changes or enter **no** to discard them.

Configuring the Event Counter

Use the **event-counter** command in signature definition submode to configure how the sensor counts events. For example, you can specify that you want the sensor to send an alert only if the same signature fires 5 times for the same address set.

The following commands apply:

- **event-count**—Specifies the number of times an event must occur before an alert is generated. The valid range is 1 to 65535. The default is 1.
- **event-count-key**—Specifies the storage type on which to count events for this signature:
 - **Axxx**—Attacker address
 - **AxBx**—Attacker and victim addresses
 - **Axxb**—Attacker address and victim port
 - **xxBx**—Victim address

- **AaBb**—Attacker and victim addresses and ports
- **specify-alert-interval [yes | no]**—Enables alert interval:
 - **alert-interval**—Specifies the time in seconds before the event count is reset. The default is 60.

Configuring the Event Counter

To configure event counter, follow these steps:

-
- Step 1** Log in to the CLI using an account with administrator or operator privileges.
- Step 2** Enter signature definition submode.
- ```
sensor# configure terminal
sensor(config)# service signature-definition sig1
```
- Step 3** Specify the signature for which you want to configure event counter.
- ```
sensor(config-sig)# signatures 9000 0
```
- Step 4** Enter event counter submode.
- ```
sensor(config-sig-sig)# event-counter
```
- Step 5** Specify how many times an event must occur before an alert is generated.
- ```
sensor(config-sig-sig-eve)# event-count 2
```
- Step 6** Specify the storage type on which you want to count events for this signature.
- ```
sensor(config-sig-sig-eve)# event-count-key AxBx
```
- Step 7** (Optional) Enable alert interval.
- ```
sensor(config-sig-sig-eve)# specify-alert-interval yes
```
- Step 8** (Optional) Specify the amount of time in seconds before the event count should be reset.
- ```
sensor(config-sig-sig-eve-yes)# alert-interval 30
```
- Step 9** Verify the settings.
- ```
sensor(config-sig-sig-eve-yes)# exit
sensor(config-sig-sig-eve)# show settings
event-counter
-----
event-count: 2 default: 1
event-count-key: AxBx default: Axxx
specify-alert-interval
-----
yes
-----
alert-interval: 30 default: 60
-----
sensor(config-sig-sig-eve)#
```
- Step 10** Exit signatures submode.
- ```
sensor(config-sig-sig-eve)# exit
sensor(config-sig-sig)# exit
sensor(config-sig)# exit
Apply Changes:[yes]:
```

**Step 11** Press **Enter** to apply the changes or enter **no** to discard them.

---

## Configuring Signature Fidelity Rating

Use the **sig-fidelity-rating** command in signature definition submode to configure the signature fidelity rating for a signature.

The following option applies:

- **sig-fidelity-rating**—Identifies the weight associated with how well this signature might perform in the absence of specific knowledge of the target. The valid value is 0 to 100.

### Configuring the Signature Fidelity Rating

To configure the signature fidelity rating for a signature, follow these steps:

---

**Step 1** Log in to the CLI using an account with administrator or operator privileges.

**Step 2** Enter signature definition submode.

```
sensor# configure terminal
sensor(config)# service signature-definition sig0
```

**Step 3** Specify the signature you want to configure.

```
sensor(config-sig)# signatures 12000 0
```

**Step 4** Specify the signature fidelity rating for this signature.

```
sensor(config-sig-sig)# sig-fidelity-rating 50
```

**Step 5** Verify the settings.

```
sensor(config-sig-sig)# show settings
<protected entry>
sig-id: 12000
subsig-id: 0

alert-severity: low <defaulted>
sig-fidelity-rating: 50 default: 85
promisc-delta: 15 <defaulted>
sig-description

sig-name: Gator Spyware Beacon <defaulted>
sig-string-info: /download/ User-Agent: Gator <defaulted>
sig-comment: <defaulted>
alert-traits: 0 <defaulted>
release: 71 <defaulted>

```

**Step 6** Exit signatures submode.

```
sensor(config-sig-sig)# exit
sensor(config-sig)# exit
Apply Changes:[yes]:
```

**Step 7** Press **Enter** to apply the changes or enter **no** to discard them.

---

## Configuring the Status of Signatures

Use the **status** command in signature definition submode to specify the status of a specific signature.

The following commands apply:

- **status**—Identifies whether the signature is enabled, disabled, or retired:
  - **enabled {true | false}**—Enables the signature.
  - **retired {true | false}**—Retires the signature.
  - **obsoletes *signature\_ID***—Shows the other signatures that have been obsoleted by this signature.



### Caution

Activating and retiring signatures can take 30 minutes or longer.

### Changing the Signature Status

To change the status of a signature, follow these steps:

**Step 1** Log in to the CLI using an account with administrator or operator privileges.

**Step 2** Enter signature definition submode.

```
sensor# configure terminal
sensor(config)# service signature-definition sig1
```

**Step 3** Choose the signature you want to configure.

```
sensor(config-sig)# signatures 12000 0
```

**Step 4** Change the status for this signature.

```
sensor(config-sig-sig)# status
sensor(config-sig-sig-sta)# enabled true
```

**Step 5** Verify the settings.

```
sensor(config-sig-sig-sta)# show settings
status

enabled: true default: false
retired: false <defaulted>

sensor(config-sig-sig-sta)#
```

**Step 6** Exit signatures submode.

```
sensor(config-sig-sig-sta)# exit
sensor(config-sig-sig)# exit
sensor(config-sig)# exit
Apply Changes?[yes]:
```

**Step 7** Press **Enter** to apply the changes or enter **no** to discard them.



## Configuring the Vulnerable OSES for a Signature

Use the **vulnerable-os** command in signature definition submode to configure the list of vulnerable OSES for a signature.

The following commands apply:

- **general-os**—Specifies all OS types
- **ios**—Specifies the variants of Cisco IOS
- **mac-os**—Specifies the variants of Macintosh OS
- **netware**—Specifies Netware
- **other**—Specifies any other OS
- **unix**—Specifies the variants of UNIX
- **aix**—Specifies the variants of AIX
- **bsd**—Specifies the variants of BSD
- **hp-ux**—Specifies the variants of HP-UX
- **irix**—Specifies the variants of IRIX
- **linux**—Specifies the variants of Linux
- **solaris**—Specifies the variants of Solaris
- **windows**—Specifies the variants of Microsoft Windows
- **windows-nt-2k-xp**—Specifies the variants of Microsoft NT, 2000, and XP
- **win-nt**—Specifies the specific variants of Windows NT

### Configuring Vulnerable OSES

To configure the vulnerable OSES for a signature, follow these steps:

---

**Step 1** Log in to the CLI using an account with administrator or operator privileges.

**Step 2** Enter signature definition submode.

```
sensor# configure terminal
sensor(config)# service signature-definition sig1
```

**Step 3** Specify the signature you want to configure.

```
sensor(config-sig)# signatures 6000 0
```

**Step 4** Specify the vulnerable OSES for this signature.

```
sensor(config-sig-sig)# vulnerable-os linux|aix
```

**Step 5** Verify the settings.

```
sensor(config-sig-sig)# show settings
sig-id: 60000
subsig-id: 0

alert-severity: medium <defaulted>
sig-fidelity-rating: 75 <defaulted>
promisc-delta: 0 <defaulted>
sig-description

sig-name: My Sig <defaulted>
```

```

sig-string-info: My Sig Info <defaulted>
sig-comment: Sig Comment <defaulted>
alert-traits: 0 <defaulted>
release: custom <defaulted>

vulnerable-os: aix|linux default: general-os
*---> engine

event-counter

event-count: 1 <defaulted>
event-count-key: Axxx <defaulted>
specify-alert-interval

--MORE--

```

**Step 6** Exit signatures submode.

```

sensor(config-sig-sig)# exit
sensor(config-sig)# exit
Apply Changes:[yes]:

```

**Step 7** Press **Enter** to apply the changes or enter **no** to discard them.

## Assigning Actions to Signatures

Use the **event-action** command in signature definition submode to configure the actions the sensor takes when the signature fires.

The following commands apply:

- **event-action**—Specifies the type of event action the sensor should perform:
  - **deny-attacker-inline** (inline only)—Does not transmit this packet and future packets from the attacker address for a specified period of time.
  - **deny-attacker-service-pair-inline** (inline only)—Does not transmit this packet and future packets on the attacker address victim port pair for a specified period of time.
  - **deny-attacker-victim-pair-inline** (inline only)—Does not transmit this packet and future packets on the attacker/victim address pair for a specified period of time.
  - **deny-connection-inline** (inline only)—Does not transmit this packet and future packets on the TCP flow.
  - **deny-packet-inline** (inline only)—Does not transmit this packet.
  - **log-attacker-packets**—Starts IP logging of packets containing the attacker address.
  - **log-pair-packets**—Starts IP logging of packets containing the attacker-victim address pair.
  - **log-victim-packets**—Starts IP logging of packets containing the victim address.
  - **produce-alert** —Writes the event to the Event Store as an alert.
  - **produce-verbose-alert**—Includes an encoded dump (possibly truncated) of the offending packet in the alert.
  - **request-block-connection**—Sends a request to the ARC to block this connection.
  - **request-block-host**—Sends a request to the ARC to block this attacker host.

- **request-rate-limit**—Sends a rate limit request to the ARC to perform rate limiting.
- **request-snmp-trap**—Sends a request to the Notification Application component of the sensor to perform SNMP notification.
- **reset-tcp-connection**—Sends TCP resets to hijack and terminate the TCP flow.
- **modify-packet-inline**— Modifies packet data to remove ambiguity about what the end point might do with the packet.
- **event-action-settings**—Enables the **external-rate-limit-type**:
  - **none**—No rate limiting configured.
  - **percentage**—Specifies the rate limit by traffic percentage (**external-rate-limit-percentage**).

### Configuring Event Actions

To configure event actions and event action settings for a signature, follow these steps:

**Step 1** Log in to the CLI using an account with administrator privileges.

**Step 2** Enter signature definition mode.

```
sensor# configure terminal
sensor(config)# service signature-definition sig0
sensor(config-sig)#
```

**Step 3** Specify the signature you want to configure.

```
sensor(config-sig)# signatures 1200 0
```

**Step 4** Specify the signature engine (for signature 1200 it is the Normalizer engine).

```
sensor(config-sig-sig)# engine normalizer
```

**Step 5** Configure the event action.

```
sensor(config-sig-sig-nor)# event-action produce-alert|request-snmp-trap
```



**Note** Each time you configure the event actions for a signature, you overwrite the previous configuration. For example, if you always want to produce an alert when the signature is fired, you must configure it along with the other event actions you want. Use the | symbol to add more than one event action, for example, **product-alert|deny-packet-inline|request-snmp-trap**.

**Step 6** Verify the settings.

```
sensor(config-sig-sig-nor)# show settings
normalizer

event-action: produce-alert|request-snmp-trap default:
produce-alert|deny-packet-inline
```

**Step 7** Specify the percentage for rate limiting.

```
sensor(config-sig-sig-nor)# event-action-settings
sensor(config-sig-sig-nor-eve)# external-rate-limit-type percentage
sensor(config-sig-sig-nor-eve-per)# external-rate-limit-percentage 50
```

**Step 8** Verify the settings.

```
sensor(config-sig-sig-nor-eve-per)# show settings
percentage
```

```

external-rate-limit-percentage: 50 default: 100

```

**Step 9** Exit event action submode.

```
sensor(config-sig-sig-nor-eve-per)# exit
sensor(config-sig-sig-nor-eve)# exit
sensor(config-sig-sig-nor)# exit
sensor(config-sig-sig)# exit
sensor(config-sig)# exit
Apply Changes:[yes]:
```

**Step 10** Press **Enter** to apply the changes or enter **no** to discard them.

---

#### For More Information

For a detailed description of event actions, see [Event Actions, page 8-5](#).

## Configuring AIC Signatures

This section describes the Application Inspection and Control (AIC) signatures and how to configure them. It contains the following topics:

- [Understanding the AIC Engine, page 7-22](#)
- [AIC Engine and Sensor Performance, page 7-23](#)
- [Configuring the Application Policy, page 7-23](#)
- [AIC Request Method Signatures, page 7-25](#)
- [AIC MIME Define Content Type Signatures, page 7-26](#)
- [AIC Transfer Encoding Signatures, page 7-30](#)
- [AIC FTP Commands Signatures, page 7-31](#)
- [Creating an AIC Signature, page 7-32](#)

## Understanding the AIC Engine

AIC provides thorough analysis of web traffic. It provides granular control over HTTP sessions to prevent abuse of the HTTP protocol. It allows administrative control over applications, such as instant messaging and gotomypc, that try to tunnel over specified ports. Inspection and policy checks for P2P and instant messaging are possible if these applications are running over HTTP. AIC also provides a way to inspect FTP traffic and control the commands being issued. You can enable or disable the predefined signatures or you can create policies through custom signatures.



#### Note

The AIC engines run when HTTP traffic is received on AIC web ports. If traffic is web traffic, but not received on the AIC web ports, the Service HTTP engine is executed. AIC inspection can be on any port if it is configured as an AIC web port and the traffic to be inspected is HTTP traffic.

---

AIC has the following categories of signatures:

- HTTP request method
  - Define request method
  - Recognized request methods
- MIME type
  - Define content type
  - Recognized content type
- Define web traffic policy

There is one predefined signature, 12674, that specifies the action to take when noncompliant HTTP traffic is seen. The parameter Alarm on Non HTTP Traffic enables the signature. By default this signature is enabled.
- Transfer encodings
  - Associate an action with each method
  - List methods recognized by the sensor
  - Specify which actions need to be taken when a chunked encoding error is seen
- FTP commands
  - Associates an action with an FTP command.

#### For More Information

- For a list of signature IDs and descriptions for these signatures, see [AIC Request Method Signatures, page 7-25](#), [AIC MIME Define Content Type Signatures, page 7-26](#), [AIC Transfer Encoding Signatures, page 7-30](#), and [AIC FTP Commands Signatures, page 7-31](#).
- For the procedure for creating a custom MIME signature, see [Creating an AIC Signature, page 7-32](#).

## AIC Engine and Sensor Performance

Application policy enforcement is a unique sensor feature. Rather than being based on traditional IPS technologies that inspect for exploits, vulnerabilities, and anomalies, AIC policy enforcement is designed to enforce HTTP and FTP service policies. The inspection work required for this policy enforcement is extreme compared with traditional IPS inspection work. A large performance penalty is associated with using this feature. When AIC is enabled, the overall bandwidth capacity of the sensor is reduced.

AIC policy enforcement is disabled in the IPS default configuration. If you want to activate AIC policy enforcement, we highly recommend that you carefully choose the exact policies of interest and disable those you do not need. Also, if your sensor is near its maximum inspection load capacity, we recommend that you not use this feature since it can oversubscribe the sensor. We recommend that you use the adaptive security appliance firewall to handle this type of policy enforcement.

## Configuring the Application Policy

Use the **application-policy** command in signature definition submode to enable the web AIC feature. You can configure the sensor to provide Layer 4 to Layer 7 packet inspection to prevent malicious attacks related to web and FTP services.

The following commands apply:

- **ftp-enable {true | false}**—Enables protection for FTP services. Set to true to require the sensor to inspect FTP traffic. The default is false.
- **http-policy**—Enables inspection of HTTP traffic:
  - **aic-web-ports**—Specifies the variable for ports to look for AIC traffic. The valid range is 0 to 65535. A comma-separated list of integer ranges a-b[,c-d] within 0-65535. The second number in the range must be greater than or equal to the first number. The default is 80-80,3128-3128,8000-8000,8010-8010,8080-8080,8888-8888,24326-24326.
  - **http-enable [true | false]**—Enables protection for web services. Set to true to require the sensor to inspect HTTP traffic for compliance with the RFC. The default is false.
  - **max-outstanding-http-requests-per-connection**—Specifies the maximum allowed HTTP requests per connection. The valid value is 1 to 16. The default is 10.

### Configuring the Application Policy

To configure the application policy, follow these steps:

---

**Step 1** Log in to the CLI using an account with administrator or operator privileges.

**Step 2** Enter application policy submode.

```
sensor# configure terminal
sensor(config)# service signature-definition sig1
sensor(config-sig)# application-policy
```

**Step 3** Enable inspection of FTP traffic.

```
sensor(config-sig-app)# ftp-enable true
```

**Step 4** Configure the HTTP application policy:

a. Enter HTTP application policy submode.

```
sensor(config-sig-app)# http-policy
```

b. Enable HTTP application policy enforcement.

```
sensor(config-sig-app-http)# http-enable true
```

c. Specify the number of outstanding HTTP requests per connection that can be outstanding without having received a response from the server.

```
sensor(config-sig-app-http)# max-outstanding-http-requests-per-connection 5
```

d. Edit the AIC ports.

```
sensor(config-sig-app-http)# aic-web-ports 80-80,3128-3128
```

**Step 5** Verify your settings.

```
sensor(config-sig-app-http)# exit
sensor(config-sig-app)# show settings
application-policy

http-policy

http-enable: true default: false
max-outstanding-http-requests-per-connection: 5 default: 10
aic-web-ports: 80-80,3128-3128 default: 80-80,3128-3128,8000-8000,8010-
8010,8080-8080,8888-8888,24326-24326
```

```

ftp-enable: true default: false

sensor(config-sig-app)#

```

**Step 6** Exit signature definition submenu.

```

sensor(config-sig-app)# exit
sensor(config-sig)# exit
Apply Changes:[yes]:

```

**Step 7** Press **Enter** to apply the changes or enter **no** to discard them.

## AIC Request Method Signatures

The HTTP request method has two categories of signatures:

- Define request method—Allows actions to be associated with request methods. You can expand and modify the signatures (Define Request Method).
- Recognized request methods—Lists methods that are recognized by the sensor (Recognized Request Methods).

Table 7-1 lists the predefined define request method signatures. Enable the signatures that have the predefined method you need.

**Table 7-1** Request Method Signatures

| Signature ID | Define Request Method         |
|--------------|-------------------------------|
| 12676        | Request Method Not Recognized |
| 12677        | Define Request Method PUT     |
| 12678        | Define Request Method CONNECT |
| 12679        | Define Request Method DELETE  |
| 12680        | Define Request Method GET     |
| 12681        | Define Request Method HEAD    |
| 12682        | Define Request Method OPTIONS |
| 12683        | Define Request Method POST    |
| 12685        | Define Request Method TRACE   |
| 12695        | Define Request Method INDEX   |
| 12696        | Define Request Method MOVE    |
| 12697        | Define Request Method MKDIR   |
| 12698        | Define Request Method COPY    |
| 12699        | Define Request Method EDIT    |
| 12700        | Define Request Method UNEDIT  |
| 12701        | Define Request Method SAVE    |
| 12702        | Define Request Method LOCK    |
| 12703        | Define Request Method UNLOCK  |

**Table 7-1 Request Method Signatures (continued)**

| Signature ID | Define Request Method                  |
|--------------|----------------------------------------|
| 12704        | Define Request Method REVLABEL         |
| 12705        | Define Request Method REVLOG           |
| 12706        | Define Request Method REVADD           |
| 12707        | Define Request Method REVNUM           |
| 12708        | Define Request Method SETATTRIBUTE     |
| 12709        | Define Request Method GETATTRIBUTENAME |
| 12710        | Define Request Method GETPROPERTIES    |
| 12711        | Define Request Method STARTENV         |
| 12712        | Define Request Method STOPREV          |

**For More Information**

For the procedure for enabling signatures, see [Configuring the Status of Signatures, page 7-18](#).

**AIC MIME Define Content Type Signatures**

There are two policies associated with MIME types:

- Define content type—Associates specific actions for the following cases (Define Content Type):
  - Deny a specific MIME type, such as an image/jpeg
  - Message size violation
  - MIME-type mentioned in header and body do not match
- Recognized content type (Recognized Content Type)

[Table 7-2](#) lists the predefined define content type signatures. Enable the signatures that have the predefined content type you need. You can also create custom define content type signatures.

**Table 7-2 Define Content Type Signatures**

| Signature ID | Signature Description                                       |
|--------------|-------------------------------------------------------------|
| 12621        | Content Type image/gif Invalid Message Length               |
| 12622 2      | Content Type image/png Verification Failed                  |
| 12623 0      | Content Type image/tiff Header Check                        |
| 12623 1      | Content Type image/tiff Invalid Message Length              |
| 12623 2      | Content Type image/tiff Verification Failed                 |
| 12624 0      | Content Type image/x-3ds Header Check                       |
| 12624 1      | Content Type image/x-3ds Invalid Message Length             |
| 12624 2      | Content Type image/x-3ds Verification Failed                |
| 12626 0      | Content Type image/x-portable-bitmap Header Check           |
| 12626 1      | Content Type image/x-portable-bitmap Invalid Message Length |
| 12626 2      | Content Type image/x-portable-bitmap Verification Failed    |



**Table 7-2 Define Content Type Signatures (continued)**

| <b>Signature ID</b> | <b>Signature Description</b>                                 |
|---------------------|--------------------------------------------------------------|
| 12627 0             | Content Type image/x-portable-graymap Header Check           |
| 12627 1             | Content Type image/x-portable-graymap Invalid Message Length |
| 12627 2             | Content Type image/x-portable-graymap Verification Failed    |
| 12628 0             | Content Type image/jpeg Header Check                         |
| 12628 1             | Content Type image/jpeg Invalid Message Length               |
| 12628 2             | Content Type image/jpeg Verification Failed                  |
| 12629 0             | Content Type image/cgf Header Check                          |
| 12629 1             | Content Type image/cgf Invalid Message Length                |
| 12631 0             | Content Type image/x-xpm Header Check                        |
| 12631 1             | Content Type image/x-xpm Invalid Message Length              |
| 12633 0             | Content Type audio/midi Header Check                         |
| 12633 1             | Content Type audio/midi Invalid Message Length               |
| 12633 2             | Content Type audio/midi Verification Failed                  |
| 12634 0             | Content Type audio/basic Header Check                        |
| 12634 1             | Content Type audio/basic Invalid Message Length              |
| 12634 2             | Content Type audio/basic Verification Failed                 |
| 12635 0             | Content Type audio/mpeg Header Check                         |
| 12635 1             | Content Type audio/mpeg Invalid Message Length               |
| 12635 2             | Content Type audio/mpeg Verification Failed                  |
| 12636 0             | Content Type audio/x-adpcm Header Check                      |
| 12636 1             | Content Type audio/x-adpcm Invalid Message Length            |
| 12636 2             | Content Type audio/x-adpcm Verification Failed               |
| 12637 0             | Content Type audio/x-aiff Header Check                       |
| 12637 1             | Content Type audio/x-aiff Invalid Message Length             |
| 12637 2             | Content Type audio/x-aiff Verification Failed                |
| 12638 0             | Content Type audio/x-ogg Header Check                        |
| 12638 1             | Content Type audio/x-ogg Invalid Message Length              |
| 12638 2             | Content Type audio/x-ogg Verification Failed                 |
| 12639 0             | Content Type audio/x-wav Header Check                        |
| 12639 1             | Content Type audio/x-wav Invalid Message Length              |
| 12639 2             | Content Type audio/x-wav Verification Failed                 |
| 12641 0             | Content Type text/html Header Check                          |
| 12641 1             | Content Type text/html Invalid Message Length                |
| 12641 2             | Content Type text/html Verification Failed                   |
| 12642 0             | Content Type text/css Header Check                           |
| 12642 1             | Content Type text/css Invalid Message Length                 |
| 12643 0             | Content Type text/plain Header Check                         |
| 12643 1             | Content Type text/plain Invalid Message Length               |
| 12644 0             | Content Type text/richtext Header Check                      |
| 12644 1             | Content Type text/richtext Invalid Message Length            |
| 12645 0             | Content Type text/sgml Header Check                          |
| 12645 1             | Content Type text/sgml Invalid Message Length                |
| 12645 2             | Content Type text/sgml Verification Failed                   |

**Table 7-2** Define Content Type Signatures (continued)

| <b>Signature ID</b> | <b>Signature Description</b>                                 |
|---------------------|--------------------------------------------------------------|
| 12627 0             | Content Type image/x-portable-graymap Header Check           |
| 12627 1             | Content Type image/x-portable-graymap Invalid Message Length |
| 12627 2             | Content Type image/x-portable-graymap Verification Failed    |
| 12628 0             | Content Type image/jpeg Header Check                         |
| 12628 1             | Content Type image/jpeg Invalid Message Length               |
| 12628 2             | Content Type image/jpeg Verification Failed                  |
| 12629 0             | Content Type image/cgf Header Check                          |
| 12629 1             | Content Type image/cgf Invalid Message Length                |
| 12631 0             | Content Type image/x-xpm Header Check                        |
| 12631 1             | Content Type image/x-xpm Invalid Message Length              |
| 12633 0             | Content Type audio/midi Header Check                         |
| 12633 1             | Content Type audio/midi Invalid Message Length               |
| 12633 2             | Content Type audio/midi Verification Failed                  |
| 12634 0             | Content Type audio/basic Header Check                        |
| 12634 1             | Content Type audio/basic Invalid Message Length              |
| 12634 2             | Content Type audio/basic Verification Failed                 |
| 12635 0             | Content Type audio/mpeg Header Check                         |
| 12635 1             | Content Type audio/mpeg Invalid Message Length               |
| 12635 2             | Content Type audio/mpeg Verification Failed                  |
| 12636 0             | Content Type audio/x-adpcm Header Check                      |
| 12636 1             | Content Type audio/x-adpcm Invalid Message Length            |
| 12636 2             | Content Type audio/x-adpcm Verification Failed               |
| 12637 0             | Content Type audio/x-aiff Header Check                       |
| 12637 1             | Content Type audio/x-aiff Invalid Message Length             |
| 12637 2             | Content Type audio/x-aiff Verification Failed                |
| 12638 0             | Content Type audio/x-ogg Header Check                        |
| 12638 1             | Content Type audio/x-ogg Invalid Message Length              |
| 12638 2             | Content Type audio/x-ogg Verification Failed                 |
| 12639 0             | Content Type audio/x-wav Header Check                        |
| 12639 1             | Content Type audio/x-wav Invalid Message Length              |
| 12639 2             | Content Type audio/x-wav Verification Failed                 |
| 12641 0             | Content Type text/html Header Check                          |
| 12641 1             | Content Type text/html Invalid Message Length                |
| 12641 2             | Content Type text/html Verification Failed                   |
| 12642 0             | Content Type text/css Header Check                           |
| 12642 1             | Content Type text/css Invalid Message Length                 |
| 12643 0             | Content Type text/plain Header Check                         |
| 12643 1             | Content Type text/plain Invalid Message Length               |
| 12644 0             | Content Type text/richtext Header Check                      |
| 12644 1             | Content Type text/richtext Invalid Message Length            |
| 12645 0             | Content Type text/sgml Header Check                          |
| 12645 1             | Content Type text/sgml Invalid Message Length                |
| 12645 2             | Content Type text/sgml Verification Failed                   |

**Table 7-2** Define Content Type Signatures (continued)

| <b>Signature ID</b> | <b>Signature Description</b>                                      |
|---------------------|-------------------------------------------------------------------|
| 12646 0             | Content Type text/xml Header Check                                |
| 12646 1             | Content Type text/xml Invalid Message Length                      |
| 12646 2             | Content Type text/xml Verification Failed                         |
| 12648 0             | Content Type video/flc Header Check                               |
| 12648 1             | Content Type video/flc Invalid Message Length                     |
| 12648 2             | Content Type video/flc Verification Failed                        |
| 12649 0             | Content Type video/mpeg Header Check                              |
| 12649 1             | Content Type video/mpeg Invalid Message Length                    |
| 12649 2             | Content Type video/mpeg Verification Failed                       |
| 12650 0             | Content Type text/xmcd Header Check                               |
| 12650 1             | Content Type text/xmcd Invalid Message Length                     |
| 12651 0             | Content Type video/quicktime Header Check                         |
| 12651 1             | Content Type video/quicktime Invalid Message Length               |
| 12651 2             | Content Type video/quicktime Verification Failed                  |
| 12652 0             | Content Type video/sgi Header Check                               |
| 12652 1             | Content Type video/sgi Verification Failed                        |
| 12653 0             | Content Type video/x-avi Header Check                             |
| 12653 1             | Content Type video/x-avi Invalid Message Length                   |
| 12654 0             | Content Type video/x-fli Header Check                             |
| 12654 1             | Content Type video/x-fli Invalid Message Length                   |
| 12654 2             | Content Type video/x-fli Verification Failed                      |
| 12655 0             | Content Type video/x-mng Header Check                             |
| 12655 1             | Content Type video/x-mng Invalid Message Length                   |
| 12655 2             | Content Type video/x-mng Verification Failed                      |
| 12656 0             | Content Type application/x-msvideo Header Check                   |
| 12656 1             | Content Type application/x-msvideo Invalid Message Length         |
| 12656 2             | Content Type application/x-msvideo Verification Failed            |
| 12658 0             | Content Type application/ms-word Header Check                     |
| 12658 1             | Content Type application/ms-word Invalid Message Length           |
| 12659 0             | Content Type application/octet-stream Header Check                |
| 12659 1             | Content Type application/octet-stream Invalid Message Length      |
| 12660 0             | Content Type application/postscript Header Check                  |
| 12660 1             | Content Type application/postscript Invalid Message Length        |
| 12660 2             | Content Type application/postscript Verification Failed           |
| 12661 0             | Content Type application/vnd.ms-excel Header Check                |
| 12661 1             | Content Type application/vnd.ms-excel Invalid Message Length      |
| 12662 0             | Content Type application/vnd.ms-powerpoint Header Check           |
| 12662 1             | Content Type application/vnd.ms-powerpoint Invalid Message Length |
| 12663 0             | Content Type application/zip Header Check                         |
| 12663 1             | Content Type application/zip Invalid Message Length               |
| 12663 2             | Content Type application/zip Verification Failed                  |

**Table 7-2** Define Content Type Signatures (continued)

| Signature ID | Signature Description                                          |
|--------------|----------------------------------------------------------------|
| 12664 0      | Content Type application/x-gzip Header Check                   |
| 12664 1      | Content Type application/x-gzip Invalid Message Length         |
| 12664 2      | Content Type application/x-gzip Verification Failed            |
| 12665 0      | Content Type application/x-java-archive Header Check           |
| 12665 1      | Content Type application/x-java-archive Invalid Message Length |
| 12666 0      | Content Type application/x-java-vm Header Check                |
| 12666 1      | Content Type application/x-java-vm Invalid Message Length      |
| 12667 0      | Content Type application/pdf Header Check                      |
| 12667 1      | Content Type application/pdf Invalid Message Length            |
| 12667 2      | Content Type application/pdf Verification Failed               |
| 12668 0      | Content Type unknown Header Check                              |
| 12668 1      | Content Type unknown Invalid Message Length                    |
| 12669 0      | Content Type image/x-bitmap Header Check                       |
| 12669 1      | Content Type image/x-bitmap Invalid Message Length             |
| 12673 0      | Recognized content type                                        |

**For More Information**

- For the procedure for enabling signatures, see [Configuring the Status of Signatures, page 7-18](#).
- For the procedure for creating an ACI signature, see [Creating an AIC Signature, page 7-32](#).

**AIC Transfer Encoding Signatures**

There are three policies associated with transfer encoding:

- Associate an action with each method (Define Transfer Encoding)
- List methods recognized by the sensor (Recognized Transfer Encodings)
- Specify which actions need to be taken when a chunked encoding error is seen (Chunked Transfer Encoding Error)

[Table 7-3](#) lists the predefined transfer encoding signatures. Enable the signatures that have the predefined transfer encoding method you need.

**Table 7-3** Transfer Encoding Signatures

| Signature ID | Transfer Encoding Method          |
|--------------|-----------------------------------|
| 12686        | Recognized Transfer Encoding      |
| 12687        | Define Transfer Encoding Deflate  |
| 12688        | Define Transfer Encoding Identity |
| 12689        | Define Transfer Encoding Compress |
| 12690        | Define Transfer Encoding GZIP     |
| 12693        | Define Transfer Encoding Chunked  |
| 12694        | Chunked Transfer Encoding Error   |

**For More Information**

For the procedure for enabling signatures, see [Configuring the Status of Signatures, page 7-18](#).

**AIC FTP Commands Signatures**

[Table 7-4](#) lists the predefined FTP commands signatures. Enable the signatures that have the predefined FTP command you need.

**Table 7-4** *FTP Commands Signatures*

| <b>Signature ID</b> | <b>FTP Command</b>       |
|---------------------|--------------------------|
| 12900               | Unrecognized FTP command |
| 12901               | Define FTP command abor  |
| 12902               | Define FTP command acct  |
| 12903               | Define FTP command allo  |
| 12904               | Define FTP command appe  |
| 12905               | Define FTP command cdup  |
| 12906               | Define FTP command cwd   |
| 12907               | Define FTP command dele  |
| 12908               | Define FTP command help  |
| 12909               | Define FTP command list  |
| 12910               | Define FTP command mkd   |
| 12911               | Define FTP command mode  |
| 12912               | Define FTP command nlst  |
| 12913               | Define FTP command noop  |
| 12914               | Define FTP command pass  |
| 12915               | Define FTP command pasv  |
| 12916               | Define FTP command port  |
| 12917               | Define FTP command pwd   |
| 12918               | Define FTP command quit  |
| 12919               | Define FTP command rein  |
| 12920               | Define FTP command rest  |
| 12921               | Define FTP command retr  |
| 12922               | Define FTP command rmd   |
| 12923               | Define FTP command rnfr  |
| 12924               | Define FTP command rnto  |
| 12925               | Define FTP command site  |
| 12926               | Define FTP command smnt  |
| 12927               | Define FTP command stat  |
| 12928               | Define FTP command stor  |
| 12929               | Define FTP command stou  |

Table 7-4 FTP Commands Signatures (continued)

| Signature ID | FTP Command             |
|--------------|-------------------------|
| 12930        | Define FTP command stru |
| 12931        | Define FTP command syst |
| 12932        | Define FTP command type |
| 12933        | Define FTP command user |

**For More Information**

For the procedure for enabling signatures, see [Configuring the Status of Signatures, page 7-18](#).

## Creating an AIC Signature

**Caution**

A custom signature can affect the performance of your sensor. Test the custom signature against a baseline sensor performance for your network to determine the overall impact of the signature.

The following example demonstrates how to create a MIME-type signature based on the AIC engine.

The following commands apply:

- **event-action**—Specifies the action(s) to perform when alert is triggered:
  - **deny-attacker-inline** (inline only)—Does not transmit this packet and future packets from the attacker address for a specified period of time.
  - **deny-attacker-service-pair-inline** (inline only)—Does not transmit this packet and future packets on the attacker address victim port pair for a specified period of time.
  - **deny-attacker-victim-pair-inline** (inline only)—Does not transmit this packet and future packets on the attacker/victim address pair for a specified period of time.
  - **deny-connection-inline** (inline only)—Does not transmit this packet and future packets on the TCP flow.
  - **deny-packet-inline** (inline only)—Does not transmit this packet.
  - **log-attacker-packets**—Starts IP logging of packets containing the attacker address.
  - **log-pair-packets**—Starts IP logging of packets containing the attacker-victim address pair.
  - **log-victim-packets**—Starts IP logging of packets containing the victim address.
  - **produce-alert** —Writes the event to the Event Store as an alert.
  - **produce-verbose-alert**—Includes an encoded dump (possibly truncated) of the offending packet in the alert.
  - **request-block-connection**—Sends a request to the ARC to block this connection.
  - **request-block-host**—Sends a request to the ARC to block this attacker host.
  - **request-rate-limit**—Sends a rate limit request to the ARC to perform rate limiting.
  - **request-snmp-trap**—Sends a request to the Notification Application component of the sensor to perform SNMP notification.
  - **reset-tcp-connection**—Sends TCP resets to hijack and terminate the TCP flow.

- **modify-packet-inline**—Modifies packet data to remove ambiguity about what the end point might do with the packet.
- **no**—Removes an entry or selection setting
- **signature-type**—Specifies the type of signature desired:
  - **content-types**—Content-types.
  - **define-web-traffic-policy**—Defines web traffic policy.
  - **max-outstanding-requests-overrun**—Inspects for large number of outstanding HTTP requests.
  - **msg-body-pattern**—Message body pattern.
  - **request-methods**—Signature types that deal with request methods.
  - **transfer-encodings**—Signature types that deal with transfer encodings.

### Defining a MIME-Type Policy Signature

To define a MIME-type policy signature, follow these steps:

---

**Step 1** Log in to the CLI using an account with administrator or operator privileges.

**Step 2** Enter application policy enforcement submode.

```
sensor# configure terminal
sensor(config)# service signature-definition sig1
sensor(config-sig)# signatures 60001 0
sensor(config-sig-sig)# engine application-policy-enforcement-http
```

**Step 3** Specify the event action.

```
sensor(config-sig-sig-app)# event-action produce-alert|log-pair-packets
```

**Step 4** Define the signature type.

```
sensor(config-sig-sig-app)# signature-type content-type define-content-type
```

**Step 5** Define the content type.

```
sensor(config-sig-sig-app-def)# name MyContent
```

**Step 6** Verify your settings.

```
sensor(config-sig-sig-app-def)# show settings
-> define-content-type

 name: MyContent
*---> content-type-details

sensor(config-sig-sig-app-def)#
```

**Step 7** Exit signatures submode.

```
sensor(config-sig-sig-app-def)# exit
sensor(config-sig-sig-app)# exit
sensor(config-sig-sig)# exit
sensor(config-sig)# exit
Apply Changes:[yes]:
```

**Step 8** Press **Enter** to apply the changes or enter **no** to discard them.

## Configuring IP Fragment Reassembly

This section describes IP fragment reassembly, lists the IP fragment reassembly signatures with the configurable parameters, describes how to configure these parameters, and how to configure the method for IP fragment reassembly. It contains the following topics:

- [Understanding IP Fragment Reassembly, page 7-34](#)
- [IP Fragment Reassembly Signatures and Configurable Parameters, page 7-34](#)
- [Configuring IP Fragment Reassembly Parameters, page 7-36](#)
- [Configuring the Method for IP Fragment Reassembly, page 7-36](#)

## Understanding IP Fragment Reassembly

You can configure the sensor to reassemble a datagram that has been fragmented over multiple packets. You can specify boundaries that the sensor uses to determine how many datagram fragments it reassembles and how long to wait for more fragments of a datagram. The goal is to ensure that the sensor does not allocate all its resources to datagrams that cannot be completely reassembled, either because the sensor missed some frame transmissions or because an attack has been launched that is based on generating random fragmented datagrams.



### Note

You configure the IP fragment reassembly per signature.

## IP Fragment Reassembly Signatures and Configurable Parameters

[Table 7-5](#) lists IP fragment reassembly signatures with the parameters that you can configure for IP fragment reassembly. The IP fragment reassembly signatures are part of the Normalizer engine.

**Table 7-5** *IP Fragment Reassembly Signatures*

| Signature ID and Name                            | Description                                                                                                           | Parameter With Default Value and Range       | Default Action                                |
|--------------------------------------------------|-----------------------------------------------------------------------------------------------------------------------|----------------------------------------------|-----------------------------------------------|
| 1200 IP Fragmentation Buffer Full                | Fires when the total number of fragments in the system exceeds the threshold set by Max Fragments.                    | Specify Max Fragments 10000 (0-42000)        | Deny Packet Inline Produce Alert <sup>1</sup> |
| 1201 IP Fragment Overlap                         | Fires when the fragments queued for a datagram overlap each other.                                                    | — <sup>2</sup>                               | Deny Packet Inline Produce Alert <sup>1</sup> |
| 1202 IP Fragment Overrun - Datagram Too Long     | Fires when the fragment data (offset and size) exceeds the threshold set with Max Datagram Size.                      | Specify Max Datagram Size 65536 (2000-65536) | Deny Packet Inline Produce Alert <sup>3</sup> |
| 1203 IP Fragment Overwrite - Data is Overwritten | Fires when the fragments queued for a datagram overlap each other and the overlapping data is different. <sup>4</sup> | —                                            | Deny Packet Inline Produce Alert <sup>5</sup> |



Table 7-5 IP Fragment Reassembly Signatures (continued)

| Signature ID and Name                             | Description                                                                                                           | Parameter With Default Value and Range                                    | Default Action                                   |
|---------------------------------------------------|-----------------------------------------------------------------------------------------------------------------------|---------------------------------------------------------------------------|--------------------------------------------------|
| 1204 IP Fragment Missing Initial Fragment         | Fires when the datagram is incomplete and missing the initial fragment.                                               | —                                                                         | Deny Packet Inline<br>Produce Alert <sup>6</sup> |
| 1205 IP Fragment Too Many Datagrams               | Fires when the total number of partial datagrams in the system exceeds the threshold set by Max Partial Datagrams.    | Specify Max Partial Datagrams 1000 (0-10000)                              | Deny Packet Inline<br>Produce Alert <sup>7</sup> |
| 1206 IP Fragment Too Small                        | Fires when there are more than Max Small Frags of a size less than Min Fragment Size in one datagram. <sup>8</sup>    | Specify Max Small Frags 2 (8-1500)<br>Specify Min Fragment Size 400 (1-8) | Deny Packet Inline<br>Produce Alert <sup>9</sup> |
| 1207 IP Fragment Too Many Fragments in a Datagram | Fires when there are more than Max Fragments per Datagram in one datagram.                                            | Specify Max Fragments per Datagram 170 (0-8192)                           | Deny Packet Inline<br>Produce Alert <sup>6</sup> |
| 1208 IP Fragment Incomplete Datagram              | Fires when all of the fragments for a datagram have not arrived during the Fragment Reassembly Timeout. <sup>10</sup> | Specify Fragment Reassembly Timeout 60 (0-360)                            | Deny Packet Inline<br>Produce Alert <sup>6</sup> |
| 1225 Fragment Flags Invalid                       | Fires when a bad combination of fragment flags is detected.                                                           | — <sup>11</sup>                                                           | —                                                |

1. Modify Packet Inline and Deny Connection Inline have no effect on this signature. Deny Packet Inline drops the packets and all associated fragments for this datagram. If you disable this signature, the default values are still used and packets are dropped (inline mode) or not analyzed (promiscuous mode) and no alert is sent.
2. This signature does not fire when the datagram is an exact duplicate. Exact duplicates are dropped in inline mode regardless of the settings. Modify Packet Inline removes the overlapped data from all but one fragment so there is no ambiguity about how the endpoint treats the datagram. Deny Connection Inline has no effect on this signature. Deny Packet Inline drops the packet and all associated fragments for this datagram.
3. Modify Packet Inline and Deny Connection Inline have no effect on this signature. Deny Packet Inline drops the packet and all associated fragments for this datagram. Regardless of the actions set the datagram is not processed by the IPS if the datagram is larger than the Max Datagram size.
4. This is a very unusual event.
5. Modify Packet Inline removes the overlapped data from all but one fragment so there is no ambiguity about how the endpoint treats the datagram. Deny Connection Inline has no effect on this signature. Deny Packet Inline drops the packets and all associated fragments for this datagram.
6. IPS does not inspect a datagram missing the first fragments regardless of the settings. Modify Packet Inline and Deny Connection Inline have no effect on this signature. Deny Packet Inline drops the packet and all associated fragments for this datagram.
7. Modify Packet Inline and Deny Connection Inline have no effect on this signature. Deny Packet Inline drops the packet and all associated fragments for this datagram.
8. IPS does not inspect the datagram if this signature is on and the number of small fragments is exceeded.
9. Modify Packet Inline and Deny Connection Inline have no effect on this signature. Deny Packet Inline drops the packet and all associated fragments for this datagram.
10. The timer starts when the packet for the datagram arrives.
11. Modify Packet Inline modifies the flags to a valid combination. Deny Connection Inline has no effect on this signature. Deny Packet Inline drops the packet and all associated fragments for this datagram.

### For More Information

For more information about the Normalizer engine, see [Normalizer Engine, page B-36](#).

## Configuring IP Fragment Reassembly Parameters

To configure IP fragment reassembly parameters for a specific signature, follow these steps:

- 
- Step 1** Log in to the CLI using an account with administrator or operator privileges.
- Step 2** Enter signature definition submode.
- ```
sensor# configure terminal
sensor(config)# service signature-definition sig1
```
- Step 3** Specify the IP fragment reassembly signature ID and subsignature ID.
- ```
sensor(config-sig)# signatures 1200 0
```
- Step 4** Specify the engine.
- ```
sensor(config-sig-sig)# engine normalizer
```
- Step 5** Enter edit default signatures submode.
- ```
sensor(config-sig-sig-nor)# edit-default-sigs-only default-signatures-only
```
- Step 6** Enable and change the default setting (if desired) of any of the IP fragment reassembly parameter for signature 1200, for example, specifying the maximum fragments.
- ```
sensor(config-sig-sig-nor-def)# specify-max-fragments yes
sensor(config-sig-sig-nor-def-yes)# max-fragments 20000
```
- Step 7** Verify the settings.
- ```
sensor(config-sig-sig-nor-def-yes)# show settings
yes

max-fragments: 20000 default: 10000

sensor(config-sig-sig-nor-def-yes)#
```
- Step 8** Exit signature definition submode.
- ```
sensor(config-sig-sig-nor-def-yes)# exit
sensor(config-sig-sig-nor-def)# exit
sensor(config-sig-sig-nor)# exit
sensor(config-sig-sig)# exit
sensor(config-sig)# exit
Apply Changes:[yes]:
```
- Step 9** Press **Enter** for apply the changes or enter **no** to discard them.
-

Configuring the Method for IP Fragment Reassembly

Use the **fragment-reassembly** command in the signature definition submode to configure the method the sensor will use to reassemble fragments. You can configure this option if your sensor is operating in promiscuous mode. If your sensor is operating in line mode, the method is NT only.

The following commands apply:

- **ip-reassemble-mode**—Identifies the method the sensor uses to reassemble the fragments based on the operating system:
 - **nt**—Specifies the Windows systems (default).

- **solaris**—Specifies the Solaris systems.
- **linux**—Specifies the GNU/Linux systems.
- **bsd**—Specifies the BSD UNIX systems.

Configuring the IP Fragment Reassembly Method

To configure the method for IP fragment reassembly, follow these steps:

-
- Step 1** Log in to the CLI using an account with administrator or operator privileges.
- Step 2** Enter fragment reassembly submenu.
- ```
sensor# configure terminal
sensor(config)# service signature-definition sig1
sensor(config-sig)# fragment-reassembly
```
- Step 3** Configure the operating system you want the sensor to use to reassemble IP fragments.
- ```
sensor(config-sig-fra)# ip-reassemble-mode linux
```
- Step 4** Verify the setting.
- ```
sensor(config-sig-fra)# show settings
fragment-reassembly

ip-reassemble-mode: linux default: nt

sensor(config-sig-fra)#
```
- Step 5** Exit signature definition submenu.
- ```
sensor(config-sig-fra)# exit
sensor(config-sig)# exit
Apply Changes:[yes]:
```
- Step 6** Press **Enter** to apply the changes or enter **no** to discard them.
-

Configuring TCP Stream Reassembly

This section describes TCP stream reassembly, lists the TCP stream reassembly signatures with the configurable parameters, describes how to configure TCP stream signatures, and how to configure the mode for TCP stream reassembly. It contains the following topics:

- [Understanding TCP Stream Reassembly, page 7-37](#)
- [TCP Stream Reassembly Signatures and Configurable Parameters, page 7-38](#)
- [Configuring TCP Stream Reassembly Signatures, page 7-43](#)
- [Configuring the Mode for TCP Stream Reassembly, page 7-44](#)

Understanding TCP Stream Reassembly

You can configure the sensor to monitor only TCP sessions that have been established by a complete three-way handshake. You can also configure how long to wait for the handshake to complete, and how long to keep monitoring a connection where no more packets have been seen. The goal is to prevent the

sensor from creating alerts where a valid TCP session has not been established. There are known attacks against sensors that try to get the sensor to generate alerts by simply replaying pieces of an attack. The TCP session reassembly feature helps to mitigate these types of attacks against the sensor.

You configure TCP stream reassembly parameters per signature. You can configure the mode for TCP stream reassembly.

TCP Stream Reassembly Signatures and Configurable Parameters

Table 7-6 lists TCP stream reassembly signatures with the parameters that you can configure for TCP stream reassembly. TCP stream reassembly signatures are part of the Normalizer engine.

Table 7-6 TCP Stream Reassembly Signatures

Signature ID and Name	Description	Parameter With Default Value and Range	Default Actions
1301 TCP Session Inactivity Timeout ¹	Fires when a TCP session has been idle for a TCP Idle Timeout.	TCP Idle Timeout 3600 (15-3600)	— ²
1302 TCP Session Embryonic Timeout ³	Fires when a TCP session has not completes the three-way handshake in TCP embryonic timeout seconds.	TCP Embryonic Timeout 15 (3-300)	— ⁴
1303 TCP Session Closing Timeout ⁵	Fires when a TCP session has not closed completely in TCP Closed Timeout seconds after the first FIN.	TCP Closed Timeout 5 (1-60)	— ⁶
1304 TCP Session Packet Queue Overflow	This signature allows for setting the internal TCP Max Queue size value for the Normalizer engine. As a result it does not function in promiscuous mode. By default this signature does not fire an alert. If a custom alert event is associated with this signature and if the queue size is exceeded, an alert fires. Note The IPS signature team discourages modifying this value.	TCP Max Queue 32 (0-128) TCP Idle Timeout 3600	— ⁷
1305 TCP Urg Flag Set ⁸	Fires when the TCP urgent flag is seen	TCP Idle Timeout 3600	Modify Packet Inline ⁹

Table 7-6 TCP Stream Reassembly Signatures (continued)

Signature ID and Name	Description	Parameter With Default Value and Range	Default Actions
1306 0 TCP Option Other	Fires when a TCP option in the range of TCP Option Number is seen. All 1306 signatures fire an alert and do not function in promiscuous mode.	TCP Option Number 6-7,9-255 (Integer Range Allow Multiple 0-255 constraints) TCP Idle Timeout 3600	Modify Packet Inline Produce Alert ¹⁰
1306 1 TCP SACK Allowed Option	Fires when a TCP selective ACK allowed option is seen. All 1306 signatures fire an alert and do not function in promiscuous mode.	TCP Idle Timeout 3600	Modify Packet Inline ¹¹
1306 2 TCP SACK Data Option	Fires when a TCP selective ACK data option is seen. All 1306 signatures fire an alert and do not function in promiscuous mode.	TCP Idle Timeout 3600	Modify Packet Inline ¹²
1306 3 TCP Timestamp Option	Fires when a TCP timestamp option is seen. All 1306 signatures fire an alert and do not function in promiscuous mode.	TCP Idle Timeout 3600	Modify Packet Inline ¹³
1306 4 TCP Window Scale Option	Fires when a TCP window scale option is seen. All 1306 signatures fire an alert and do not function in promiscuous mode.	TCP Idle Timeout 3600	Modify Packet Inline ¹⁴
1306 5 TCP MSS Option	Fires when a TCP MSS option is detected. All 1306 signatures fire an alert and do not function in promiscuous mode.	TCP Idle Timeout 3600	Modify Packet Inline
1306 6 TCP option data after EOL option	Fires when the TCP option list has data after the EOL option. All 1306 signatures fire an alert and do not function in promiscuous mode.	TCP Idle Timeout 3600	Modify Packet Inline
1307 TCP Window Variation	Fires when the right edge of the rcv window for TCP moves to the right (decreases).	TCP Idle Timeout 3600	Deny Connection Inline Produce Alert ¹⁵
1308 TTL Evasion ¹⁶	Fires when the TTL seen on one direction of a session is higher than the minimum that has been observed.	TCP Idle Timeout 3600	Modify Packet Inline ¹⁷

Table 7-6 TCP Stream Reassembly Signatures (continued)

Signature ID and Name	Description	Parameter With Default Value and Range	Default Actions
1309 TCP Reserved Flags Set	Fires when the reserved bits (including bits used for ECN) are set on the TCP header.	TCP Idle Timeout 3600	Modify Packet Inline Produce Alert ¹⁸
1311 TCP Packet Exceeds MSS	Fires when a packet exceeds the MSS that was exchanged during the three-way handshake.	TCP Idle Timeout 3600	Produce Alert ¹⁹
1312 TCP MSS Below Minimum	Fires when the MSS value in a packet containing a SYN flag is less than TCP Min MSS.	TCP Min MSS 400 (0-16000) TCP Idle Timeout 3600	Modify Packet Inline ²⁰
1313 TCP Max MSS	Fires when the MSS value in a packet containing a SYN flag exceeds TCP Max MSS	TCP Max MSS 1460 (0-16000)	Modify Packet Inline disabled ²¹
1314 TCP Data SYN	Fires when TCP payload is sent in the SYN packet.	—	Deny Packet Inline disabled ²²
1315 ACK Without TCP Stream	Fires when an ACK packet is sent that does not belong to a stream.	—	Produce Alert disabled ²³
1317 Zero Window Probe	Fires when a zero window probe packet is detected.	Modify Packet Inline removes data from the Zero Window Probe packet.	Modify Packet Inline
1330 ²⁴ 0 TCP Drop - Bad Checksum	Fires when TCP packet has bad checksum.	Modify Packet Inline corrects the checksum.	Deny Packet Inline
1330 1 TCP Drop - Bad TCP Flags	Fires when TCP packet has bad flag combination.	—	Deny Packet Inline
1330 2 TCP Drop - Urgent Pointer With No Flag	Fires when TCP packet has a URG pointer and no URG flag.	Modify Packet Inline clears the pointer.	Modify Packet Inline disabled
1330 3 TCP Drop - Bad Option List	Fires when TCP packet has a bad option list.	—	Deny Packet Inline
1330 4 TCP Drop - Bad Option Length	Fires when TCP packet has a bad option length.	—	Deny Packet Inline
1330 5 TCP Drop - MSS Option Without SYN	Fires when TCP MSS option is seen in packet without the SYN flag set.	Modify Packet Inline clears the MSS option.	Modify Packet Inline
1330 6 TCP Drop - WinScale Option Without SYN	Fires when TCP window scale option is seen in packet without the SYN flag set.	Modify Packet Inline clears the window scale option.	Modify Packet Inline

Table 7-6 TCP Stream Reassembly Signatures (continued)

Signature ID and Name	Description	Parameter With Default Value and Range	Default Actions
1330 7 TCP Drop - Bad WinScale Option Value	Fires when a TCP packet has a bad window scale value.	Modify Packet Inline sets the value to the closest constraint value.	Modify Packet Inline
1330 8 TCP Drop - SACK Allow Without SYN	Fires when the TCP SACK allowed option is seen in a packet without the SYN flags set.	Modify Packet Inline clears the SACK allowed option.	Modify Packet Inline
1330 9 TCP Drop - Data in SYN ACK	Fires when TCP packet with SYN and ACK flags set also contains data.	—	Deny Packet Inline
1330 10 TCP Drop - Data Past FIN	Fires when TCP data is sequenced after FIN.	—	Deny Packet Inline
1330 11 TCP Drop - Timestamp not Allowed	Fires when TCP packet has timestamp option when timestamp option is not allowed.	—	Deny Packet Inline
1330 12 TCP Drop - Segment Out of Order	Fires when TCP segment is out of order and cannot be queued.	—	Deny Packet Inline
1330 13 TCP Drop - Invalid TCP Packet	Fires when TCP packet has invalid header.	—	Deny Packet Inline
1330 14 TCP Drop - RST or SYN in window	Fires when TCP packet with RST or SYN flag was sent in the sequence window but was not the next sequence.	—	Deny Packet Inline
1330 15 TCP Drop - Segment Already ACKed	Fires when TCP packet sequence is already ACKed by peer (excluding keepalives).	—	Deny Packet Inline
1330 16 TCP Drop - PAWS Failed	Fires when TCP packet fails PAWS check.	—	Deny Packet Inline
1330 17 TCP Drop - Segment out of State Order	Fires when TCP packet is not proper for the TCP session state.	—	Deny Packet Inline
1330 18 TCP Drop - Segment out of Window	Fires when TCP packet sequence number is outside of allowed window.	—	Deny Packet Inline
3050 Half Open SYN Attack		syn-flood-max-embryonic 5000	
3250 TCP Hijack		max-old-ack 200	
3251 TCP Hijack Simplex Mode		max-old-ack 100	

- The timer is reset to 0 after each packet on the TCP session. by default, this signature does not produce an alert. You can choose to produce alerts for expiring TCP connections if desired. A statistic of total number of expired flows is updated any time a flow expires.

2. Modify Packet Inline, Deny Connection Inline, and Deny Packet Inline have no effect on this signature.
3. The timer starts with the first SYN packet and is not reset. State for the session is reset and any subsequent packets for this flow appear to be out of order (unless it is a SYN).
4. Modify Packet Inline, Deny Connection Inline, and Deny Packet Inline have no effect on this signature.
5. The timer starts with the first FIN packet and is not reset. State for the session is reset and any subsequent packets for this flow appear to be out of order (unless it is a SYN).
6. Modify Packet Inline, Deny Connection Inline, and Deny Packet Inline have no effect on this signature.
7. Modify Packet Inline and Deny Packet Inline have no effect on this signature. Deny Connection Inline drops the current packet and the TCP session.
8. Phrak 57 describes a way to evade security policy using URG pointers. You can normalize the packet when it is in inline mode with this signature.
9. Modify Packet Inline strips the URG flag and zeros the URG pointer from the packet. Deny Connection Inline drops the current packet and the TCP session. Deny Packet Inline drops the packet.
10. Modify Packet Inline strips the selected option(s) from the packet. Deny Connection Inline drops the current packet and the TCP session. Deny Packet Inline drops the packet.
11. Modify Packet Inline strips the selected ACK allowed option from the packet. Deny Connection Inline drops the current packet and the TCP session. Deny Packet Inline drops the packet.
12. Modify Packet Inline strips the selected ACK allowed option from the packet. Deny Connection Inline drops the current packet and the TCP session. Deny Packet Inline drops the packet.
13. Modify Packet Inline strips the timestamp option from the packet. Deny Connection Inline drops the current packet and the TCP session. Deny Packet Inline drops the packet.
14. Modify Packet Inline strips the window scale option from the packet. Deny Connection Inline drops the current packet and the TCP session. Deny Packet Inline drops the packet.
15. Modify Packet Inline has no effect on this signature. Deny Connection Inline drops the current packet and the TCP connection. Deny Packet Inline drops the packet.
16. This signature is used to cause TTLs to monotonically decrease for each direction on a session. For example, if TTL 45 is the lowest TTL seen from A to B, then all future packets from A to B will have a maximum of 45 if Modify Packet Inline is set. Each new low TTL becomes the new maximum for packets on that session.
17. Modify Packet Inline ensures that the IP TTL monotonically decreases. Deny Connection Inline drops the current packet and the TCP session. Deny Packet Inline drops the packet.
18. Modify Packet Inline clears all reserved TCP flags. Deny Connection Inline drops the current packet and the TCP session. Deny Packet Inline drops the packet.
19. Modify Packet Inline has no effect on this signature. Deny Connection Inline drops the current packet and the TCP connection. Deny Packet Inline drops the packet.
20. 2.4.21-15.EL.cisco.1 Modify Packet Inline raises the MSS value to TCP Min MSS. Deny Connection Inline drops the current packet and the TCP session. Deny Packet Inline drops the packet 2.4.21-15.EL.cisco.1.
21. Modify Packet Inline lowers the MSS value to TCP Max MSS. Deny Connection Inline drops the current packet and the TCP session. Deny Packet Inline drops the packet 2.4.21-15.EL.cisco.1.
22. Modify Packet Inline has no effect on this signature. Deny Connection Inline drops the current packet and the TCP session. Deny Packet Inline drops the packet.
23. Modify Packet Inline, Deny Connection Inline, and Deny Packet Inline have no effect on this signature. By default, the 1330 signatures drop packets for which this signature sends alerts.
24. These subsignatures represent the reasons why the Normalizer might drop a TCP packet. By default these subsignatures drop packets. These subsignatures let you permit packets that fail the checks in the Normalizer through the IPS. The drop reasons have an entry in the TCP statistics. By default these subsignatures do not produce an alert.

For More Information

For more information about the Normalizer engine, see [Normalizer Engine, page B-36](#).

Configuring TCP Stream Reassembly Signatures

To configure TCP stream reassembly for a specific signature, follow these steps:

Step 1 Log in to the CLI using an account with administrator or operator privileges.

Step 2 Enter signature definition submenu.

```
sensor# configure terminal
sensor(config)# service signature-definition sig1
```

Step 3 Specify the TCP stream reassembly signature ID and subsignature ID.

```
sensor(config-sig)# signatures 1313 0
```

Step 4 Specify the engine.

```
sensor(config-sig-sig)# engine normalizer
```

Step 5 Enter edit default signatures submenu.

```
sensor(config-sig-sig-nor)# edit-default-sigs-only default-signatures-only
```

Step 6 Enable and change the default setting (if desired) of the maximum MSS parameter for signature 1313.

```
sensor(config-sig-sig-nor-def)# specify-tcp-max-mss yes
sensor(config-sig-sig-nor-def-yes)# tcp-max-mss 1380
```



Note Changing this parameter from the default of 1460 to 1380 helps prevent fragmentation of traffic going through a VPN tunnel.

Step 7 Verify the settings.

```
sensor(config-sig-sig-nor-def-yes)# show settings
yes
-----
      tcp-max-mss: 1380 default: 1460
-----
sensor(config-sig-sig-nor-def-yes)#
```

Step 8 Exit signature definition submenu.

```
sensor(config-sig-sig-nor-def-yes)# exit
sensor(config-sig-sig-nor-def)# exit
sensor(config-sig-sig-nor)# exit
sensor(config-sig-sig)# exit
sensor(config-sig)# exit
Apply Changes:[yes]:
```

Step 9 Press **Enter** for apply the changes or enter **no** to discard them.

Configuring the Mode for TCP Stream Reassembly

Use the **stream-reassembly** command in the signature definition submode to configure the mode that the sensor will use to reassemble TCP sessions.



Note

The parameters **tcp-3-way-handshake-required** and **tcp-reassembly-mode** only impact sensors inspecting traffic in promiscuous mode, not inline mode. To configure asymmetric options for sensors inspecting inline traffic, use the **inline-TCP-evasion-protection-mode** parameter.

The following commands apply:

- **tcp-3-way-handshake-required [true | false]**—Specifies that the sensor should only track sessions for which the 3-way handshake is completed. The default is true.
- **tcp-reassembly-mode**—Specifies the mode the sensor should use to reassemble TCP sessions:
 - **strict**—Only allows the next expected in the sequence (default).
 - **loose**—Allows gaps in the sequence.
 - **asym**—Allows asymmetric traffic to be reassembled.



Caution

The asymmetric option disables TCP window evasion checking.

Configuring the TCP Stream Reassembly Parameters

To configure the TCP stream reassembly parameters, follow these steps:

-
- Step 1** Log in to the CLI using an account with administrator or operator privileges.
- Step 2** Enter TCP stream reassembly submode.
- ```
sensor# configure terminal
sensor(config)# service signature-definition sig1
sensor(config-sig)# stream-reassembly
```
- Step 3** Specify that the sensor should only track session for which the 3-way handshake is completed.
- ```
sensor(config-sig-str)# tcp-3-way-handshake-required true
```
- Step 4** Specify the mode the sensor should use to reassemble TCP sessions.
- ```
sensor(config-sig-str)# tcp-reassembly-mode strict
```
- Step 5** Verify the settings.
- ```
sensor(config-sig-str)# show settings
stream-reassembly
-----
tcp-3-way-handshake-required: true default: true
tcp-reassembly-mode: strict default: strict
-----
sensor(config-sig-str)#
```
- Step 6** Exit signature definition submode.
- ```
sensor(config-sig-str)# exit
sensor(config-sig)# exit
Apply Changes?[yes]:
```

**Step 7** Press **Enter** to apply the changes or enter **no** to discard them.

---

#### For More Information

For information on asymmetric inspection options for sensors configured in inline mode, see [Inline TCP Session Tracking Mode, page 6-3](#) and [Adding, Editing, and Deleting Virtual Sensors, page 6-5](#).

## Configuring IP Logging

You can configure a sensor to generate an IP session log when the sensor detects an attack. When IP logging is configured as a response action for a signature and the signature is triggered, all packets to and from the source address of the alert are logged for a specified period of time.



#### Note

IP logging allows a maximum limit of 20 concurrent IP log files. Once the limit of 20 is reached, you receive the following message in main.log: Cid/W errWarnIpLogProcessor::addIpLog: Ran out of file descriptors.

---

Use the **ip-log** command in service signature definition mode to configure IP logging.

The following commands apply:

- **ip-log-bytes**—Identifies the maximum number of bytes you want logged. The valid value is 0 to 2147483647. The default is 0.
- **ip-log-packets**—Identifies the number of packets you want logged. The valid value is 0 to 65535. The default is 0.
- **ip-log-time**—Identifies the duration you want the sensor to log. The valid value is 30 to 300 seconds. The default is 30 seconds.



#### Note

When the sensor meets any one of the IP logging conditions, it stops IP logging.

---

#### Configuring IP Logging Parameters

To configure the IP logging parameters, follow these steps:

---

**Step 1** Log in to the CLI using an account with administrator or operator privileges.

**Step 2** Enter IP log submode.

```
sensor# configure terminal
sensor(config)# service signature-definition sig1
sensor(config-sig)# ip-log
```

**Step 3** Specify the IP logging parameters:

- a. Specify the maximum number of bytes you want logged.

```
sensor(config-sig-ip)# ip-log-bytes 200000
```

- b. Specify the number of packets you want logged.

```
sensor(config-sig-ip)# ip-log-packets 150
```

- c. Specify the length of time you want the sensor to log.

```
sensor(config-sig-ip)# ip-log-time 60
```

- Step 4** Verify the settings.

```
sensor(config-sig-ip)# show settings
ip-log

ip-log-packets: 150 default: 0
ip-log-time: 60 default: 30
ip-log-bytes: 200000 default: 0

sensor(config-sig-ip)#
```

- Step 5** Exit signature definition mode.

```
sensor(config-sig-ip)# exit
sensor(config-sig)# exit
Apply Changes?[yes]:
```

- Step 6** Press **Enter** to apply the changes or enter **no** to discard them.
- 

## Creating Custom Signatures

This section describes how to create custom signatures and contains the following topics:

- [Sequence for Creating a Custom Signature, page 7-46](#)
- [Example String TCP Engine Signature, page 7-47](#)
- [Example Service HTTP Engine Signature, page 7-50](#)
- [Example Meta Engine Signature, page 7-52](#)
- [Example IPv6 Engine Signature, page 7-57](#)
- [Example String XL TCP Engine Match Offset Signature, page 7-58](#)
- [Example String XL TCP Engine Minimum Match Length Signature, page 7-61](#)

## Sequence for Creating a Custom Signature

Use the following sequence when you create a custom signature:

---

- Step 1** Select a signature engine.
- Step 2** Assign the signature identifiers:
- Signature ID
  - SubSignature ID
  - Signature name
  - Alert notes (optional)
  - User comments (optional)

- Step 3** Assign the engine-specific parameters. The parameters differ for each signature engine, although there is a group of master parameters that applies to each engine.
- Step 4** Assign the alert response:
- Signature fidelity rating
  - Severity of the alert
- Step 5** Assign the alert behavior.
- Step 6** Apply the changes.
- 

## Example String TCP Engine Signature

The String engine is a generic-based pattern-matching inspection engine for ICMP, TCP, and UDP protocols. The String engine uses a regular expression engine that can combine multiple patterns into a single pattern-matching table allowing for a single search through the data. There are three String engines: String ICMP, String TCP, and String UDP.



### Caution

A custom signature can affect the performance of your sensor. Test the custom signature against a baseline sensor performance for your network to determine the overall impact of the signature.

---



### Note

This procedure also applies to String UDP and ICMP signatures.

---

The following commands apply:

- **default**—Sets the value back to the system default setting.
- **direction**—Specifies the direction of the traffic:
  - **from-service**—Traffic from service port destined to client port.
  - **to-service**—Traffic from client port destined to service port.
- **event-action**—Specifies the action(s) to perform when alert is triggered:
  - **deny-attacker-inline** (inline only)—Does not transmit this packet and future packets from the attacker address for a specified period of time.
  - **deny-attacker-service-pair-inline** (inline only)—Does not transmit this packet and future packets on the attacker address victim port pair for a specified period of time.
  - **deny-attacker-victim-pair-inline** (inline only)—Does not transmit this packet and future packets on the attacker/victim address pair for a specified period of time.
  - **deny-connection-inline** (inline only)—Does not transmit this packet and future packets on the TCP flow.
  - **deny-packet-inline** (inline only)—Does not transmit this packet.
  - **log-attacker-packets**—Starts IP logging of packets containing the attacker address.
  - **log-pair-packets**—Starts IP logging of packets containing the attacker-victim address pair.
  - **log-victim-packets**—Starts IP logging of packets containing the victim address.
  - **produce-alert** —Writes the event to the Event Store as an alert.

- **produce-verbose-alert**—Includes an encoded dump (possibly truncated) of the offending packet in the alert.
- **request-block-connection**—Sends a request to the ARC to block this connection.
- **request-block-host**—Sends a request to the ARC to block this attacker host.
- **request-rate-limit**—Sends a rate limit request to the ARC to perform rate limiting.
- **request-snmp-trap**—Sends a request to the Notification Application component of the sensor to perform SNMP notification.
- **reset-tcp-connection**—Sends TCP resets to hijack and terminate the TCP flow.
- **modify-packet-inline**—Modifies packet data to remove ambiguity about what the end point might do with the packet.
- **no**—Removes an entry or selection setting.
- **regex-string** —Specifies a regular expression to search for in a single TCP packet.
- **service-ports**—Specifies the ports or port ranges where the target service may reside. The valid range is 0 to 65535. It is a separated list of integer ranges a-b[,c-d] within 0 to 65535. The second number in the range must be greater than or equal to the first number.
- **specify-exact-match-offset {yes | no}**—(Optional) Enables exact match offset:
  - **exact-match-offset**—Specifies the exact stream offset the regular expression string must report for a match to be valid. The value is 0 to 65535.
- **specify-min-match-length {yes | no}**—(Optional) Enables minimum match length:
  - **min-match-length**—Specifies the minimum number of bytes the regular expression string must match. The value is 0 to 65535.
- **strip-telnet-options {true | false}**—Strips the Telnet option characters from the data before the pattern is searched.
- **swap-attacker-victim {true | false}**—Swaps the attacker and victim addresses and ports (source and destination) in the alert message and in any actions taken. The default is false.

### Creating a String TCP Engine Signature

To create a signature based on the String TCP engine, follow these steps:

- 
- Step 1** Log in to the CLI using an account with administrator or operator privileges.
- Step 2** Enter signature definition submode.
- ```
sensor# configure terminal
sensor(config)# service signature-definition sig1
```
- Step 3** Specify a signature ID and subsignature ID for the signature. Custom signatures are in the range of 60000 to 65000.
- ```
sensor(config-sig)# signatures 60025 0
```
- Step 4** Enter signature description submode.
- ```
sensor(config-sig-sig)# sig-description
```
- Step 5** Specify a name for the new signature. You can also specify a additional comments about the sig using the **sig-comment** command or additional information about the signature using the **sig-string-info** command.
- ```
sensor(config-sig-sig-sig)# sig-name This is my new name
```

**Step 6** Exit signature description submode.

```
sensor(config-sig-sig-sig)# exit
```

**Step 7** Specify the string TCP engine.

```
sensor(config-sig-sig)# engine string-tcp
```

**Step 8** Specify the service ports.

```
sensor(config-sig-sig-str)# service-ports 23
```

**Step 9** Specify the direction.

```
sensor(config-sig-sig-str)# direction to-service
```

**Step 10** Specify the regex string to search for in the TCP packet. You can change the event actions if needed according to your security policy using the **event-action** command. The default event action is **produce-alert**.

```
sensor(config-sig-sig-str)# regex-string This-is-my-new-Sig-regex
```

**Step 11** You can modify the following optional parameters for this custom String TCP signature:

- **specify-exact-match-offset**
- **specify-min-match-length**
- **strip-telnet-options**
- **swap-attacker-victim**.

**Step 12** Verify the settings.

```
sensor(config-sig-sig-str)# show settings
string-tcp
```

```

event-action: produce-alert <defaulted>
strip-telnet-options: false <defaulted>
specify-min-match-length
```

```
no
```

```

regex-string: This-is-my-new-Sig-regex
service-ports: 23
direction: to-service default: to-service
specify-exact-match-offset
```

```
no
```

```

specify-max-match-offset
```

```
no
```

```

specify-min-match-offset
```

```
no
```

```

swap-attacker-victim: false <defaulted>

sensor(config-sig-sig-str)#

```

**Step 13** Exit signature definition submenu.

```

sensor(config-sig-sig-str)# exit
sensor(config-sig-sig)# exit
sensor(config-sig)# exit
Apply Changes:[yes]:

```

**Step 14** Press **Enter** to apply the changes or enter **no** to discard them.

## Example Service HTTP Engine Signature

The Service HTTP engine is a service-specific string-based pattern-matching inspection engine. The HTTP protocol is one of the most commonly used in networks of today. In addition, it requires the most amount of preprocessing time and has the most number of signatures requiring inspection making it critical to the overall performance of the system.

The Service HTTP engine uses a Regex library that can combine multiple patterns into a single pattern-matching table allowing a single search through the data. This engine searches traffic directed only to web services, or HTTP requests. You cannot inspect return traffic with this engine. You can specify separate web ports of interest in each signature in this engine.

HTTP deobfuscation is the process of decoding an HTTP message by normalizing encoded characters to ASCII equivalent characters. It is also known as ASCII normalization.

Before an HTTP packet can be inspected, the data must be deobfuscated or normalized to the same representation that the target system sees when it processes the data. It is ideal to have a customized decoding technique for each host target type, which involves knowing what operating system and web server version is running on the target. The Service HTTP engine has default deobfuscation behavior for the Microsoft IIS web server.

The following commands apply:

- **de-obfuscate {true | false}**—Applies anti-evasive deobfuscation before searching.
- **default**—Sets the value back to the system default setting.
- **event-action** —Specifies the action(s) to perform when alert is triggered:
  - **deny-attacker-inline** (inline only)—Does not transmit this packet and future packets from the attacker address for a specified period of time.
  - **deny-attacker-service-pair-inline** (inline only)—Does not transmit this packet and future packets on the attacker address victim port pair for a specified period of time.
  - **deny-attacker-victim-pair-inline** (inline only)—Does not transmit this packet and future packets on the attacker/victim address pair for a specified period of time.
  - **deny-connection-inline** (inline only)—Does not transmit this packet and future packets on the TCP flow.
  - **deny-packet-inline** (inline only)—Does not transmit this packet.
  - **log-attacker-packets**—Starts IP logging of packets containing the attacker address.
  - **log-pair-packets**—Starts IP logging of packets containing the attacker-victim address pair.
  - **log-victim-packets**—Starts IP logging of packets containing the victim address.



- **produce-alert** —Writes the event to the Event Store as an alert.
- **produce-verbose-alert**—Includes an encoded dump (possibly truncated) of the offending packet in the alert.
- **request-block-connection**—Sends a request to the ARC to block this connection.
- **request-block-host**—Sends a request to the ARC to block this attacker host.
- **request-rate-limit**—Sends a rate limit request to the ARC to perform rate limiting.
- **request-snmp-trap**—Sends a request to the Notification Application component of the sensor to perform SNMP notification.
- **reset-tcp-connection**—Sends TCP resets to hijack and terminate the TCP flow.
- **modify-packet-inline**— Modifies packet data to remove ambiguity about what the end point might do with the packet.
- **max-field-sizes** —Grouping for maximum field sizes:
  - **specify-max-arg-field-length {yes | no}**—Enables max-arg-field-length (optional).
  - **specify-max-header-field-length {yes | no}**—Enables max-header-field-length (optional).
  - **specify-max-request-length {yes | no}**—Enables max-request-length (optional).
  - **specify-max-uri-field-length {yes | no}**—Enables max-uri-field-length (optional).
- **no**—Removes an entry or selection setting.
- **regex**—Regular expression grouping:
  - **specify-arg-name-regex**—Enables arg-name-regex (optional).
  - **specify-header-regex** —Enables header-regex (optional).
  - **specify-request-regex**—Enables request-regex (optional).
  - **specify-uri-regex**—Enables uri-regex (optional).
- **service-ports** —A comma-separated list of ports or port ranges where the target service may reside.
- **swap-attacker-victim {true | false}**—Whether address (and ports) source and destination are swapped in the alarm message. The default is false for no swapping.

### Creating a Service HTTP Engine Signature

To create a custom signature based on the Service HTTP engine, follow these steps:

- 
- Step 1** Log in to the CLI using an account with administrator or operator privileges.
- Step 2** Enter signature definition submode.
- ```
sensor# configure terminal
sensor(config)# service signature-definition sig1
```
- Step 3** Specify a signature ID and a subsignature ID for the signature. Custom signatures are in the range of 60000 to 65000.
- ```
sensor(config-sig)# signatures 63000 0
```
- Step 4** Enter signature description mode.
- ```
sensor(config-sig-sig)# sig-description
```
- Step 5** Specify a signature name.
- ```
sensor(config-sig-sig-sig)# sig-name myWebSig
```

**Step 6** Specify the alert traits. The valid range is from 0 to 65535.

```
sensor(config-sig-sig-sig)# alert-traits 2
```

**Step 7** Exit signature description submenu.

```
sensor(config-sig-sig-sig)# exit
```

**Step 8** Specify the alert frequency.

```
sensor(config-sig-sig)# alert-frequency
sensor(config-sig-sig-ale)# summary-mode fire-all
sensor(config-sig-sig-ale-fir)# summary-key Axxx
sensor(config-sig-sig-ale-fir)# specify-summary-threshold yes
sensor(config-sig-sig-ale-fir-yes)# summary-threshold 200
```

**Step 9** Exit alert frequency submenu.

```
sensor(config-sig-sig-ale-fir-yes)# exit
sensor(config-sig-sig-ale-fir)# exit
sensor(config-sig-sig-ale)# exit
```

**Step 10** Configure the signature to apply anti-evasive deobfuscation before searching:

```
sensor(config-sig-sig)# engine service-http
sensor(config-sig-sig-ser)# de-obfuscate true
```

**Step 11** Configure the Regex parameters.

```
sensor(config-sig-sig)# engine service-http
sensor(config-sig-sig-ser)# regex
sensor(config-sig-sig-ser-reg)# specify-uri-regex yes
sensor(config-sig-sig-ser-reg-yes)# uri-regex [Mm][Yy][Ff][Oo][Oo]
```

**Step 12** Exit Regex submenu.

```
sensor(config-sig-sig-ser-reg-yes)# exit
sensor(config-sig-sig-ser-reg-)# exit
```

**Step 13** Configure the service ports using the signature variable WEBPORTS.

```
sensor(config-sig-sig-ser)# service-ports $WEBPORTS
```

**Step 14** Exit signature definition submenu.

```
sensor(config-sig-sig-ser)# exit
sensor(config-sig-sig)# exit
sensor(config-sig)# exit
Apply Changes?[yes]:
```

**Step 15** Press **Enter** to apply the changes or enter **no** to discard them.

## Example Meta Engine Signature



### Caution

A large number of Meta engine signatures could adversely affect overall sensor performance.

The Meta engine defines events that occur in a related manner within a sliding time interval. This engine processes events rather than packets. As signature events are generated, the Meta engine inspects them to determine if they match any or several Meta definitions. The Meta engine generates a signature event after all requirements for the event are met.

All signature events are handed off to the Meta engine by the Signature Event Action Processor. The Signature Event Action Processor hands off the event after processing the minimum hits option. Summarization and event action are processed after the Meta engine has processed the component events.

### Meta Signature Engine Enhancement

The purpose of the Meta engine is to detect a specified payload from an attacker and a corresponding payload from the victim. It is also used to inspect streams at different offsets. The Meta engine supports the AND and OR logical operators. ANDNOT capability has been added to the Meta engine. This clause is a negative clause used to complement the existing positive clause-based signatures. The previous signature format had the following form:

```
IF (A and B and C) then Alarm; alternatively, IF (A or B or C) then Alarm is also supported; where A, B, and C are meta component signatures.
```

The addition of the negative clause allows for the following logic:

```
IF (A and/or B) AND NOT (C and/or D) then Alarm.
```

The (C and/or D) is the negative clause and is satisfied if (C and D) [alternatively (C or D)] do not occur before the Meta Reset Interval time expires.

A component of the positive clause must occur before the negative clause(s) to establish the Meta tracking state. The Meta engine cannot track the lack of past behavior. The state of the negative clause is evaluated when the Meta Reset Interval time expires.



#### Caution

---

A custom signature can affect the performance of your sensor. Test the custom signature against a baseline sensor performance for your network to determine the overall impact of the signature.

---

The Meta engine is different from other engines in that it takes alerts as input where most engines take packets as input.

The following commands apply:

- **component-list** *name1*—Specifies the list of Meta components:
  - **edit**—Edits an existing entry in the list.
  - **insert**—Inserts a new entry into the list.
  - **move**—Moves an entry in the list.
  - **begin**—Places the entry at the beginning of the active list.
  - **end**—Places the entry at the end of the active list.
  - **inactive**—Places the entry into the inactive list.
  - **before**—Places the entry before the specified entry.
  - **after**—Places the entry after the specified entry.
  - **component-count**—Specifies the number of times component must fire before this component is satisfied.
  - **component-sig-id**—Specifies the signature ID of the signature to match this component on.

- **component-subsig-id**—Specifies the subsignature ID of the signature to match this component on.
- **is-not-component {true | false}**—Specifies that the component is a NOT component.
- **component-list-in-order {true | false}**—Specifies whether to have the component list fire in order. For example, if signature 1001 in the m2 component fires before signature 1000 in the m1 component, the Meta signature will not fire.
- **all-components-required {true | false}**—Specifies to use all components. This option works with the **all-not-components-required** option, if you have NOT components configured as required, the Meta signature will not fire.
- **all-not-components-required {true | false}**—Specifies to use all of the NOT components.
- **swap-attacker-victim {true | false}**—Swaps the attacker and victim addresses and ports (source and destination) in the alert message and in any actions taken.
- **meta-reset-interval**—Specifies the time in seconds to reset the Meta signature. The valid range is 0 to 3600 seconds. The default is 60 seconds.
- **meta-key**—Specifies the storage type for the Meta signature:
  - **AaBb**—Attacker and victim addresses and ports.
  - **AxBx**—Attacker and victim addresses.
  - **Axxx**—Attacker address.
  - **xxBx**—Victim address.
- **unique-victim-ports**—Specifies the number of unique victims ports required per Meta signature. The valid range is 1 to 256.
- **event-action**—Specifies the action(s) to perform when an alert is triggered:
  - **deny-attacker-inline** (inline only)—Does not transmit this packet and future packets from the attacker address for a specified period of time.
  - **deny-attacker-service-pair-inline** (inline only)—Does not transmit this packet and future packets on the attacker address victim port pair for a specified period of time.
  - **deny-attacker-victim-pair-inline** (inline only)—Does not transmit this packet and future packets on the attacker/victim address pair for a specified period of time.
  - **deny-connection-inline** (inline only)—Does not transmit this packet and future packets on the TCP flow.
  - **deny-packet-inline** (inline only)—Does not transmit this packet.
  - **log-attacker-packets**—Starts IP logging of packets containing the attacker address.
  - **log-pair-packets**—Starts IP logging of packets containing the attacker-victim address pair.
  - **log-victim-packets**—Starts IP logging of packets containing the victim address.
  - **produce-alert** —Writes the event to the Event Store as an alert.
  - **produce-verbose-alert**—Includes an encoded dump (possibly truncated) of the offending packet in the alert.
  - **request-block-connection**—Sends a request to the ARC to block this connection.
  - **request-block-host**—Sends a request to the ARC to block this attacker host.
  - **request-rate-limit**—Sends a rate limit request to the ARC to perform rate limiting.
  - **request-snmp-trap**—Sends a request to the Notification Application component of the sensor to perform SNMP notification.

- **reset-tcp-connection**—Sends TCP resets to hijack and terminate the TCP flow.
- **modify-packet-inline**— Modifies packet data to remove ambiguity about what the end point might do with the packet.

**Note**

Signature 64000 subsignature 0 will fire when it sees the alerts from signature 1000 subsignature 0 and signature 1001 subsignature 0 on the same source address. The source address selection is a result of the meta key default value of Axxx. You can change the behavior by changing the meta key setting to xxBx (destination address) for example.

**Creating a Meta Engine Signature**

To create a signature based on the Meta engine, follow these steps:

**Step 1** Log in to the CLI using an account with administrator or operator privileges.

**Step 2** Enter signature definition submode.

```
sensor# configure terminal
sensor(config)# service signature-definition sig1
```

**Step 3** Specify a signature ID and a subsignature ID for the signature. Custom signatures are in the range of 60000 to 65000.

```
sensor(config-sig)# signatures 64000 0
```

**Step 4** Specify the signature engine.

```
sensor(config-sig-sig)# engine meta
```

**Step 5** Insert a signature (named m1) at the beginning of the list.

```
sensor(config-sig-sig-met)# component-list insert m1 begin
```

**Step 6** Specify the signature ID of the signature on which to match this component.

```
sensor(config-sig-sig-met-com)# component-sig-id 1000
```

**Step 7** Exit component list submode.

```
sensor(config-sig-sig-met-com)# exit
```

**Step 8** Insert another signature (named m2) at the end of the list.

```
sensor(config-sig-sig-met)# component-list insert m2 end
```

**Step 9** Specify the signature ID of the signature on which to match this component.

```
sensor(config-sig-sig-met-com)# component-sig-id 1001
```

**Step 10** Configure the component list not to fire in order.

```
sensor(config-sig-sig-met-com)# component-list-in-order false
```

**Step 11** Specify to use all components you have created.

```
sensor(config-sig-sig-met-com)# all-components-required true
```

**Step 12** Specify not to use all of the NOT components.

```
sensor(config-sig-sig-met-com)# all-not-components-required false
```

**Step 13** Verify the settings.

```

sensor(config-sig-sig-met-com)# exit
sensor-128(config-sig-sig-met)# show settings
meta

event-action: produce-alert <defaulted>
swap-attacker-victim: false <defaulted>
meta-reset-interval: 60 <defaulted>
component-list (ordered min: 1, max: 32, current: 2 - 2 active, 0 inactive)

ACTIVE list-contents

NAME: m1

component-sig-id: 1000
component-subsig-id: 0 default: 0
component-count: 1 default: 1
is-not-component: false <defaulted>

NAME: m2

component-sig-id: 1001
component-subsig-id: 0 <defaulted>
component-count: 1 <defaulted>
is-not-component: true default: false

meta-key

Axxx

unique-victims: 1 <defaulted>

component-list-in-order: false default: false
all-components-required: true default: true
all-nots-required: false default: false

sensor(config-sig-sig-met)#

```

**Step 14** Exit signature definition submenu.

```

sensor(config-sig-sig-met)# exit
sensor(config-sig-sig)# exit
sensor(config-sig)# exit
Apply Changes?[yes]:

```

**Step 15** Press **Enter** to apply the changes or enter **no** to discard them.**For More Information**

- For more information on Signature Event Action Processor, see [Signature Event Action Processor, page 8-3](#).
- For more information on the Meta engine, see [Meta Engine, page B-33](#).

## Example IPv6 Engine Signature



### Caution

A custom signature can affect the performance of your sensor. Test the custom signature against a baseline sensor performance for your network to determine the overall impact of the signature.

The following example Atomic IP Advanced custom signature prohibits Protocol ID 88 over IPv6. To create a signature based on the Atomic IP Advanced signature engine, follow these steps:

**Step 1** Log in to the CLI using an account with administrator or operator privileges.

**Step 2** Enter signature definition submode.

```
sensor# configure terminal
sensor(config)# service signature-definition sig0
```

**Step 3** Specify a signature ID and a subsignature ID for the signature. Custom signatures are in the range of 60000 to 65000.

```
sensor(config-sig)# signatures 60000 0
```

**Step 4** Specify the signature engine.

```
sensor(config-sig-sig)# engine atomic-ip-advanced
```

**Step 5** Specify the IP version.

```
sensor(config-sig-sig-ato)# specify-ip-version yes
```

**Step 6** Specify IPv6.

```
sensor(config-sig-sig-ato-yes)# version ipv6
```

**Step 7** Specify the L4 protocol.

```
sensor(config-sig-sig-ato-yes-ipv)# exit
sensor(config-sig-sig-ato-yes)# exit
sensor(config-sig-sig-ato)# specify-l4-protocol yes
```

**Step 8** Specify protocol ID 88.

```
sensor(config-sig-sig-ato-yes)# l4-protocol other-protocol
sensor(config-sig-sig-ato-yes-oth)# other-ip-protocol-id 88
```

**Step 9** Verify the settings.

```
sensor(config-sig-sig-ato-yes-oth)# show settings
 other-protocol

 other-ip-protocol-id: 88

sensor(config-sig-sig-ato-yes-oth)#
```

**Step 10** Exit signature definition submode.

```
sensor(config-sig-sig-ato-yes-oth)# exit
sensor(config-sig-sig-ato-yes)# exit
sensor(config-sig-sig-ato)# exit
sensor(config-sig-sig)# exit
sensor(config-sig)# exit
Apply Changes?[yes]:
```

**Step 11** Press **Enter** to apply the changes or enter **no** to discard them.

---

#### For More Information

- For more information about the Atomic IP Advanced engine and a list of the parameters, see [Atomic IP Advanced Engine, page B-15](#).
- For more information on the Atomic engines, see [Atomic Engine, page B-14](#).

## Example String XL TCP Engine Match Offset Signature



#### Caution

A custom signature can affect the performance of your sensor. Test the custom signature against a baseline sensor performance for your network to determine the overall impact of the signature.

---



#### Note

This procedure also applies to String XLUDP and String XL ICMP signatures, with the exception of the parameter **service-ports**, which does not apply to String XL ICMP signatures.

---

The following example demonstrates how to create a custom String XL TCP signature that searches for exact, maximum, or minimum offsets. You can modify the following optional match offset parameters for this custom String XL TCP signature:

- **specify-exact-match-offset {yes |no}**—Enables exact match offset:
  - **exact-match-offset**—Specifies the exact stream offset in bytes the regular expression string must report for a match to be valid. The value is 0 to 65535.
- **specify-max-match-offset {yes |no}**—Enables maximum match length:
  - **max-match-offset**—Specifies the maximum stream offset in bytes the regular expression string must report for a match to be valid. The value is 0 to 65535.
- **specify-min-match-offset {yes |no}**—Enables minimum match offset:
  - **min-match-offset**—Specifies the minimum stream offset in bytes the regular expression string must report for a match to be valid. The value is 0 to 65535.

#### Creating a String XL TCP Engine Signature

To create a custom signature based on the String XL TCP engine that searches for matches, follow these steps:

---

**Step 1** Log in to the CLI using an account with administrator or operator privileges.

**Step 2** Enter signature definition submode.

```
sensor# configure terminal
sensor(config)# service signature-definition sig1
```

**Step 3** Specify a signature ID and subsignature ID for the signature. Custom signatures are in the range of 60000 to 65000.

```
sensor(config-sig)# signatures 60003 0
```



- Step 4** Enter signature description submenu.
- ```
sensor(config-sig-sig)# sig-description
```
- Step 5** Specify a name for the new signature. You can also specify a additional comments about the sig using the **sig-comment** command or additional information about the signature using the **sig-string-info** command.
- ```
sensor(config-sig-sig-sig)# sig-name This is my new name
```
- Step 6** Exit signature description submenu.
- ```
sensor(config-sig-sig-sig)# exit
```
- Step 7** Specify the String XL TCP engine.
- ```
sensor(config-sig-sig)# engine string-xl-tcp
```
- Step 8** Specify the service ports.
- ```
sensor(config-sig-sig-str)# service-ports 80
```
- Step 9** Specify the direction.
- ```
sensor(config-sig-sig-str)# direction to-service
```
- Step 10** Change the event actions if needed according to your security policy by using the **event-action** command. The default event action is **produce-alert**.
- Step 11** Make sure raw regex is turned off:
- ```
sensor(config-sig-sig-str)# specify-raw-regex-string no
```



Note Raw Regex is regular expression syntax used for raw mode processing. It is expert mode only and targeted for use by the Cisco IPS signature development team or only those who are under supervision by the Cisco IPS signature development team. You can configure a String XL signature in either regular Regex or raw Regex.

- Step 12** Specify the regex string to search for in the TCP packet.
- ```
sensor(config-sig-sig-str-no)# regex-string tcpstring
```
- Step 13** Exit raw regex mode to configure optional String XL TCP parameters.
- ```
sensor(config-sig-sig-str-no)# exit  
sensor(config-sig-sig-str)#
```
- Step 14** Specify an exact match offset for this signature.
- ```
sensor(config-sig-sig-str)# specify-exact-match-offset yes
sensor(config-sig-sig-str-yes)# exact-match-offset 20
```




---

**Note** If you have exact match offset set to yes, you cannot configure maximum or minimum match offset. If you have exact match offset set to no, you can configure both maximum and minimum match offset at the same time.

---

- Step 15** Turn off exact match offset and specify a maximum match offset for this signature.
- ```
sensor(config-sig-sig-str-yes)# exit  
sensor(config-sig-sig-str)# specify-exact-match-offset no  
sensor(config-sig-sig-str-no)# specify-max-match-offset yes
```

```
sensor(config-sig-sig-str-no-yes)# max-match-offset 30
```

Step 16 Specify a minimum match offset for this signature.

```
sensor(config-sig-sig-str-no-yes)# exit
sensor(config-sig-sig-str-no)# specify-min-match-offset yes
sensor(config-sig-sig-str-no-yes)# min-match-offset 20
```

Step 17 Verify the settings.

```
sensor(config-sig-sig-str-no-yes)# exit
sensor(config-sig-sig-str-no)# exit
sensor(config-sig-sig-str)# show settings
string-xl-tcp
-----
event-action: produce-alert <defaulted>
strip-telnet-options: false <defaulted>
direction: to-service default: to-service
service-ports: 80
specify-max-stream-length
-----
no
-----
-----
specify-raw-regex-string
-----
no
-----
regex-string: tcpstring
dot-all: false <defaulted>
end-optional: false <defaulted>
no-case: false <defaulted>
stingy: false <defaulted>
utf8: false <defaulted>
specify-min-match-length
-----
no
-----
-----
swap-attacker-victim: false <defaulted>
specify-exact-match-offset
-----
no
-----
specify-max-match-offset
-----
yes
-----
max-match-offset: 30
-----
specify-min-match-offset
-----
yes
-----
min-match-offset: 20
-----
-----
-----
```

```
-----
sensor(config-sig-sig-str)#
```

Step 18 Exit signature definition submenu.

```
sensor(config-sig-sig-str)# exit
sensor(config-sig-sig)# exit
sensor(config-sig)# exit
Apply Changes:[yes]:
```

Step 19 Press **Enter** to apply the changes or enter **no** to discard them.

For More Information

For detailed information about the String XL signature engine, see [String XL Engines, page B-64](#).

Example String XL TCP Engine Minimum Match Length Signature



Caution

A custom signature can affect the performance of your sensor. Test the custom signature against a baseline sensor performance for your network to determine the overall impact of the signature.



Note

This procedure also applies to String XL UDP and String XL ICMP signatures, with the exception of the parameter **service-ports**, which does not apply to String XL ICMP signatures.

You can modify the following optional parameters to work with a specific Regex string:

- **dot-all true {true | false}**—If set to true, matches `[\x00-\xFF]` including `\n`; if set to false, matches anything in the range `[\x00-\xFF]` except `\n`. The default is false.
- **specify-min-match-length {yes | no}**—Enables minimum match length:
 - **min-match-length**—Specifies the maximum number of bytes the regular expression string must match for the pattern to be considered a hit. The value is 0 to 65535.
- **stingy {true | false}**—If set to true, specifies to stop looking for larger matches after the first completed match. The default is false.



Note

Stingy can only be used with **min-match-length**; otherwise, it is ignored.

- **utf8 {true | false}**—If set to true, treats all legal UTF-8 byte sequences in the expression as a single character. The default is false.

Creating a String XL TCP Engine Signature

The following example demonstrates how to create a custom String XL TCP signature that searches for minimum match length with stingy, dot all, and UTF-8 turned on.

To create a custom signature based on the String XL TCP engine that searches for minimum match length with stingy, dot all, and UTF-8 turned on, follow these steps:

Step 1 Log in to the CLI using an account with administrator or operator privileges.

Step 2 Enter signature definition submode.

```
sensor# configure terminal
sensor(config)# service signature-definition sig1
```

Step 3 Specify a signature ID and subsignature ID for the signature.

```
sensor(config-sig)# signatures 60004 0
```

Custom signatures are in the range of 60000 to 65000.

Step 4 Enter signature description submode.

```
sensor(config-sig-sig)# sig-description
```

Step 5 Specify a name for the new signature. You can also specify a additional comments about the sig using the **sig-comment** command or additional information about the signature using the **sig-string-info** command.

```
sensor(config-sig-sig-sig)# sig-name This is my new name
```

Step 6 Exit signature description submode.

```
sensor(config-sig-sig-sig)# exit
```

Step 7 Specify the String XL TCP engine.

```
sensor(config-sig-sig)# engine string-xl-tcp
```

Step 8 Specify the service ports.

```
sensor(config-sig-sig-str)# service-ports 80
```

Step 9 Specify the direction.

```
sensor(config-sig-sig-str)# direction to-service
```

Step 10 Change the event actions if needed according to your security policy by using the **event-action** command. The default event action is **produce-alert**.

Step 11 Make sure raw regex is turned off:

```
sensor(config-sig-sig-str)# specify-raw-regex-string no
```



Note Raw Regex is regular expression syntax used for raw mode processing. It is expert mode only and targeted for use by the Cisco IPS signature development team or only those who are under supervision by the Cisco IPS signature development team. You can configure a String XL signature in either regular Regex or raw Regex.

Step 12 Specify the regex string to search for in the TCP packet with dot all turned on.

```
sensor(config-sig-sig-str-no)# regex-string ht+p[\r].
sensor(config-sig-sig-str-no)# dot-all true
```

Step 13 Specify a minimum match length for this signature that can only be used with stingy.

```
sensor(config-sig-sig-str-no)# specify-min-match-length yes
sensor(config-sig-sig-str-no-yes)# min-match-length 100
sensor(config-sig-sig-str-no-yes)# exit
sensor(config-sig-sig-str-no)# stingy true
```

Step 14 Verify the settings:

```
sensor(config-sig-sig-str-no)# show settings
no
-----
regex-string: ht+p[\r\].
dot-all: true default: false
end-optional: false <defaulted>
no-case: false <defaulted>
stingy: true default: false
utf8: false <defaulted>
specify-min-match-length
-----
yes
-----
min-match-length: 100
-----
sensor(config-sig-sig-str-no)#
```

Step 15 Specify a new Regex string to search for and turn on UTF-8.

```
sensor(config-sig-sig-str-no)# regex-string \x5c\x31\x30\x2e\x30[\x00-\xff]+\
x2e\x31\x5c\x74\x65\x6d\x70
sensor(config-sig-sig-str-no)# utf8 true
```

Step 16 Verify the settings:

```
sensor(config-sig-sig-str-no)# show settings
no
-----
regex-string: \x5c\x31\x30\x2e\x30[\x00-\xff]+\x2e\x31\x5c\x74\x65\x6d\x70
dot-all: true default: false
end-optional: false <defaulted>
no-case: false <defaulted>
stingy: true default: false
utf8: true default: false
specify-min-match-length
-----
yes
-----
min-match-length: 100
-----
sensor(config-sig-sig-str-no)#
```

Step 17 Exit signature definition submenu.

```
sensor(config-sig-sig-str-no)# exit
sensor(config-sig-sig-str)# exit
sensor(config-sig-sig)# exit
sensor(config-sig)# exit
Apply Changes:[yes]:
```

Step 18 Press **Enter** to apply the changes or enter **no** to discard them.

For More Information

For detailed information about the String XL signature engine, see [String XL Engines, page B-64](#).



Configuring Event Action Rules

This chapter explains how to add event action rules policies and how to configure event action rules. It contains the following sections:

- [Event Action Rules Notes and Caveats, page 8-1](#)
- [Understanding Security Policies, page 8-2](#)
- [Understanding Event Action Rules, page 8-2](#)
- [Working With Event Action Rules Policies, page 8-8](#)
- [Event Action Variables, page 8-10](#)
- [Configuring Target Value Ratings, page 8-13](#)
- [Configuring Event Action Overrides, page 8-17](#)
- [Configuring Event Action Filters, page 8-20](#)
- [Configuring OS Identifications, page 8-26](#)
- [Configuring General Settings, page 8-33](#)
- [Configuring the Denied Attackers List, page 8-36](#)
- [Monitoring Events, page 8-39](#)

Event Action Rules Notes and Caveats

The following notes and caveats apply to configuring event action rules:

- Rate limiting and blocking are not supported for IPv6 traffic. If a signature is configured with a block or rate limit event action and is triggered by IPv6 traffic, an alert is generated but the action is not carried out.
- Global correlation inspection and the reputation filtering deny features do not support IPv6 addresses. For global correlation inspection, the sensor does not receive or process reputation data for IPv6 addresses. The risk rating for IPv6 addresses is not modified for global correlation inspection. Similarly, network participation does not include event data for attacks from IPv6 addresses. And finally, IPv6 addresses do not appear in the deny list.
- You must preface the event variable with a dollar (\$) sign to indicate that you are using a variable rather than a string.
- Connection blocks and network blocks are not supported on adaptive security appliances. Adaptive security appliances only support host blocks with additional connection information.

- You cannot delete the event action override for deny-packet-inline because it is protected. If you do not want to use that override, set the override-item-status to disabled for that entry.
- Passive OS fingerprinting is enabled by default and the IPS contains a default vulnerable OS list for each signature.

Understanding Security Policies

You can create multiple security policies and apply them to individual virtual sensors. A security policy is made up of a signature definition policy, an event action rules policy, and an anomaly detection policy. Cisco IPS contains a default signature definition policy called sig0, a default event action rules policy called rules0, and a default anomaly detection policy called ad0. You can assign the default policies to a virtual sensor or you can create new policies. The use of multiple security policies lets you create security policies based on different requirements and then apply these customized policies per VLAN or physical interface.

Understanding Event Action Rules

Event action rules are a group of settings you configure for the event action processing component of the sensor. These rules dictate the actions the sensor performs when an event occurs. The event action processing component is responsible for the following functions:

- Calculating the risk rating
- Adding event action overrides
- Filtering event action
- Executing the resulting event action
- Summarizing and aggregating events
- Maintaining a list of denied attackers

**Note**

Rate limiting and blocking are not supported for IPv6 traffic. If a signature is configured with a block or rate limit event action and is triggered by IPv6 traffic, an alert is generated but the action is not carried out.

Signature Event Action Processor

The Signature Event Action Processor coordinates the data flow from the signature event in the Alarm Channel to processing through the Signature Event Action Override, the Signature Event Action Filter, and the Signature Event Action Handler. It consists of the following components:

- Alarm Channel—The unit that represents the area to communicate signature events from the SensorApp inspection path to signature event handling.
- Signature Event Action Override—Adds actions based on the risk rating value. Signature Event Action Override applies to all signatures that fall in the range of the configured risk rating threshold. Each Signature Event Action Override is independent and has a separate configuration value for each action type.
- Signature Event Action Filter—Subtracts actions based on the signature ID, addresses, and risk rating of the signature event. The input to the Signature Event Action Filter is the signature event with actions possibly added by the Signature Event Action Override.



Note The Signature Event Action Filter can only subtract actions, it cannot add new actions.

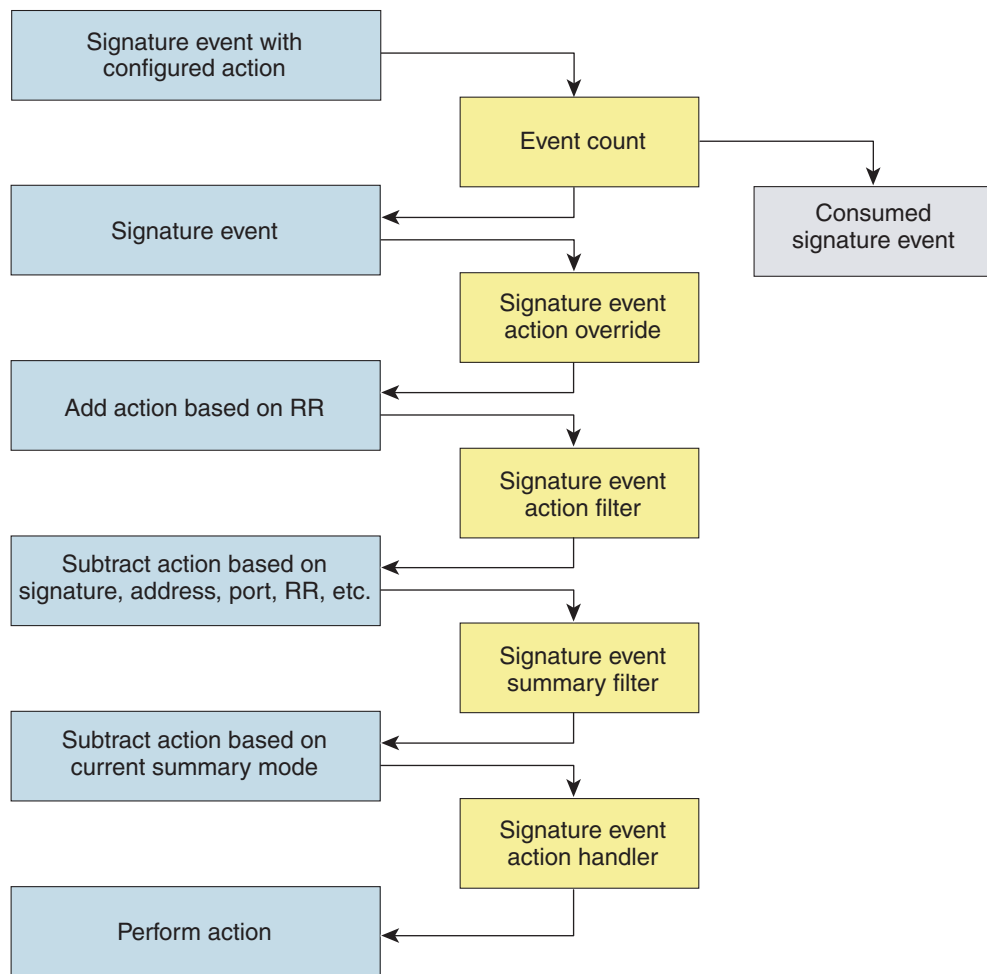
The following parameters apply to the Signature Event Action Filter:

- Signature ID
- Subsignature ID
- Attacker address
- Attacker port
- Victim address
- Victim port
- Risk rating threshold range
- Actions to subtract
- Sequence identifier (optional)
- Stop-or-continue bit
- Enable action filter line bit
- Victim OS relevance or OS relevance

- Signature Event Action Handler—Performs the requested actions. The output from the Signature Event Action Handler is the actions being performed and possibly an evIdsAlert written to the Event Store.

Figure 8-1 illustrates the logical flow of the signature event through the Signature Event Action Processor and the operations performed on the action for this event. It starts with the signature event with configured action received in the Alarm Channel and flows top to bottom as the signature event passes through the functional components of the Signature Event Action Processor.

Figure 8-1 Signature Event Through Signature Event Action Processor



132188

For More Information

For more information on risk rating, see [Calculating the Risk Rating, page 8-13](#).

Event Actions

An event action is the response of the sensor to an event. Event actions are configurable on a per-signature basis.

The IPS has the following event actions:

Alert and Log Actions

- produce-alert—Writes the event to the Event Store as an alert.



Note The produce-alert action is not automatic when you enable alerts for a signature. To have an alert created in the Event Store, you must select produce-alert. If you add a second action, you must include produce-alert if you want an alert sent to the Event Store. Also, every time you configure the event actions, a new list is created and it replaces the old list. Make sure you include all the event actions you need for each signature.



Note There are other event actions that force a produce-alert. These actions use produce-alert as the vehicle for performing the action. Even if produce-alert is not selected or is filtered, the alert is still produced. The actions are the following: produce-verbose-alert, request-snmp-trap, log-attacker-packets, log-victim-packets, and log-pair-packets.



Note A produce-alert event action is added for an event when global correlation has increased the risk rating of an event, and has added either the deny-packet-inline or deny-attacker-inline event action.

- produce-verbose-alert—Includes an encoded dump of the offending packet in the alert. This action causes an alert to be written to the Event Store, even if produce-alert is not selected.
- log-attacker-packets—Starts IP logging on packets that contain the attacker address and sends an alert. This action causes an alert to be written to the Event Store, even if produce-alert is not selected.
- log-victim-packets—Starts IP logging on packets that contain the victim address and sends an alert. This action causes an alert to be written to the Event Store, even if produce-alert is not selected.
- log-pair-packets—Starts IP logging on packets that contain the attacker/victim address pair. This action causes an alert to be written to the Event Store, even if produce-alert is not selected.
- request-snmp-trap—Sends a request to the Notification Application component of the sensor to perform SNMP notification. This action causes an alert to be written to the Event Store, even if produce-alert is not selected. You must have SNMP configured on the sensor to implement this action.

Deny Actions

- deny-packet-inline (inline only)—Terminates the packet.



Note You cannot delete the event action override for deny-packet-inline because it is protected. If you do not want to use that override, set the override-item-status to disabled for that entry.

- deny-connection-inline (inline only)—Terminates the current packet and future packets on this TCP flow.
- deny-attacker-victim-pair-inline (inline only)—Does not transmit this packet and future packets on the attacker/victim address pair for a specified period of time.
- deny-attacker-service-pair-inline (inline only)—Does not transmit this packet and future packets on the attacker address victim port pair for a specified period of time.
- deny-attacker-inline (inline only)—Terminates the current packet and future packets from this attacker address for a specified period of time.
- The sensor maintains a list of attackers being denied by the system. To remove an entry from the denied attacker list, you can view the list of attackers and clear the entire list, or you can wait for the timer to expire. The timer is a sliding timer for each entry. Therefore, if attacker A is being denied, but issues another attack, the timer for attacker A is reset and attacker A remains in the denied attacker list until the timer expires. If the denied attacker list is at capacity and cannot add a new entry, the packet is still denied.
- modify-packet-inline (inline only)—Modifies packet data to remove ambiguity about what the end point might do with the packet.



Note You cannot use modify-packet-inline as an action when adding event action filters or overrides.

Other Actions

- request-block-connection—Sends a request to ARC to block this connection. You must have blocking devices configured to implement this action.



Note Connection blocks and network blocks are not supported on adaptive security appliances. Adaptive security appliances only support host blocks with additional connection information.



Note IPv6 does not support request-block-connection.

- request-block-host—Sends a request to ARC to block this attacker host. You must have blocking devices configured to implement this action.



Note IPv6 does not support request-block-host.

- request-rate-limit—Sends a rate limit request to ARC to perform rate limiting. You must have rate limiting devices configured to implement this action.



Note The request-rate-limit action applies to a select set of signatures.



Note IPv6 does not support request-rate-limit.

- `reset-tcp-connection`—Sends TCP resets to hijack and terminate the TCP flow. The `reset-tcp-connection` action only works on TCP signatures that analyze a single connection. It does not work for sweeps or floods.

Understanding Deny Packet Inline

For signatures that have `deny-packet-inline` configured as an action or for an event action override that adds `deny-packet-inline` as an action, the following actions may be taken:

- `dropped-packet`
- `denied-flow`
- `tcp-one-way-reset-sent`

The `deny-packet-inline` action is represented as a dropped packet action in the alert. When a `deny-packet-inline` occurs for a TCP connection, it is automatically upgraded to a `deny-connection-inline` action and seen as a denied flow in the alert. If the IPS denies just one packet, the TCP continues to try to send that same packet again and again, so the IPS denies the entire connection to ensure it never succeeds with the resends.

When a `deny-connection-inline` occurs, the IPS also automatically sends a TCP one-way reset, which shows up as a TCP one-way reset sent in the alert. When the IPS denies the connection, it leaves an open connection on both the client (generally the attacker) and the server (generally the victim). Too many open connections can result in resource problems on the victim. So the IPS sends a TCP reset to the victim to close the connection on the victim side (usually the server), which conserves the resources of the victim. It also prevents a failover that would otherwise allow the connection to fail over to a different network path and reach the victim. The IPS leaves the attacker side open and denies all traffic from it.

TCP Reset Differences Between IPS Appliances and ASA IPS Modules

The IPS appliance sends TCP reset packets to both the attacker and victim when `reset-tcp-connection` is selected. The IPS appliance sends a TCP reset packet only to the victim under the following circumstances:

- When a `deny-packet-inline` or `deny-connection-inline` is selected
- When TCP-based signatures and `reset-tcp-connection` have NOT been selected

In the case of the ASA 5500-X IPS SSP and ASA 5585-X IPS SSP, the TCP reset request is sent to the ASA, and then the ASA sends the TCP reset packets. The ASA sends TCP reset packets to both the attacker and victim when the `reset-tcp-connection` is selected. When `deny-packet-inline` or `deny-connection-inline` is selected, the ASA sends the TCP reset packet to either the attacker or victim depending on the configuration of the signature. Signatures configured to swap the attacker and victim when reporting the alert can cause the ASA to send the TCP reset packet to the attacker.

TCP Normalizer Signature Warning

You receive the following warning if you disable a default-enabled TCP Normalizer signature or remove a default-enabled `modify-packet-inline`, `deny-packet-inline`, or `deny-connection-inline` action:

```
Use caution when disabling, retiring, or changing the event action settings of a <Sig ID>
TCP Normalizer signature for a sensor operating in IPS mode. The TCP Normalizer signature
default values are essential for proper operation of the sensor.
If the sensor is seeing duplicate packets, consider assigning the traffic to multiple
virtual sensors. If you are having problems with asymmetric or out-of-order TCP packets,
consider changing the normalizer mode from strict evasion protection to asymmetric mode
protection. Contact Cisco TAC if you require further assistance.
```

Understanding deny-packet-inline and reset-tcp-connection

Pay attention to the following when configuring `deny-packet-inline` and `reset-tcp-connection`:

- If you want to deny attack packets from reaching the victim and also reset the TCP connection for that flow, then you must configure BOTH deny-packet-inline AND reset-tcp-connection.
- Configuring reset-tcp-connection alone only resets the TCP connection but the attack packet is not denied from reaching the victim.
- Configuring deny-packet-inline alone only denies the attack packet from reaching the victim. It does not trigger a TCP reset.

For More Information

- For procedure for configuring denied attackers, see [Monitoring and Clearing the Denied Attackers List, page 8-37](#).
- For the procedure for configuring the general settings, see [Configuring the General Settings, page 8-34](#).
- For the procedures for configuring blocking devices, see [Chapter 14, “Configuring Attack Response Controller for Blocking and Rate Limiting.”](#)
- For the procedures for configuring SNMP, see [Chapter 15, “Configuring SNMP.”](#)

Event Action Rules Configuration Sequence

Follow these steps when configuring the event action rules component of the IPS:

1. Create any variables that you want to use in event action filters.
2. Create target value ratings. Assign target value ratings to your network assets so that you can calculate the risk rating.
3. Create overrides to add actions based on the risk rating value. Assign a risk rating to each event action type.
4. Create filters. Assign filters to subtract actions based on the ID, IP addresses, and risk rating of the signature.
5. Create OS mappings. OS mappings are used for the attack relevance rating in the calculation of the risk rating for an alert.
6. Configure the general settings. Specify whether you want to use the summarizer, the meta event generator, or configure denied attacker parameters.

Working With Event Action Rules Policies

Use the **service event-action-rules** *name* command in service event action rules submode to create an event action rules policy. The values of this event action rules policy are the same as the default event action rules policy, rules0, until you edit them. Or you can use the **copy event-action-rules** *source_destination* command in privileged EXEC mode to make a copy of an existing policy and then edit the values of the new policy as needed. Use the **list event-action-rules-configurations** command in privileged EXEC mode to list the event action rules policies. Use the **no service event-action-rules** *name* command in global configuration mode to delete an event action rules policy. Use the **default service event-action-rules** *name* command in global configuration mode to reset the event action rules policy to factory settings.

Working With Event Action Rules Policies

To create, copy, display, edit, and delete event action rules policies, follow these steps:

Step 1 Log in to the CLI using an account with administrator privileges.

Step 2 Create an event action rules policy.

```
sensor# configure terminal
sensor(config)# service event-action-rules MyRules
sensor(config-eve)# exit
Apply Changes?[yes]: yes
sensor(config)# exit
sensor#
```

Step 3 Copy an existing event action rules policy to a new event action rules policy.

```
sensor# copy event-action-rules rules0 rules1
sensor#
```



Note You receive an error if the policy already exists or if there is not enough space available for the new policy.

Step 4 Accept the default event action rules policy values or edit the following parameters.

- a. Add event action rules variables.
- b. Configure event action rules overrides.
- c. Configure event action rules filters.
- d. Configure the event action rules general settings.
- e. Configure the event action rules target value rating.
- f. Configure the event action rules OS identification settings.

Step 5 Display a list of event action rules policies on the sensor:

```
sensor# list event-action-rules-configurations
Event Action Rules
  Instance  Size  Virtual Sensor
  rules0    255  vs0
  temp      707  N/A
  MyRules   255  N/A
  rules1    141  vs1
sensor#
```

Step 6 Delete an event action rules policy.

```
sensor(config)# no service event-action-rules MyRules
sensor(config)#
```



Note You cannot delete the default event action rules policy, rules0.

Step 7 Confirm the event action rules instance has been deleted.

```
sensor# list event-action-rules-configurations
Event Action Rules
  Instance  Size  Virtual Sensor
  rules0    112  vs0
  rules1    142  N/A
```

```
sensor#
```

Step 8 Reset an event action rules policy to factory settings.

```
sensor# configure terminal
sensor(config)# default service event-action-rules rules1
sensor(config)#
```

For More Information

- For the procedure for adding event action rules variables, see [Event Action Variables, page 8-10](#).
- For the procedure for configuring event action rules overrides, see [Configuring Event Action Overrides, page 8-17](#).
- For the procedure for configuring event action rules filters, see [Configuring Event Action Filters, page 8-20](#).
- For the procedure for configuring the general settings, see [Configuring General Settings, page 8-33](#).
- For the procedure for configuring event action rules target value ratings, see [Configuring Target Value Ratings, page 8-13](#).
- For the procedure for configuring OS maps, see [Configuring OS Identifications, page 8-26](#).

Event Action Variables

This section describes event action variables, and contains the following topics:

- [Understanding Event Action Variables, page 8-10](#)
- [Adding, Editing, and Deleting Event Action Variables, page 8-11](#)

Understanding Event Action Variables



Note

Global correlation inspection and the reputation filtering deny features do not support IPv6 addresses. For global correlation inspection, the sensor does not receive or process reputation data for IPv6 addresses. The risk rating for IPv6 addresses is not modified for global correlation inspection. Similarly, network participation does not include event data for attacks from IPv6 addresses. And finally, IPv6 addresses do not appear in the deny list.



Note

Rate limiting and blocking are not supported for IPv6 traffic. If a signature is configured with a block or rate limit event action and is triggered by IPv6 traffic, an alert is generated but the action is not carried out.

You can create event variables and then use those variables in event action filters. When you want to use the same value within multiple filters, use a variable. When you change the value of the variable, any filter that uses that variable is updated with the new value.

**Note**

You must preface the event variable with a dollar (\$) sign to indicate that you are using a variable rather than a string.

Some variables cannot be deleted because they are necessary to the signature system. If a variable is protected, you cannot select it to edit it. You receive an error message if you try to delete protected variables. You can edit only one variable at a time.

IPv4 Addresses

When configuring IPv4 addresses, specify the full IP address or ranges or set of ranges:

- 192.0.2.3-192.0.2.26
- 10.90.1.1
- 192.56.10.1-192.56.10.255
- 10.1.1.1-10.2.255.255, 192.0.2.3-192.0.2.26

IPv6 Addresses

When configuring IPv6 addresses, use the following format:

```
<XXXX:XXXX:XXXX:XXXX:XXXX:XXXX:XXXX:XXXX>-<XXXX:XXXX:XXXX:XXXX:XX
XX:XXXX:XXXX:XXXX>[,<XXXX:XXXX:XXXX:XXXX:XXXX:XXXX:XXXX:XXXX>-<XXX
X:XXXX:XXXX:XXXX:XXXX:XXXX:XXXX>]
```

**Note**

IPv6 addresses are 128 bits represented in hexadecimal and divided into eight 16-bit groups separated by colons. You can skip the leading zeros and you can represent the zeroed groups in the middle with a double colon (::). You must start the address with the 2001:db8 prefix.

**Timesaver**

If you have an IP address space that applies to your engineering group and there are no Windows systems in that group, and you are not worried about any Windows-based attacks to that group, you could set up a variable to be the IP address space of the engineering group. You could then use this variable to configure a filter that would ignore all Windows-based attacks for this group.

Adding, Editing, and Deleting Event Action Variables

**Note**

Global correlation inspection and the reputation filtering deny features do not support IPv6 addresses. For global correlation inspection, the sensor does not receive or process reputation data for IPv6 addresses. The risk rating for IPv6 addresses is not modified for global correlation inspection. Similarly, network participation does not include event data for attacks from IPv6 addresses. And finally, IPv6 addresses do not appear in the deny list.

**Note**

Rate limiting and blocking are not supported for IPv6 traffic. If a signature is configured with a block or rate limit event action and is triggered by IPv6 traffic, an alert is generated but the action is not carried out.

Use the **variables** *variable_name* **address** *ip_address* command in service event action rules submode to create an IPv4 event action variable. The IPv4 address can be one address, a range, or ranges separated by a comma. Use the **variables** *variable_name* **ipv6-address** *ip_address* command in service event action rules submode to create an IPv6 event action variable. Use the **no variables** *variable_name* command in service event action rules submode to delete an event action variable.

**Note**

IPv6 addresses are 128 bits represented in hexadecimal and divided into eight 16-bit groups separated by colons. You can skip the leading zeros and you can represent the zeroed groups in the middle with a double colon (::). You must start the address with the 2001:db8 prefix.

Working With Event Action Variables

To add, delete, and edit event action variables, follow these steps:

Step 1 Log in to the CLI using an account with administrator privileges.

Step 2 Enter event action rules submode.

```
sensor# configure terminal
sensor(config)# service event-action-rules rules0
```

Step 3 Add an IPv4 event action rules variable. The valid values for **address** are A.B.C.D-A.B.C.D [,A.B.C.D-A.B.C.D].

```
sensor(config-eve)# variables variable-ipv4 address 192.0.2.3
```

Step 4 Add an IPv6 event action rules variable. The valid form for **ipv6-address** is:

```
<XXXX:XXXX:XXXX:XXXX:XXXX:XXXX:XXXX:XXXX>-<XXXX:XXXX:XXXX:XXXX:XX
XX:XXXX:XXXX:XXXX>[,<XXXX:XXXX:XXXX:XXXX:XXXX:XXXX:XXXX:XXXX>-<XXX
X:XXXX:XXXX:XXXX:XXXX:XXXX:XXXX:XXXX>]
```

```
sensor(config-eve)# variables variable-ipv6 ipv6-address
2001:0db8:3c4d:0015:0000:0000:abcd:ef12
```

Step 5 Verify that you added the event action rules variable.

```
sensor(config-eve)# show settings
variables (min: 0, max: 256, current: 2)
-----
variableName: variable-ipv6
-----
ipv6-address: 2001:0db8:3c4d:0015:0000:0000:abcd:ef12 default: ::0-FFFF
:FFFF:FFFF:FFFF:FFFF:FFFF:FFFF:FFFF:FFFF
-----
variableName: variable-ipv4
-----
address: 192.0.2.3 default: 0.0.0.0-255.255.255.255
-----
```

Step 6 To edit an event action rules variable, change the IPv6 address to a range.

```
sensor(config-eve)# variables variable-ipv6 ipv6-address
::0-FFFF:FFFF:FFFF:FFFF:FFFF:FFFF:FFFF:FFFF
```

Step 7 Verify that you edited the event action rules variable.

```
sensor(config-eve)# show settings
variables (min: 0, max: 256, current: 2)
-----
variableName: variable-ipv6
```

```

-----
      ipv6-address: ::0-FFFF:FFFF:FFFF:FFFF:FFFF:FFFF:FFFF:FFFF default: ::0
FFFF:FFFF:FFFF:FFFF:FFFF:FFFF:FFFF:FFFF
-----

```

Step 8 Delete an event action rules variable.

```
sensor(config-eve)# no variables variable-ipv6
```

Step 9 Verify the event action rules variable you deleted.

```

sensor(config-eve)# show settings
  variables (min: 0, max: 256, current: 1)
-----
  variableName: variableipv4
-----
      address: 192.0.2.3 default: 0.0.0.0-255.255.255.255
-----
-----

```

Step 10 Exit event action rules submode.

```
sensor(config-eve)# exit
Apply Changes?[yes]:
```

Step 11 Press **Enter** to apply your changes or enter **no** to discard them.

Configuring Target Value Ratings

This section describes what risk rating is and how to use it to configure target value ratings. This section contains the following topics:

- [Calculating the Risk Rating, page 8-13](#)
- [Understanding Threat Rating, page 8-15](#)
- [Adding, Editing, and Deleting Target Value Ratings, page 8-15](#)

Calculating the Risk Rating

A risk rating (RR) is a value between 0 and 100 that represents a numerical quantification of the risk associated with a particular event on the network. The calculation takes into account the value of the network asset being attacked (for example, a particular server), so it is configured on a per-signature basis using the attack severity rating and the signature fidelity rating, and on a per-server basis using the target value rating. The risk rating is calculated from several components, some of which are configured, some collected, and some derived.



Note

The risk rating is associated with alerts not signatures.

Risk ratings let you prioritize alerts that need your attention. These risk rating factors take into consideration the severity of the attack if it succeeds, the fidelity of the signature, the reputation score of the attacker from the global correlation data, and the overall value of the target host to you. The risk rating is reported in the evIdsAlert.

The following values are used to calculate the risk rating for a particular event:

- **Signature fidelity rating (SFR)**—A weight associated with how well this signature might perform in the absence of specific knowledge of the target. The signature fidelity rating is configured per signature and indicates how accurately the signature detects the event or condition it describes.

Signature fidelity rating is calculated by the signature author on a per-signature basis. The signature author defines a baseline confidence ranking for the accuracy of the signature in the absence of qualifying intelligence on the target. It represents the confidence that the detected behavior would produce the intended effect on the target platform if the packet under analysis were allowed to be delivered. For example, a signature that is written with very specific rules (specific regular expression) has a higher signature fidelity rating than a signature that is written with generic rules.



Note The signature fidelity rating does not indicate how bad the detected event may be.

- **Attack severity rating (ASR)**—A weight associated with the severity of a successful exploit of the vulnerability. The attack severity rating is derived from the alert severity parameter (informational, low, medium, or high) of the signature. The attack severity rating is configured per signature and indicates how dangerous the event detected is.



Note The attack severity rating does not indicate how accurately the event is detected.

- **Target value rating (TVR)**—A weight associated with the perceived value of the target. Target value rating is a user-configurable value (zero, low, medium, high, or mission critical) that identifies the importance of a network asset (through its IP address). You can develop a security policy that is more stringent for valuable corporate resources and looser for less important resources. For example, you could assign a target value rating to the company web server that is higher than the target value rating you assign to a desktop node. In this example, attacks against the company web server have a higher risk rating than attacks against the desktop node. Target value rating is configured in the event action rules policy.
- **Attack relevance rating (ARR)**—A weight associated with the relevancy of the targeted operating system. Attack relevancy rating is a derived value (relevant, unknown, or not relevant), which is determined at alert time. The relevant operating systems are configured per signature.
- **Promiscuous delta (PD)**—A weight associated with the promiscuous delta, which can be subtracted from the overall risk rating in promiscuous mode. Promiscuous delta is in the range of 0 to 30 and is configured per signature.



Note If the trigger packet is not inline, the promiscuous delta is subtracted from the rating.

- **Watch list rating (WLR)**—A weight associated with the CSA MC watch list in the range of 0 to 100 (CSA MC only uses the range 0 to 35). If the attacker for the alert is found on the watch list, the watch list rating for that attacker is added to the rating.

Figure 8-2 illustrates the risk rating formula:

Figure 8-2 Risk Rating Formula

$$RR = \frac{ASR * TVR * SFR}{10000} + ARR - PD + WLR$$

191016

Understanding Threat Rating

Threat rating is risk rating that has been lowered by event actions that have been taken. Nonlogging event actions have a threat rating adjustment. The largest threat rating from all the event actions taken is subtracted from the risk rating. The event actions have the following threat ratings:

- deny-attacker-inline—45
- deny-attacker-victim-pair-inline—40
- deny-attacker-service-pair-inline—40
- deny-connection-inline—35
- deny-packet-inline—35
- modify-packet-inline—35
- request-block-host—20
- request-block-connection—20
- reset-tcp-connection—20
- request-rate-limit—20

Adding, Editing, and Deleting Target Value Ratings

**Note**

Global correlation inspection and the reputation filtering deny features do not support IPv6 addresses. For global correlation inspection, the sensor does not receive or process reputation data for IPv6 addresses. The risk rating for IPv6 addresses is not modified for global correlation inspection. Similarly, network participation does not include event data for attacks from IPv6 addresses. And finally, IPv6 addresses do not appear in the deny list.

**Note**

Rate limiting and blocking are not supported for IPv6 traffic. If a signature is configured with a block or rate limit event action and is triggered by IPv6 traffic, an alert is generated but the action is not carried out.

You can assign a target value rating to your network assets. The target value rating is one of the factors used to calculate the risk rating value for each alert. You can assign different target value ratings to different targets. Events with a higher risk rating trigger more severe signature event actions.

For IPv4 address, use the **target-value** {**zerovalue** | **low** | **medium** | **high** | **mission-critical**} **target-address** *ip_address* command in service event action rules submode to add target value ratings for your network assets. The default is medium. Use the **no target-value** {**zerovalue** | **low** | **medium** | **high** | **mission-critical**} command in service event action rules submode to delete target value ratings.

For IPv6 addresses, use the **ipv6-target-value** {**zerovalue** | **low** | **medium** | **high** | **mission-critical**} **ipv6-target-address** *ip_address* command in service event action rules submode to add target value ratings for your network assets. The default is medium. Use the **no ipv6-target-value** {**zerovalue** | **low** | **medium** | **high** | **mission-critical**} command in service event action rules submode to delete target value ratings.

The following commands apply:

- **target-value**—Specifies the IPv4 target value rating:
 - **zerovalue**—No value of this target.
 - **low**—Lower value of this target.
 - **medium**—Normal value of this target (default).
 - **high**—Elevated value of this target.
 - **mission-critical**—Extreme value of this target.
- **no target-value**—Removes the IPv4 target value rating.
- **target-address ip_address**—Specifies the range set of IP address(es) for IPv4 addresses in the following form: <A.B.C.D>-<A.B.C.D>[,<A.B.C.D>-<A.B.C.D>]
- **ipv6-target-value**—Specifies the IPv6 target value rating:
 - **zerovalue**—No value of this target.
 - **low**—Lower value of this target.
 - **medium**—Normal value of this target (default).
 - **high**—Elevated value of this target.
 - **mission-critical**—Extreme value of this target.
- **no ipv6-target-value**—Removes the IPv6 target value rating.
- **ipv6-target-address ip_address**—Specifies the range set of IP address(es) for IPv6 addresses in the following form:
 <XXXX:XXXX:XXXX:XXXX:XXXX:XXXX:XXXX:XXXX>-<XXXX:XXXX:XXXX:XXXX:XXXX:XXXX:XXXX:XXXX>[,<XXXX:XXXX:XXXX:XXXX:XXXX:XXXX:XXXX:XXXX>-<XXXX:XXXX:XXXX:XXXX:XXXX:XXXX:XXXX:XXXX>]

Adding, Editing, and Deleting Target Value Ratings

To add, edit, and delete target value ratings for your network assets, follow these steps:

Step 1 Log in to the CLI using an account with administrator privileges.

Step 2 Enter event action rules submode.

```
sensor# configure terminal
sensor(config)# service event-action-rules rules1
```

Step 3 Assign an IPv4 target value rating to the network asset.

```
sensor(config-eve)# target-value mission-critical target-address 192.0.2.0
```

Step 4 Assign an IPv6 target value rating to the network asset.

```
sensor(config-eve)# ipv6-target-value mission-critical ipv6-target-address
2001:0db8:3c4d:0015:0000:0000:abcd:ef12
```

Step 5 Verify that you added the target value rating.

```
sensor(config-eve)# show settings
-----
target-value (min: 0, max: 5, current: 1)
-----
target-value-setting: mission-critical
target-address: 192.0.2.0 default: 0.0.0.0-255.255.255.255
-----
```

```

ipv6-target-value (min: 0, max: 5, current: 2)
-----
  ipv6-target-value-setting: mission-critical
  ipv6-target-address: 2001:0db8:3c4d:0015:0000:0000:abcd:ef12 default: ::0-
FFFF:FFFF:FFFF:FFFF:FFFF:FFFF:FFFF:FFFF
-----
sensor(config-eve)#

```

Step 6 To edit a target value rating, change the target value rating setting of the asset.

```

sensor(config-eve)# target-value low target-address 192.0.2.0

```

Step 7 Verify that you edited the target value rating.

```

sensor(config-eve)# show settings
-----
target-value (min: 0, max: 5, current: 1)
-----
  target-value-setting: low
  target-address: 192.0.2.0 default: 0.0.0.0-255.255.255.255
-----

```

Step 8 Delete the target value rating.

```

sensor(config-eve)# no ipv6-target-value mission-critical

```

Step 9 Verify that you deleted the target value rating.

```

sensor(config-eve)# show settings
-----
ipv6-target-value (min: 0, max: 5, current: 0)
-----
-----

```

Step 10 Exit event action rules submode.

```

sensor(config-rul)# exit
Apply Changes?[yes]:

```

Step 11 Press **Enter** to apply your changes or enter **no** to discard them.

Configuring Event Action Overrides

This section describes event action overrides, and contains the following topics:

- [Understanding Event Action Overrides, page 8-17](#)
- [Adding, Editing, Enabling, and Disabling Event Action Overrides, page 8-18](#)

Understanding Event Action Overrides

You can add an event action override to change the actions associated with an event based on the risk rating of that event. Event action overrides are a way to add event actions globally without having to configure each signature individually. Each event action has an associated risk rating range. If a signature event occurs and the risk rating for that event falls within the range for an event action, that

action is added to the event. For example, if you want any event with a risk rating of 85 or more to generate an SNMP trap, you can set the risk rating range for request-snmpt-trap to 85-100. If you do not want to use action overrides, you can disable the entire event action override component.

**Note**

Connection blocks and network blocks are not supported on adaptive security appliances. Adaptive security appliances only support host blocks with additional connection information.

Adding, Editing, Enabling, and Disabling Event Action Overrides

Use the overrides {**request-block-connection** | **request-block-host** | **deny-attacker-inline** | **deny-packet-inline** | **deny-attacker-service-pair-inline** | **deny-attacker-victim-pair-inline** | **deny-connection-inline** | **log-attacker-packets** | **log-victim-packets** | **log-pair-packets** | **reset-tcp-connection** | **produce-alert** | **produce-verbose-alert** | **request-rate-limit** | **request-snmpt-trap**} command in service event action rules submode to configure the parameters of event action overrides. Use the **no overrides** command in service event action rules submode to delete the parameters of event action overrides.

Configure the override event actions, then the risk rating range, then enable or disable the override.

**Note**

You cannot delete the event action override for deny-packet-inline because it is protected. If you do not want to use that override, set the override-item-status to disabled for that entry.

The following commands apply:

- **no overrides**—Removes an entry or selection setting.
- **override-item-status** {**enabled** | **disabled**}—Enables or disables the use of this override item. The default is enabled.
- **risk-rating-range**—Specifies the range of risk rating values for this override item. The default is 0 to 100.
- **show**—Displays system settings and/or history information.

Configuring Event Action Overrides

To add event action overrides, follow these steps:

Step 1 Log in to the CLI using an account with administrator privileges.

Step 2 Enter event action rules submode.

```
sensor# configure terminal
sensor(config)# service event-action-rules rules0
sensor(config-eve)#
```

Step 3 Assign the action for the override:

- Deny packets from the source IP address of the attacker.


```
sensor(config-eve)# overrides deny-attacker-inline
sensor(config-eve-ove)#
```
- Do not transmit the single packet causing the alert.


```
sensor(config-eve)# overrides deny-packet-inline
sensor(config-eve-ove)#
```


- Do not transmit packets on the specified TCP connection.

```
sensor(config-eve)# overrides deny-connection-inline
sensor(config-eve-ove)#
```

- Send TCP RST packets to terminate the connection.

```
sensor(config-eve)# overrides reset-tcp-connection
sensor(config-eve-ove)#
```

- Request a block of the connection.

```
sensor(config-eve)# overrides request-block-connection
sensor(config-eve-ove)#
```

- Request a block of the attacker host.

```
sensor(config-eve)# overrides request-block-host
sensor(config-eve-ove)#
```

- Log the packets from the attacker IP address.

```
sensor(config-eve)# overrides log-attacker-packets
sensor(config-eve-ove)#
```

- Log the packets from the victim IP address.

```
sensor(config-eve)# overrides log-victim-packets
sensor(config-eve-ove)#
```

- Log packets from both the attacker and victim IP addresses.

```
sensor(config-eve)# overrides log-pair-packets
sensor(config-eve-ove)#
```

- Write an alert to Event Store.

```
sensor(config-eve)# overrides produce-alert
sensor(config-eve-ove)#
```

- Write verbose alerts to Event Store.

```
sensor(config-eve)# overrides produce-verbose-alert
sensor(config-eve-ove)#
```

- Write events that request an SNMP trap to the Event Store.

```
sensor(config-eve)# overrides request-snmp-trap
sensor(config-eve-ove)#
```

- Step 4** Configure the risk rating for this override item. The default risk rating range is 0 to 100. Set it to a different value, such as 85 to 100.

```
sensor(config-eve-ove)# risk-rating-range 85-100
```

- Step 5** Enable or disable the use of this override item. The default is enabled.

```
sensor(config-eve-ove)# override-item-status {enabled | disabled}
```

- Step 6** Verify the settings.

```
sensor(config-eve-ove)# exit
sensor(config-eve)# show settings
action-to-add: deny-attacker-inline
-----
override-item-status: Enabled default: Enabled
```

```

risk-rating-range: 85-100 default: 0-100
-----

```

Step 7 Edit the risk rating of an event action override.

```

sensor(config-eve)# overrides deny-attacker-inline
sensor(config-eve-ove)# risk-rating 95-100

```

Step 8 Verify that you edited the event action override.

```

sensor(config-eve-ove)# exit
sensor(config-eve)# show settings
-----
overrides (min: 0, max: 14, current: 1)
-----

override-item-status: Enabled <defaulted>
risk-rating-range: 95-100 default: 0-100
-----

```

Step 9 Delete the event action override.

```

sensor(config-eve)# no overrides deny-attacker-inline
sensor(config-eve-ove)#

```

Step 10 Verify that you deleted the event action override.

```

sensor(config-eve-ove)# exit
sensor(config-eve)# show settings
overrides (min: 0, max: 14, current: 1)
-----
action-to-add: deny-attacker-inline
-----
override-item-status: Enabled <defaulted>
risk-rating-range: 95 default: 0-100
-----
override-item-status: Enabled <defaulted>
risk-rating-range: 90-100 <defaulted>
-----
-----

```

Step 11 Exit event action rules submode.

```

sensor(config-eve)# exit
Apply Changes?[yes]:

```

Step 12 Press **Enter** to apply your changes or enter **no** to discard them.

For More Information

For a detailed description of all the event actions, see [Event Actions, page 8-5](#).

Configuring Event Action Filters

This section describes event action filters, and contains the following topics:

- [Understanding Event Action Filters, page 8-21](#)
- [Configuring Event Action Filters, page 8-21](#)

Understanding Event Action Filters

**Note**

Global correlation inspection and the reputation filtering deny features do not support IPv6 addresses. For global correlation inspection, the sensor does not receive or process reputation data for IPv6 addresses. The risk rating for IPv6 addresses is not modified for global correlation inspection. Similarly, network participation does not include event data for attacks from IPv6 addresses. And finally, IPv6 addresses do not appear in the deny list.

**Note**

Rate limiting and blocking are not supported for IPv6 traffic. If a signature is configured with a block or rate limit event action and is triggered by IPv6 traffic, an alert is generated but the action is not carried out.

Event action filters are processed as an ordered list and you can move filters up or down in the list. Filters let the sensor perform certain actions in response to the event without requiring the sensor to perform all actions or remove the entire event. Filters work by removing actions from an event. A filter that removes all actions from an event effectively consumes the event.

**Note**

When filtering sweep signatures, we recommend that you do not filter the destination addresses. If there are multiple destination addresses, only the last address is used for matching the filter.

**Caution**

Event action filters based on source and destination IP addresses do not function for the Sweep engine, because they do not filter as regular signatures. To filter source and destination IP addresses in sweep alerts, use the source and destination IP address filter parameters in the Sweep engine signatures.

Configuring Event Action Filters

**Note**

Global correlation inspection and the reputation filtering deny features do not support IPv6 addresses. For global correlation inspection, the sensor does not receive or process reputation data for IPv6 addresses. The risk rating for IPv6 addresses is not modified for global correlation inspection. Similarly, network participation does not include event data for attacks from IPv6 addresses. And finally, IPv6 addresses do not appear in the deny list.

**Note**

Rate limiting and blocking are not supported for IPv6 traffic. If a signature is configured with a block or rate limit event action and is triggered by IPv6 traffic, an alert is generated but the action is not carried out.

You can configure event action filters to remove specific actions from an event or to discard an entire event and prevent further processing by the sensor. You can use event action variables that you defined to group addresses for your filters.

**Note**

You must preface the event variable with a dollar sign (\$) to indicate that you are using a variable rather than a string. Otherwise, you receive the `Bad source and destination error`.

Use the **filters** `{edit | insert | move} name1 [begin | end | inactive | before | after]` command in service event action rules submode to set up event action filters.

The following commands apply:

- **actions-to-remove**—Specifies the event actions to remove for this filter item.
- **attacker-address-range**—Specifies the range set of IPv4 attacker address(es) for this item (for example, 192.0.2.0-192.0.2.254,192.3.2.0-192.3.2.254).

**Note**

The second IP address in the range must be greater than or equal to the first IP address. If you do not specify an attacker address range, all IPv4 attacker addresses are matched.

- **attacker-port-range**—Specifies the range set of attacker port(s) for this item (for example, 147-147,8000-10000).
- **default**—Sets the value back to the system default setting.
- **deny-attacker-percentage**—Specifies the percentage of packets to deny for deny attacker features. The valid range is 0 to 100. The default is 100.
- **filter-item-status** `{enabled | disabled}`—Enables or disables the use of this filter item.
- **ipv6-attacker-address-range**—Specifies the range set of IPv6 attacker address(es) for this item (for example, `<XXXX:XXXX:XXXX:XXXX:XXXX:XXXX:XXXX:XXXX>-<XXXX:XXXX:XXXX:XXXX:XXXX:XXXX:XXXX:XXXX>[,<XXXX:XXXX:XXXX:XXXX:XXXX:XXXX:XXXX:XXXX>-<XXXX:XXXX:XXXX:XXXX:XXXX:XXXX:XXXX:XXXX>]`).

**Note**

The second IPv6 address in the range must be greater than or equal to the first IPv6 address. If you do not specify an IPv6 attacker address range, all IPv6 attacker addresses are matched.

- **ipv6-victim-address-range**—Specifies the range set of victim address(es) for this item (for example, `<XXXX:XXXX:XXXX:XXXX:XXXX:XXXX:XXXX:XXXX>-<XXXX:XXXX:XXXX:XXXX:XXXX:XXXX:XXXX:XXXX>[,<XXXX:XXXX:XXXX:XXXX:XXXX:XXXX:XXXX:XXXX>-<XXXX:XXXX:XXXX:XXXX:XXXX:XXXX:XXXX:XXXX>]`).

**Note**

The second IPv6 address in the range must be greater than or equal to the first IPv6 address. If you do not specify an IPv6 victim address range, all IPv6 victim addresses are matched.

- **no**—Removes an entry or selection setting.
- **os-relevance**—Specifies the event OS relevance for this filter:
 - **relevant**—Specifies that the event is relevant to the target OS.
 - **not-relevant**—Specifies that the event is not relevant to the target OS.
 - **unknown**—It is unknown whether the event is relevant to the target OS.
- **risk-rating-range**—Specifies the range of risk rating values for this filter item.

- **signature-id-range**—Specifies the range set of signature ID(s) for this item (for example, 1000-2000,3000-3000).
- **stop-on-match {true | false}**—Specifies to continue evaluating filters or stop when this filter item is matched.
- **subsignature-id-range**—Specifies the range set of subsignature ID(s) for this item (for example, 0-2,5-5).
- **user-comment** —Lets you add your comments about this filter item.
- **victim-address-range**—Specifies the range set of victim address(es) for this item (for example, 10.20.1.0-10.20.1.255,10.20.5.0-10.20.5.255).



Note The second IP address in the range must be greater than or equal to the first IP address. If you do not specify a victim address range, all IPv4 attacker addresses are matched.

- **victim-port-range**—Specifies the range set of victim port(s) for this item (for example, 147-147,8000-10000).

Configuring Event Action Filters

To configure event action filters, follow these steps:

Step 1 Log in to the CLI using an account with administrator privileges.

Step 2 Enter event action rules submode.

```
sensor# configure terminal
sensor(config)# service event-action-rules rules1
sensor(config-eve)#
```

Step 3 Create the filter name. Use **name1**, **name2**, and so forth to name your event action filters. Use the **begin** | **end** | **inactive** | **before** | **after** keywords to specify where you want to insert the filter.

```
sensor(config-eve)# filters insert name1 begin
```

Step 4 Specify the values for this filter:

- a. Specify the signature ID range. The default is 900 to 65535.

```
sensor(config-eve-fil)# signature-id-range 1000-1005
```

- b. Specify the subsignature ID range. The default is 0 to 255.

```
sensor(config-eve-fil)# subsignature-id-range 1-5
```

- c. Specify the attacker address range for IPv4 or IPv6.

```
sensor(config-eve-fil)# attacker-address-range 192.0.2.3-192.0.2.26
sensor(config-eve-fil)# ipv6-attacker-address-range
2001:0db8:3c4d:0015:0000:0000:abcd:ef12
```

- d. Specify the victim address range for IPv4 or IPv6.

```
sensor(config-eve-fil)# victim-address-range 192.56.10.1-192.56.10.255
sensor(config-eve-fil)# ipv6-victim-address-range ::0-FFFF:FFFF:FFFF:FFFF:FFFF:
FFFF:FFFF:FFFF
```

- e. Specify the victim port range. The default is 0 to 65535.

```
sensor(config-eve-fil)# victim-port-range 0-434
```

- f. Specify the OS relevance. The default is 0 to 100.

```
sensor(config-eve-fil)# os-relevance relevant
```

- g. Specify the risk rating range. The default is 0 to 100.

```
sensor(config-eve-fil)# risk-rating-range 85-100
```

- h. Specify the actions to remove.

```
sensor(config-eve-fil)# actions-to-remove reset-tcp-connection
```

- i. If you are filtering a deny action, set the percentage of deny actions you want. The default is 100.

```
sensor(config-eve-fil)# deny-attacker-percentage 90
```

- j. Specify the status of the filter to either disabled or enabled. The default is enabled.

```
sensor(config-eve-fil)# filter-item-status {enabled | disabled}
```

- k. Specify the stop on match parameter. **True** tells the sensor to stop processing filters if this item matches. **False** tells the sensor to continue processing filters even if this item matches.

```
sensor(config-eve-fil)# stop-on-match {true | false}
```

- l. Add any comments you want to use to explain this filter.

```
sensor(config-eve-fil)# user-comment NEW FILTER
```

Step 5 Verify the settings for the filter.

```
sensor(config-eve-fil)# show settings
```

```
NAME: name1
```

```
-----
signature-id-range: 1000-10005 default: 900-65535
subsignature-id-range: 1-5 default: 0-255
attacker-address-range: 192.0.2.3-192.0.2.26 default: 0.0.0.0-255.255.255.255
victim-address-range: 192.56.10.1-192.56.10.255 default: 0.0.0.0-255.255.255.255
  ipv6-attacker-address-range: 2001:0db8:3c4d:0015:0000:0000:abcd:ef12 default:
::0-FFFF:FFFF:FFFF:FFFF:FFFF:FFFF:FFFF:FFFF
  ipv6-victim-address-range: ::0-FFFF:FFFF:FFFF:FFFF:FFFF:FFFF:FFFF:FFFF default:
::0-FFFF:FFFF:FFFF:FFFF:FFFF:FFFF:FFFF:FFFF
attacker-port-range: 0-65535 <defaulted>
victim-port-range: 1-343 default: 0-65535
risk-rating-range: 85-100 default: 0-100
actions-to-remove: reset-tcp-connection default:
deny-attacker-percentage: 90 default: 100
filter-item-status: Enabled default: Enabled
stop-on-match: True default: False
user-comment: NEW FILTER default:
os-relevance: relevant default: relevant|not-relevant|unknown
-----
```

```
sensor(config-eve-fil)#
```

Step 6 Edit an existing filter.

```
sensor(config-eve)# filters edit name1
```

Step 7 Edit the parameters (see Steps 4a through 4l).

Step 8 Move a filter up or down in the filter list.

```
sensor(config-eve-fil)# exit
```

```
sensor(config-eve)# filters move name5 before name1
```

Step 9 Verify that you have moved the filters.

```

sensor(config-eve-fil)# exit
sensor(config-eve)# show settings
-----
filters (min: 0, max: 4096, current: 5 - 4 active, 1 inactive)
-----
ACTIVE list-contents
-----
NAME: name5
-----
signature-id-range: 900-65535 <defaulted>
subsignature-id-range: 0-255 <defaulted>
attacker-address-range: 0.0.0.0-255.255.255.255 <defaulted>
victim-address-range: 0.0.0.0-255.255.255.255 <defaulted>
attacker-port-range: 0-65535 <defaulted>
victim-port-range: 0-65535 <defaulted>
risk-rating-range: 0-100 <defaulted>
actions-to-remove: <defaulted>
filter-item-status: Enabled <defaulted>
stop-on-match: False <defaulted>
user-comment: <defaulted>
-----
NAME: name1
-----
signature-id-range: 900-65535 <defaulted>
subsignature-id-range: 0-255 <defaulted>
attacker-address-range: 0.0.0.0-255.255.255.255 <defaulted>
victim-address-range: 0.0.0.0-255.255.255.255 <defaulted>
attacker-port-range: 0-65535 <defaulted>
victim-port-range: 0-65535 <defaulted>
risk-rating-range: 0-100 <defaulted>
actions-to-remove: <defaulted>
filter-item-status: Enabled <defaulted>
stop-on-match: False <defaulted>
user-comment: <defaulted>
-----
NAME: name2
-----
signature-id-range: 900-65535 <defaulted>
subsignature-id-range: 0-255 <defaulted>
attacker-address-range: 0.0.0.0-255.255.255.255 <defaulted>
victim-address-range: 0.0.0.0-255.255.255.255 <defaulted>
attacker-port-range: 0-65535 <defaulted>
victim-port-range: 0-65535 <defaulted>
risk-rating-range: 0-100 <defaulted>
actions-to-remove: <defaulted>
filter-item-status: Enabled <defaulted>
stop-on-match: False <defaulted>
user-comment: <defaulted>
-----
-----
INACTIVE list-contents
-----
sensor(config-eve)#

```

Step 10 Move a filter to the inactive list.

```

sensor(config-eve)# filters move name1 inactive

```

Step 11 Verify that the filter has been moved to the inactive list.

```

sensor(config-eve-fil)# exit
sensor(config-eve)# show settings
-----
INACTIVE list-contents
-----
NAME: name1
-----
signature-id-range: 900-65535 <defaulted>
subsignature-id-range: 0-255 <defaulted>
attacker-address-range: 0.0.0.0-255.255.255.255 <defaulted>
victim-address-range: 0.0.0.0-255.255.255.255 <defaulted>
attacker-port-range: 0-65535 <defaulted>
victim-port-range: 0-65535 <defaulted>
risk-rating-range: 0-100 <defaulted>
actions-to-remove: <defaulted>
filter-item-status: Enabled <defaulted>
stop-on-match: False <defaulted>
user-comment: <defaulted>
-----
sensor(config-eve)#

```

Step 12 Exit event action rules submode.

```

sensor(config-eve)# exit
Apply Changes?[yes]:

```

Step 13 Press **Enter** to apply your changes or enter **no** to discard them.

For More Information

For the procedure for configuring event action variables, see [Adding, Editing, and Deleting Event Action Variables, page 8-11](#).

Configuring OS Identifications

This section describes OS identifications and how to configure OS maps, and contains the following topics:

- [Understanding Passive OS Fingerprinting, page 8-26](#)
- [Passive OS Fingerprinting Configuration Considerations, page 8-28](#)
- [Adding, Editing, Deleting, and Moving Configured OS Maps, page 8-28](#)
- [Displaying and Clearing OS Identifications, page 8-32](#)

Understanding Passive OS Fingerprinting

Passive OS fingerprinting lets the sensor determine the OS that hosts are running. The sensor analyzes network traffic between hosts and stores the OS of these hosts with their IP addresses. The sensor inspects TCP SYN and SYNACK packets exchanged on the network to determine the OS type.

The sensor then uses the OS of the target host OS to determine the relevance of the attack to the victim by computing the attack relevance rating component of the risk rating. Based on the relevance of the attack, the sensor may alter the risk rating of the alert for the attack and/or the sensor may filter the alert for the attack. You can then use the risk rating to reduce the number of false positive alerts (a benefit in IDS mode) or definitively drop suspicious packets (a benefit in IPS mode). Passive OS fingerprinting also enhances the alert output by reporting the victim OS, the source of the OS identification, and the relevance to the victim OS in the alert.

Passive OS fingerprinting consists of three components:

- **Passive OS learning**—Passive OS learning occurs as the sensor observes traffic on the network. Based on the characteristics of TCP SYN and SYNACK packets, the sensor makes a determination of the OS running on the host of the source IP address.
- **User-configurable OS identification**—You can configure OS host maps, which take precedence over learned OS maps.
- **Computation of attack relevance rating and risk rating**—The sensor uses OS information to determine the relevance of the attack signature to the targeted host. The attack relevance is the attack relevance rating component of the risk rating value for the attack alert. The sensor uses the OS type reported in the host posture information imported from the CSA MC to compute the attack relevance rating.

There are three sources of OS information. The sensor ranks the sources of OS information in the following order:

1. **Configured OS maps**—OS maps you enter. Configured OS maps reside in the event action rules policy and can apply to one or many virtual sensors.



Note You can specify multiple operating systems for the same IP address. The last one in the list is the operating system that is matched.

2. **Imported OS maps**—OS maps imported from an external data source. Imported OS maps are global and apply to all virtual sensors.



Note Currently the CSA MC is the only external data source.

3. **Learned OS maps**—OS maps observed by the sensor through the fingerprinting of TCP packets with the SYN control bit set. Learned OS maps are local to the virtual sensor that sees the traffic.

When the sensor needs to determine the OS for a target IP address, it consults the configured OS maps. If the target IP address is not in the configured OS maps, the sensor looks in the imported OS maps. If the target IP address is not in the imported OS maps, the sensor looks in the learned OS maps. If it cannot find it there, the sensor treats the OS of the target IP address as unknown.



Note Passive OS fingerprinting is enabled by default and the IPS contains a default vulnerable OS list for each signature.

Passive OS Fingerprinting Configuration Considerations

You do not have to configure passive OS fingerprinting for it to function. IPS provides a default vulnerable OS list for each signature and passive analysis is enabled by default.

You can configure the following aspects of passive OS fingerprinting:

- Define OS maps—We recommend configuring OS maps to define the identity of the OS running on critical systems. It is best to configure OS maps when the OS and IP address of the critical systems are unlikely to change.
- Limit the attack relevance rating calculation to a specific IP address range—This limits the attack relevance rating calculations to IP addresses on the protected network.
- Import OS maps—Importing OS maps provides a mechanism for accelerating the learning rate and fidelity of the OS identifications made through passive analysis. If you have an external product interface, such as the CSA MC, you can import OS identifications from it.
- Define event action rules filters using the OS relevance value of the target—This provides a way to filter alerts solely on OS relevance.
- Disable passive analysis—Stops the sensor from learning new OS maps.
- Edit signature vulnerable OS lists—The vulnerable OS list specifies what OS types are vulnerable to each signature. The default, `general-os`, applies to all signatures that do not specify a vulnerable OS list.

Adding, Editing, Deleting, and Moving Configured OS Maps

Use the `os-identifications` command in the service event action rules submode to configure OS host mappings, which take precedence over learned OS mappings. You can add, edit, and delete configured OS maps. You can move them up and down in the list to change the order in which the sensor computes the attack relevance rating and risk rating for that particular IP address and OS type combination.

You can also move them up and down in the list to change the order in which the sensor resolves the OS associated with a particular IP address. Configured OS mappings allow for ranges, so for network 192.168.1.0/24 an administrator might define the following (Table 8-1):

Table 8-1 Example Configured OS Mapping

IP Address Range Set	OS
192.168.1.1	IOS
192.168.1.2-192.168.1.10,192.168.1.25	UNIX
192.168.1.1-192.168.1.255	Windows

More specific mappings should be at the beginning of the list. Overlap in the IP address range sets is allowed, but the entry closest to the beginning of the list takes precedence.

The following commands apply:

- **calc-arr-for-ip-range**—Calculates the attack relevance rating for victims in this range. The value is <A.B.C.D>-<A.B.C.D>[,<A.B.C.D>-<A.B.C.D>], for example, 10.20.1.0-10.20.1.255,10.20.5.0-10.20.5.255).



Note The second IP address in the range must be greater than or equal to the first IP address.

- **configured-os-map {edit | insert | move} name1[begin | end | inactive | before | after]**—Specifies a collection of administrator-defined mappings of IP addresses to OS IDs (configured OS mappings take precedence over imported and learned OS mappings).
- **ip**—Specifies the host IP address (or addresses) running the specified OS. The value is <A.B.C.D>-<A.B.C.D>[,<A.B.C.D>-<A.B.C.D>], for example, 10.20.1.0-10.20.1.255,10.20.5.0-10.20.5.255.



Note The second IP address in the range must be greater than or equal to the first IP address.

- **os**—Specifies the OS type the host (or hosts) is running:
 - **general-os**—All OS types
 - **ios**—Variants of Cisco IOS
 - **mac-os**—Variants of the Apple System OS prior to OS X
 - **netware**—Netware
 - **other**—Any Other OS
 - **unix**—Variants of UNIX
 - **aix**—Variants of AIX
 - **bsd**—Variants of BSD
 - **hp-ux**—Variants of HP-UX
 - **irix**—Variants of IRIX
 - **linux**—Variants of Linux
 - **solaris**—Variants of Solaris
 - **windows**—Variants of Microsoft Windows
 - **windows-nt-2k-xp**—Variants of NT, 2000, and XP
 - **win-nt**—Specific variants of Windows NT
 - **unknown**—Unknown OS
- **default**—Sets the value back to the system default setting.
- **no**—Removes an entry or selection setting.
- **passive-traffic-analysis {enabled | disabled}**—Enables/disables passive OS fingerprinting analysis.

Configuring OS Maps

To configure OS maps, follow these steps:

Step 1 Log in to the CLI using an account with administrator privileges.

Step 2 Enter event action rules submode.

```
sensor# configure terminal
sensor(config)# service event-action-rules rules1
sensor(config-eve)#
```

Step 3 Create the OS map. Use **name1**, **name2**, and so forth to name your OS maps. Use the **begin | end | inactive | before | after** keywords to specify where you want to insert the filter.

```
sensor(config-eve)# os-identification
sensor(config-eve-os)# configured-os-map insert name1 begin
sensor(config-eve-os-con)#
```

Step 4 Specify the values for this OS map:

a. Specify the host IP address.

```
sensor(config-eve-os-con)# ip 192.0.2.0-192.0.2.255
```

b. Specify the host OS type.

```
sensor(config-eve-os-con)# os unix
```



Caution

You can specify multiple operating systems for the same IP address. The last one in the list is the operating system that is matched.

Step 5 Verify the settings for the OS map.

```
sensor(config-eve-os-con)# show settings
NAME: name1
-----
ip: 192.0.2.0-192.0.2.255 default:
os: unix
-----
sensor(config-eve-os-con)#
```

Step 6 Specify the attack relevance rating range for the IP address.

```
sensor(config-eve-os-con)# exit
sensor(config-eve-os)# calc-arr-for-ip-range 192.0.2.1 to 192.0.2.25
```

Step 7 Enable passive OS fingerprinting.

```
sensor(config-eve-os)# passive-traffic-analysis enabled
```

Step 8 Edit an existing OS map.

```
sensor(config-eve-os)# configured-os-map edit name1
sensor(config-eve-os-con)#
```

Step 9 Edit the parameters (see Steps 4 through 7).

Step 10 Move an OS map up or down in the OS maps list.

```
sensor(config-eve-os-con)# exit
sensor(config-eve-os)# configured-os-map move name5 before name1
```

Step 11 Verify that you have moved the OS maps.

```

sensor(config-eve-os)# show settings
os-identification
-----
calc-arr-for-ip-range: 192.0.2.1-192.0.2.25 default: 0.0.0.0-255.255.255.255
configured-os-map (ordered min: 0, max: 50, current: 2 - 2 active, 0 inactive)
-----
ACTIVE list-contents
-----
NAME: name2
-----
ip: 192.0.2.33 default:
os: aix
-----
NAME: name1
-----
ip: 192.0.2.0-192.0.2.255 default:
os: unix
-----
passive-traffic-analysis: Enabled default: Enabled
-----
ips-ssp(config-eve-os)#

```

Step 12 Move an OS map to the inactive list.

```

sensor(config-eve-os)# configured-os-map move name1 inactive

```

Step 13 Verify that the filter has been moved to the inactive list.

```

sensor(config-eve-os)# show settings
os-identification
-----
calc-arr-for-ip-range: 192.0.2.33 default: 0.0.0.0-255.255.255.255
configured-os-map (ordered min: 0, max: 50, current: 2 - 1 active, 1 inactive)
-----
ACTIVE list-contents
-----
NAME: name2
-----
ip: 192.0.2.33 default:
os: aix
-----
INACTIVE list-contents
-----
NAME: name1
-----
ip: 192.0.2.0-192.0.2.255 default:
os: unix
-----
passive-traffic-analysis: Enabled default: Enabled
--MORE--#

```

Step 14 Delete an OS map.

```

sensor(config-eve-os)# no configured-os-map name2

```

Step 15 Verify that the OS map has been deleted.

```

sensor(config-eve-os)# show settings
os-identification
-----
calc-arr-for-ip-range: 192.0.2.33 default: 0.0.0.0-255.255.255.255
configured-os-map (ordered min: 0, max: 50, current: 1 - 0 active, 1 inactive)
-----
INACTIVE list-contents
-----
NAME: name1
-----
ip: 192.0.2.0-192.0.2.255 default:
os: unix
-----
-----
passive-traffic-analysis: Enabled default: Enabled
-----
ips-ssp(config-eve-os)#

```

Step 16 Exit event action rules submode.

```

sensor(config-eve-os)# exit
sensor(config-eve)# exit
Apply Changes?[yes]:

```

Step 17 Press **Enter** to apply your changes or enter **no** to discard them.

Displaying and Clearing OS Identifications

Use the **show os-identification** [*virtual-sensor*] **learned** [*ip-address*] command in EXEC mode to display OS IDs associated with IP addresses that were learned by the sensor through passive analysis.

Use the **clear os-identification** [*virtual-sensor*] **learned** [*ip-address*] command in EXEC mode to delete OS IDs associated with IP addresses that were learned by the sensor through passive analysis.

When you specify an IP address, only the OS identification for the specified IP address is displayed or cleared. If you specify a virtual sensor, only the OS identifications for the specified sensor is displayed or cleared. If you specify an IP address without a virtual sensor, the IP address is displayed or cleared on all virtual sensors.

The following commands apply:

- *virtual-sensor*—(Optional) Specifies the learned addresses of the virtual sensor that should be displayed or cleared.
- *ip-address*—(Optional) Specifies the IP address to query or clear. The sensor displays or clears the OS ID mapped to the specified IP address.

Displaying and Clearing OS Identifications

To display and clear OS IDs, follow these steps:

Step 1 Log in to the CLI using an account with administrator or operator privileges.



Note An account with viewer privileges can display OS IDs.

Step 2 Display the learned OS IDs associated with a specific IP address.

```
sensor# show os-identification learned 192.0.2.0
Virtual Sensor vs0:
  10.1.1.12 windows
sensor# show os-identification learned
Virtual Sensor vs0:
  10.1.1.12 windows
Virtual Sensor vs1:
  10.1.0.1  unix
  10.1.0.2  windows
  10.1.0.3  windows
sensor#
```

Step 3 Clear the learned OS IDs for a specific IP address on all virtual sensors.

```
sensor# clear os-identification learned 192.0.2.0
```

Step 4 Verify that the OS IDs have been cleared.

```
sensor# show statistics os-identification
Statistics for Virtual Sensor vs0
  OS Identification
    Configured
    Imported
    Learned
Statistics for Virtual Sensor vs1
  OS Identification
    Configured
    Imported
    Learned
sensor#
```

Configuring General Settings

This section describes the general settings, and contains the following topics:

- [Understanding Event Action Summarization, page 8-33](#)
- [Understanding Event Action Aggregation, page 8-34](#)
- [Configuring the General Settings, page 8-34](#)

Understanding Event Action Summarization

Summarization decreases the volume of alerts sent out from the sensor by providing basic aggregation of events into a single alert. Special parameters are specified for each signature and they influence the handling of the alerts. Each signature is created with defaults that reflect a preferred normal behavior. However, you can tune each signature to change this default behavior within the constraints for each engine type.

The nonalert-generating actions (deny, block, TCP reset) go through the filters for each signature event unsummarized. The alert-generating actions are not performed on these summarized alerts; instead the actions are applied to the one summary alert and then put through the filters.

If you select one of the other alert-generating actions and do not have it filtered out, the alert is created even if you do not select produce-alert. To prevent alerts from being created, you must have all alert-generating actions filtered out.

Summarization and event actions are processed after the Meta engine has processed the component events. This lets the sensor watch for suspicious activity transpiring over a series of events.

Understanding Event Action Aggregation

Basic aggregation provides two operating modes. The simple mode involves configuring a threshold number of hits for a signature that must be met before the alert is sent. A more advanced mode is timed-interval counting. In this mode, the sensor tracks the number of hits per second and only sends alerts when that threshold is met. In this example, a hit is a term used to describe an event, which is basically an alert, but it is not sent out of the sensor as an alert until the threshold number of hits has been exceeded.

You can choose from the following summarization options:

- **fire-all**—Fires an alert each time the signature is triggered. If the threshold is set for summarization, alerts are fired for each execution until summarization occurs. After summarization starts, only one alert every summary interval fires for each address set. Alerts for other address sets are either all seen or separately summarized. The signature reverts to fire all mode after a period of no alerts for that signature.
- **summary**—Fires an alert the first time a signature is triggered, and then additional alerts for that signature are summarized for the duration of the summary interval. Only one alert every summary interval should fire for each address set. If the global summary threshold is reached, the signature goes into global summarization mode.
- **global-summarization**—Fires an alert for every summary interval. Signatures can be preconfigured for global summarization.
- **fire-once**—Fires an alert for each address set. You can upgrade this mode to global summarization mode.

Configuring the General Settings

Use the following commands in service event action rules submode to configure general event action rules settings:

- **global-block-timeout** —Specifies the number of minutes to block a host or connection. The valid range is 0 to 10000000. The default is 30 minutes.
- **global-deny-timeout**—Specifies the number of seconds to deny attackers inline. The valid range is 0 to 518400. The default is 3600.
- **global-filters-status {enabled | disabled}**—Enables or disables the use of the filters. The default is enabled.
- **global-metaevent-status {enabled | disabled}**—Enables or disables the use of the Meta Event Generator. The default is enabled.
- **global-overrides-status {enabled | disabled}**—Enables or disables the use of the overrides. The default is enabled.

- **global-summarization-status {enabled | disabled}**—Enables or disables the use of the summarizer. The default is enabled.
- **max-denied-attackers**—Limits the number of denied attackers possible in the system at any one time. The valid range is 0 to 100000000. The default is 10000.

Configuring Event Action General Settings

To configure event action general settings, follow these steps:

-
- Step 1** Log in to the CLI using an account with administrator privileges.
- Step 2** Enter event action rules submode.
- ```
sensor# configure terminal
sensor(config)# service event-action-rules rules0
```
- Step 3** Enter general submode.
- ```
sensor(config)# general
```
- Step 4** Enable or disable the meta event generator. The default is enabled.
- ```
sensor(config-eve-gen)# global-metaevent-status {enabled | disabled}
```
- Step 5** Enable or disable the summarizer. The default is enabled.
- ```
sensor(config-eve-gen)# global-summarization-status {enabled | disabled}
```
- Step 6** Configure the denied attackers inline event action:
- Limit the number of denied attackers in the system at any given time. The default is 1000.


```
sensor(config-eve-gen)# max-denied-attackers 100
```
 - Configure the amount of seconds to deny attackers in the system. The default is 3600 seconds.


```
sensor(config-eve-gen)# global-deny-timeout 1000
```
- Step 7** Configure the number of minutes to block a host or a connection. The default is 30 minutes.
- ```
sensor(config-eve-gen)# global-block-timeout 20
```
- Step 8** Enable or disable any overrides that you have set up. The default is enabled.
- ```
sensor(config-eve-gen)# global-overrides-status {enabled | disabled}
```
- Step 9** Enable or disable any filters that you have set up. The default is enabled.
- ```
sensor(config-eve-gen)# global-filters-status {enabled | disabled}
```
- Step 10** Verify the settings for general submode.
- ```
sensor(config-eve-gen)# show settings
general
-----
global-overrides-status: Enabled default: Enabled
global-filters-status: Enabled default: Enabled
global-summarization-status: Enabled default: Enabled
global-metaevent-status: Enabled default: Enabled
global-deny-timeout: 1000 default: 3600
global-block-timeout: 20 default: 30
max-denied-attackers: 100 default: 10000
-----
sensor(config-eve-gen)#
```

Step 11 Exit event action rules submode.

```
sensor(config-eve-gen)# exit
sensor(config-eve)# exit
Apply Changes?[yes]:
```

Step 12 Press **Enter** to apply your changes or enter **no** to discard them.

Configuring the Denied Attackers List

This section describes the denied attackers list and how to add, clear, and monitor the list. It contains the following topics:

- [Adding a Deny Attacker Entry to the Denied Attackers List, page 8-36](#)
- [Monitoring and Clearing the Denied Attackers List, page 8-37](#)

Adding a Deny Attacker Entry to the Denied Attackers List

Use the **deny attacker** [*virtual-sensor name*] [*ip-address attacker-ip-address*] | **victim** [*victim-ip-address*] | **port** [*port-number*] command to add a single deny attacker entry to the list of denied attackers. Use the **no** form of the command to delete the deny attacker entry from the list.

The following commands apply:

- *name*—(Optional) Specifies the name of the virtual sensor to which the deny attackers entry should be added.
- *attacker-ip-address*—Specifies the attacker IP address.
- *victim-ip-address*—(Optional) Specifies the victim IP address.
- *port-number*—(Optional) Specifies the victim port number. The valid range is 0 to 65535.

Adding Entries to the Denied Attacker List

To add a deny attacker entry to the list of denied attackers, follow these steps:

Step 1 Log in to the CLI using an account with administrator or operator privileges.

Step 2 Add a deny attacker entry with an IP address of 192.0.2.0.

```
sensor# deny attacker ip-address 192.0.2.0
Warning: Executing this command will add deny attacker address on all virtual sensors.
Continue? [yes]:
```

Step 3 Enter **yes** to add this deny attacker entry for all virtual sensors.

Step 4 Add a deny attacker entry to a specific virtual sensor.

```
sensor# deny attacker virtual-sensor vs0 ip-address 192.0.2.0
```

Step 5 Remove the deny attacker entry from the list.

```
sensor# no deny attacker ip-address 10.1.1.1
Warning: Executing this command will delete this address from the list of attackers being
denied by all virtual sensors.
Continue? [yes]:
```

Step 6 Enter **yes** to remove the deny attacker entry from the list.



Note To immediately stop denying attackers, you must use the **clear denied-attackers** command to clear the denied attackers list.

For More Information

For the procedure for clearing denied attackers permanently from the denied attackers list, see [Monitoring and Clearing the Denied Attackers List, page 8-37](#).

Monitoring and Clearing the Denied Attackers List

Use the **show statistics denied-attackers** command to display the list of denied attackers. Use the **clear denied-attackers** [*virtual_sensor*] [*ip-address ip_address*] command to delete the denied attackers list and clear the virtual sensor statistics.

If your sensor is configured to operate in inline mode, the traffic is passing through the sensor. You can configure signatures to deny packets, connections, and attackers while in inline mode, which means that single packets, connections, and specific attackers are denied, that is, not transmitted, when the sensor encounters them. When the signature fires, the attacker is denied and placed in a list. As part of sensor administration, you may want to delete the list or clear the statistics in the list.

The following commands apply:

- *virtual_sensor*—(Optional) Specifies the virtual sensor whose denied attackers list should be cleared.
- *ip_address*—(Optional) Specifies the IP address to clear.

Displaying and Deleting Denied Attackers

To display the list of denied attackers and delete the list and clear the statistics, follow these steps:

Step 1 Log in to the CLI using an account with administrator privileges.

Step 2 Display the list of denied IP addresses. The statistics show that there are two IP addresses being denied at this time.

```
sensor# show statistics denied-attackers
Denied Attackers and hit count for each.
 10.20.4.2 = 9
 10.20.5.2 = 5
```

Step 3 Delete the denied attackers list.

```
sensor# clear denied-attackers
Warning: Executing this command will delete all addresses from the list of attackers
currently being denied by the sensor.
Continue with clear? [yes]:
```

Step 4 Enter **yes** to clear the list.

Step 5 Delete the denied attackers list for a specific virtual sensor.

```
sensor# clear denied-attackers vs0
Warning: Executing this command will delete all addresses from the list of attackers being
denied by virtual sensor vs0.
```

Continue with clear? [yes]:

Step 6 Enter **yes** to clear the list.

Step 7 Remove a specific IP address from the denied attackers list for a specific virtual sensor.

```
sensor# clear denied-attackers vs0 ip-address 192.0.2.0
Warning: Executing this command will delete ip address 192.0.2.0 from the list of
attackers being denied by virtual sensor vs0.
Continue with clear? [yes]:
```

Step 8 Enter **yes** to clear the list.

Step 9 Verify that you have cleared the list. You can use the **show statistics denied-attackers** or **show statistics virtual-sensor** command.

```
sensor# show statistics denied-attackers
Denied Attackers and hit count for each.
Denied Attackers and hit count for each.
Statistics for Virtual Sensor vs0
  Denied Attackers with percent denied and hit count for each.
```

```
  Denied Attackers with percent denied and hit count for each.
```

```
Statistics for Virtual Sensor vs1
  Denied Attackers with percent denied and hit count for each.
```

```
  Denied Attackers with percent denied and hit count for each.
sensor#
```

```
sensor# show statistics virtual-sensor
Virtual Sensor Statistics
  Statistics for Virtual Sensor vs0
    Name of current Signature-Definition instance = sig0
    Name of current Event-Action-Rules instance = rules0
    List of interfaces monitored by this virtual sensor = mypair
  Denied Address Information
    Number of Active Denied Attackers = 0
    Number of Denied Attackers Inserted = 2
    Number of Denied Attackers Total Hits = 287
    Number of times max-denied-attackers limited creation of new entry = 0
    Number of exec Clear commands during uptime = 1
  Denied Attackers and hit count for each.
```

Step 10 Clear only the statistics.

```
sensor# show statistics virtual-sensor clear
```

Step 11 Verify that you have cleared the statistics. The statistics have all been cleared except for the **Number of Active Denied Attackers** and **Number of exec Clear commands during uptime** categories. It is important to know if the list has been cleared.

```
sensor# show statistics virtual-sensor
Virtual Sensor Statistics
  Statistics for Virtual Sensor vs0
    Name of current Signature-Definition instance = sig0
    Name of current Event-Action-Rules instance = rules0
    List of interfaces monitored by this virtual sensor = mypair
  Denied Address Information
    Number of Active Denied Attackers = 2
    Number of Denied Attackers Inserted = 0
    Number of Denied Attackers Total Hits = 0
```

```

Number of times max-denied-attackers limited creation of new entry = 0
Number of exec Clear commands during uptime = 1
Denied Attackers and hit count for each.
10.20.2.5 = 0
10.20.5.2 = 0

```

Monitoring Events

This section describes how to display and clear events from the Event Store, and contains the following topics:

- [Displaying Events, page 8-39](#)
- [Clearing Events from Event Store, page 8-42](#)

Displaying Events



Note

The Event Store has a fixed size of 30 MB for all platforms.



Note

Events are displayed as a live feed. To cancel the request, press **Ctrl-C**.

Use the **show events** [**alert** [informational] [low] [medium] [high] [**include-traits** *traits*] [**exclude-traits** *traits*] [**min-threat-rating** *min-rr*] [**max-threat-rating** *max-rr*] | **error** [warning] [error] [fatal] | **NAC** | **status**]] [*hh:mm:ss* [*month day* [*year*]] | **past** *hh:mm:ss*] command to display events from Event Store. Events are displayed beginning at the start time. If you do not specify a start time, events are displayed beginning at the current time. If you do not specify an event type, all events are displayed.

The following commands apply:

- **alert**—Displays alerts. Provides notification of some suspicious activity that may indicate an attack is in process or has been attempted. Alert events are generated by the Analysis Engine whenever a signature is triggered by network activity. If no level is selected (informational, low, medium, or high), all alert events are displayed.
- **include-traits**—Displays alerts that have the specified traits.
- **exclude-traits**—Does not display alerts that have the specified traits.
- **traits**—Specifies the trait bit position in decimal (0 to 15).
- **min-threat-rating**—Displays events with a threat rating above or equal to this value. The default is 0. The valid range is 0 to 100.
- **max-threat-rating**—Displays events with a threat rating below or equal to this value. The default is 100. The valid range is 0 to 100.
- **error**—Displays error events. Error events are generated by services when error conditions are encountered. If no level is selected (warning, error, or fatal), all error events are displayed.
- **NAC**—Displays the ARC (block) requests.



Note The ARC is formerly known as NAC. This name change has not been completely implemented throughout the IDM, the IME, and the CLI.

- **status**—Displays status events.
- **past**—Displays events starting in the past for the specified hours, minutes, and seconds.
- *hh:mm:ss*—Specifies the hours, minutes, and seconds in the past to begin the display.

**Note**

The **show events** command continues to display events until a specified event is available. To exit, press **Ctrl-C**.

Displaying Events

To display events from the Event Store, follow these steps:

Step 1 Log in to the CLI.

Step 2 Display all events starting now. The feed continues showing all events until you press **Ctrl-C**.

```
sensor# show events
evError: eventId=1041472274774840147 severity=warning vendor=Cisco
  originator:
    hostId: sensor2
    appName: cidwebserver
    appInstanceId: 12075
  time: 2011/01/07 04:41:45 2011/01/07 04:41:45 UTC
  errorMessage: name=errWarning received fatal alert: certificate_unknown

evError: eventId=1041472274774840148 severity=error vendor=Cisco
  originator:
    hostId: sensor2
    appName: cidwebserver
    appInstanceId: 351
  time: 2011/01/07 04:41:45 2011/01/07 04:41:45 UTC
  errorMessage: name=errTransport WebSession::sessionTask(6) TLS connection exception: handshake incomplete.
```

Step 3 Display the block requests beginning at 10:00 a.m. on February 9, 2011.

```
sensor# show events NAC 10:00:00 Feb 9 2011
evShunRqst: eventId=1106837332219222281 vendor=Cisco
  originator:
    deviceName: Sensor1
    appName: NetworkAccessControllerApp
    appInstance: 654
  time: 2011/02/09 10:33:31 2011/08/09 13:13:31
  shunInfo:
    host: connectionShun=false
    srcAddr: 11.0.0.1
    destAddr:
    srcPort:
    destPort:
    protocol: numericType=0 other
    timeoutMinutes: 40
  evAlertRef: hostId=esendHost 123456789012345678
sensor#
```

Step 4 Display errors with the warning level starting at 10:00 a.m. on February 9, 2011.

```

sensor# show events error warning 10:00:00 Feb 9 2011
evError: eventId=1041472274774840197 severity=warning vendor=Cisco
  originator:
    hostId: sensor
    appName: cidwebserver
    appInstanceId: 12160
  time: 2011/01/07 04:49:25 2011/01/07 04:49:25 UTC
  errorMessage: name=errWarning received fatal alert: certificate_unknown

```

Step 5 Display alerts from the past 45 seconds.

```

sensor# show events alert past 00:00:45

evIdsAlert: eventId=1109695939102805307 severity=medium vendor=Cisco
  originator:
    hostId: sensor
    appName: sensorApp
    appInstanceId: 367
  time: 2011/03/02 14:15:59 2011/03/02 14:15:59 UTC
  signature: description=Nachi Worm ICMP Echo Request id=2156 version=S54
  subsigId: 0
  sigDetails: Nachi ICMP
  interfaceGroup:
  vlan: 0
  participants:
    attacker:
      addr: locality=OUT 10.89.228.202
    target:
      addr: locality=OUT 10.89.150.185
  riskRatingValue: 70
  interface: fe0_1
  protocol: icmp

evIdsAlert: eventId=1109695939102805308 severity=medium vendor=Cisco
  originator:
  --MORE--

```

Step 6 Display events that began 30 seconds in the past.

```

sensor# show events past 00:00:30
evStatus: eventId=1041526834774829055 vendor=Cisco
  originator:
    hostId: sensor
    appName: mainApp
    appInstanceId: 2215
  time: 2011/01/08 02:41:00 2011/01/08 02:41:00 UTC
  controlTransaction: command=getVersion successful=true
  description: Control transaction response.
  requestor:
    user: cids
    application:
      hostId: 64.101.182.101
      appName: -cidcli
      appInstanceId: 2316

evStatus: eventId=1041526834774829056 vendor=Cisco
  originator:
    hostId: sensor
    appName: login(pam_unix)
    appInstanceId: 2315
  time: 2011/01/08 02:41:00 2011/01/08 02:41:00 UTC

```

```
syslogMessage:  
  description: session opened for user cisco by cisco(uid=0)
```

Clearing Events from Event Store

Use the **clear events** command to clear the Event Store.

To clear events from the Event Store, follow these steps:

Step 1 Log in to the CLI using an account with administrator privileges.

Step 2 Clear the Event Store.

```
sensor# clear events  
Warning: Executing this command will remove all events currently stored in the event  
store.  
Continue with clear? []:
```

Step 3 Enter **yes** to clear the events.



Configuring Anomaly Detection

This chapter describes anomaly detection (AD) and its features and how to configure them. This chapter contains the following topics:

- [Anomaly Detection Notes and Caveats, page 9-1](#)
- [Understanding Security Policies, page 9-2](#)
- [Understanding Anomaly Detection, page 9-2](#)
- [Understanding Worms, page 9-2](#)
- [Anomaly Detection Modes, page 9-3](#)
- [Anomaly Detection Zones, page 9-4](#)
- [Anomaly Detection Configuration Sequence, page 9-5](#)
- [Anomaly Detection Signatures, page 9-6](#)
- [Enabling Anomaly Detection, page 9-8](#)
- [Working With Anomaly Detection Policies, page 9-8](#)
- [Configuring Anomaly Detection Operational Settings, page 9-10](#)
- [Configuring the Internal Zone, page 9-11](#)
- [Configuring the Illegal Zone, page 9-20](#)
- [Configuring the External Zone, page 9-28](#)
- [Configuring Learning Accept Mode, page 9-36](#)
- [Working With KB Files, page 9-40](#)
- [Displaying Anomaly Detection Statistics, page 9-47](#)
- [Disabling Anomaly Detection, page 9-48](#)

Anomaly Detection Notes and Caveats

The following notes and caveats apply to configuring anomaly detection:

- Anomaly detection is disabled by default. You must enable it to configure or apply an anomaly detection policy. Enabling anomaly detection results in a decrease in performance.
- Anomaly detection assumes it gets traffic from both directions. If the sensor is configured to see only one direction of traffic, you should turn off anomaly detection. Otherwise, when anomaly detection is running in an asymmetric environment, it identifies all traffic as having incomplete

connections, that is, as scanners, and sends alerts for all traffic flows. Using asymmetric mode protection with anomaly detection enabled causes excessive resource usage and possible false positives for anomaly detection signatures.

Understanding Security Policies

You can create multiple security policies and apply them to individual virtual sensors. A security policy is made up of a signature definition policy, an event action rules policy, and an anomaly detection policy. Cisco IPS contains a default signature definition policy called sig0, a default event action rules policy called rules0, and a default anomaly detection policy called ad0. You can assign the default policies to a virtual sensor or you can create new policies. The use of multiple security policies lets you create security policies based on different requirements and then apply these customized policies per VLAN or physical interface.

Understanding Anomaly Detection

The anomaly detection component of the sensor detects worm-infected hosts. This enables the sensor to be less dependent on signature updates for protection against worms and scanners, such as Code Red and SQL Slammer and so forth. The anomaly detection component lets the sensor learn normal activity and send alerts or take dynamic response actions for behavior that deviates from what it has learned as normal behavior.

**Note**

Anomaly detection does not detect email-based worms, such as Nimda.

Anomaly detection detects the following two situations:

- When the network starts on the path of becoming congested by worm traffic.
- When a single worm-infected source enters the network and starts scanning for other vulnerable hosts.

Understanding Worms

**Caution**

Anomaly detection assumes it gets traffic from both directions. If the sensor is configured to see only one direction of traffic, you should turn off anomaly detection. Otherwise, when anomaly detection is running in an asymmetric environment, it identifies all traffic as having incomplete connections, that is, as scanners, and sends alerts for all traffic flows. Using asymmetric mode protection with anomaly detection enabled causes excessive resource usage and possible false positives for anomaly detection signatures.

Worms are automated, self-propagating, intrusion agents that make copies of themselves and then facilitate their spread. Worms attack a vulnerable host, infect it, and then use it as a base to attack other vulnerable hosts. They search for other hosts by using a form of network inspection, typically a scan, and then propagate to the next target. A scanning worm locates vulnerable hosts by generating a list of IP addresses to probe, and then contacts the hosts. Code Red worm, Sasser worm, Blaster worm, and the Slammer worm are examples of worms that spread in this manner.

Anomaly detection identifies worm-infected hosts by their behavior as scanners. To spread, a worm must find new hosts. It finds them by scanning the Internet or network using TCP, UDP, and other protocols to generate unsuccessful attempts to access different destination IP addresses. A scanner is defined as a source IP address that generates events on the same destination port (in TCP and UDP) for too many destination IP addresses.

The events that are important for TCP protocol are nonestablished connections, such as a SYN packet that does not have its SYN-ACK response for a given amount of time. A worm-infected host that scans using TCP protocol generates nonestablished connections on the same destination port for an anomalous number of IP addresses.

The events that are important for UDP protocol are unidirectional connections, such as a UDP connection where all packets are going only in one direction. A worm-infected host that scans using UDP protocol generates UDP packets but does not receive UDP packets on the same quad within a timeout period on the same destination port for multiple destination IP addresses.

The events that are important for other protocols, such as ICMP, are from a source IP address to many different destination IP addresses, that is, packets that are received in only one direction.

**Caution**

If a worm has a list of IP addresses it should infect and does not have to use scanning to spread itself (for example, it uses passive mapping—listening to the network as opposed to active scanning), it is not detected by the anomaly detection worm policies. Worms that receive a mailing list from probing files within the infected host and email this list are also not detected, because no Layer 3/Layer 4 anomaly is generated.

For More Information

For the procedure for turning off anomaly detection, see [Disabling Anomaly Detection, page 9-48](#).

Anomaly Detection Modes

If you have anomaly detection enabled, it initially conducts a “peacetime” learning process when the most normal state of the network is reflected. Anomaly detection then derives a set of policy thresholds that best fit the normal network.

Anomaly detection has the following modes:

- Learning accept mode—Anomaly detection conducts an initial learning accept mode for the default period of 24 hours. We assume that during this phase no attack is being carried out. Anomaly detection creates an initial baseline, known as a knowledge base (KB), of the network traffic. The default interval value for periodic schedule is 24 hours and the default action is rotate, meaning that a new KB is saved and loaded, and then replaces the initial KB after 24 hours.

**Note**

Anomaly detection does not detect attacks when working with the initial KB, which is empty. After the default of 24 hours, a KB is saved and loaded and now anomaly detection also detects attacks.

**Note**

Depending on your network complexity, you may want to have anomaly detection in learning accept mode for longer than the default 24 hours.

- **Detect mode**—For ongoing operation, the sensor should remain in detect mode. This is for 24 hours a day, 7 days a week. Once a KB is created and replaces the initial KB, anomaly detection detects attacks based on it. It looks at the network traffic flows that violate thresholds in the KB and sends alerts. As anomaly detection looks for anomalies, it also records gradual changes to the KB that do not violate the thresholds and thus creates a new KB. The new KB is periodically saved and takes the place of the old one thus maintaining an up-to-date KB.
- **Inactive mode**—You can turn anomaly detection off by putting it in inactive mode. Under certain circumstances, anomaly detection should be in inactive mode, for example, if the sensor is running in an asymmetric environment. Because anomaly detection assumes it gets traffic from both directions, if the sensor is configured to see only one direction of traffic, anomaly detection identifies all traffic as having incomplete connections, that is, as scanners, and sends alerts for all traffic flows. Having anomaly detection running also lowers performance.

Example

The following example summarizes the default anomaly detection configuration. If you add a virtual sensor at 11:00 pm and do not change the default anomaly detection configuration, anomaly detection begins working with the initial KB and only performs learning. Although it is in detect mode, it cannot detect attacks until it has gathered information for 24 hours and replaced the initial KB. At the first start time (10:00 am by default), and the first interval (24 hours by default), the learning results are saved to a new KB and this KB is loaded and replaces the initial KB. Because the anomaly detection is in detect mode by default, now that anomaly detection has a new KB, the anomaly detection begins to detect attacks.

For More Information

- For the procedures for putting anomaly detection in different modes, see [Adding, Editing, and Deleting Virtual Sensors, page 6-5](#).
- For more information about how worms operate, see [Understanding Worms, page 9-2](#).

Anomaly Detection Zones

By subdividing the network into zones, you can achieve a lower false negative rate. A zone is a set of destination IP addresses. There are three zones, internal, illegal, and external, each with its own thresholds.

The external zone is the default zone with the default Internet range of 0.0.0.0-255.255.255.255. By default, the internal and illegal zones contain no IP addresses. Packets that do not match the set of IP addresses in the internal or illegal zone are handled by the external zone.

We recommend that you configure the internal zone with the IP address range of your internal network. If you configure it in this way, the internal zone is all the traffic that comes to your IP address range, and the external zone is all the traffic that goes to the Internet.

You can configure the illegal zone with IP address ranges that should never be seen in normal traffic, for example, unallocated IP addresses or part of your internal IP address range that is unoccupied. An illegal zone can be very helpful for accurate detection, because we do not expect any legal traffic to reach this zone. This allows very low thresholds, which in turn can lead to very quick worm virus detection.

For More Information

For the procedures for configuring zones, see [Configuring the Internal Zone, page 9-11](#), [Configuring the Illegal Zone, page 9-20](#), and [Configuring the External Zone, page 9-28](#).

Anomaly Detection Configuration Sequence

You can configure the detection part of anomaly detection. You can configure a set of thresholds that override the KB learned thresholds. However, anomaly detection continues learning regardless of how you configure the detection. You can also import, export, and load a KB and you can view a KB for data.

Follow this sequence when configuring anomaly detection:

1. Create an anomaly detection policy to add to the virtual sensors. Or you can use the default anomaly detection policy, `ad0`.
2. Add the anomaly detection policy to your virtual sensors.
3. Enable anomaly detection.
4. Configure the anomaly detection zones and protocols.
5. For the first 24 hours anomaly detection performs learning to create a populated KB. The initial KB is empty and during the default 24 hours, anomaly detection collects data to use to populate the KB. If you want the learning period to be longer than the default period of 24 hours, you must manually set the mode to learning accept.
6. Let the sensor run in learning accept mode for at least 24 hours (the default). You should let the sensor run in learning accept mode for at least 24 hours so it can gather information on the normal state of the network for the initial KB. However, you should change the amount of time for learning accept mode according to the complexity of your network. After the time period, the sensor saves the initial KB as a baseline of the normal activity of your network.



Note We recommend leaving the sensor in learning accept mode for at least 24 hours, but letting the sensor run in learning accept mode for longer, even up to a week, is better.

7. If you manually set anomaly detection to learning accept mode, switch back to detect mode.
8. Configure the anomaly detection parameters:
 - Configure the worm timeout and which source and destination IP addresses should be bypassed by anomaly detection. After this timeout, the scanner threshold returns to the configured value.
 - Decide whether you want to enable automatic KB updates when anomaly detection is in detect mode.
 - Configure the 18 anomaly detection worm signatures to have more event actions than just the default produce-alert. For example, configure them to have deny-attacker event actions.

For More Information

- For the procedures for putting anomaly detection in different modes, see [Adding, Editing, and Deleting Virtual Sensors, page 6-5](#).
- For the procedure for configuring a new anomaly detection policy, see [Working With Anomaly Detection Policies, page 9-8](#).
- For more information on configuring zones, see [Configuring the Internal Zone, page 9-11](#), [Configuring the Illegal Zone, page 9-20](#), and [Configuring the External Zone, page 9-28](#).
- For more information on anomaly detection modes, see [Anomaly Detection Modes, page 9-3](#).
- For more information about configuring learning accept mode, see [Configuring Learning Accept Mode, page 9-36](#).

- For more information on configuring anomaly detection signatures, see [Anomaly Detection Signatures, page 9-6](#).
- For more information on Deny Attacker event actions, see [Event Actions, page 8-5](#).

Anomaly Detection Signatures

The Traffic Anomaly engine contains nine anomaly detection signatures covering three protocols (TCP, UDP, and other). Each signature has two subsignatures, one for the scanner and the other for the worm-infected host (or a scanner under worm attack). When anomaly detection discovers an anomaly, it triggers an alert for these signatures. All anomaly detection signatures are enabled by default and the alert severity for each one is set to high.

When a scanner is detected but no histogram anomaly occurred, the scanner signature fires for that attacker (scanner) IP address. If the histogram signature is triggered, the attacker addresses that are doing the scanning each trigger the worm signature (instead of the scanner signature). The alert details state which threshold is being used for the worm detection now that the histogram has been triggered. From that point on, all scanners are detected as worm-infected hosts.

The following anomaly detection event actions are possible:

- produce-alert—Writes the event to the Event Store.
- deny-attacker-inline—(Inline only) Does not transmit this packet and future packets originating from the attacker address for a specified period of time.
- log-attacker-packets—Starts IP logging for packets that contain the attacker address.
- deny-attacker-service-pair-inline—Blocks the source IP address and the destination port.
- request-snmp-trapRequest—Sends a request to NotificationApp to perform SNMP notification.
- request-block-host—Sends a request to ARC to block this host (the attacker).

[Table 9-1](#) lists the anomaly detection worm signatures.

Table 9-1 Anomaly Detection Worm Signatures

Signature ID	Subsignature ID	Name	Description
13000	0	Internal TCP Scanner	Identified a single scanner over a TCP protocol in the internal zone.
13000	1	Internal TCP Scanner	Identified a worm attack over a TCP protocol in the internal zone; the TCP histogram threshold was crossed and a scanner over a TCP protocol was identified.
13001	0	Internal UDP Scanner	Identified a single scanner over a UDP protocol in the internal zone.
13001	1	Internal UDP Scanner	Identified a worm attack over a UDP protocol in the internal zone; the UDP histogram threshold was crossed and a scanner over a UDP protocol was identified.
13002	0	Internal Other Scanner	Identified a single scanner over an Other protocol in the internal zone.

Table 9-1 *Anomaly Detection Worm Signatures (continued)*

Signature ID	Subsignature ID	Name	Description
13002	1	Internal Other Scanner	Identified a worm attack over an Other protocol in the internal zone; the Other histogram threshold was crossed and a scanner over an Other protocol was identified.
13003	0	External TCP Scanner	Identified a single scanner over a TCP protocol in the external zone.
13003	1	External TCP Scanner	Identified a worm attack over a TCP protocol in the external zone; the TCP histogram threshold was crossed and a scanner over a TCP protocol was identified.
13004	0	External UDP Scanner	Identified a single scanner over a UDP protocol in the external zone.
13004	1	External UDP Scanner	Identified a worm attack over a UDP protocol in the external zone; the UDP histogram threshold was crossed and a scanner over a UDP protocol was identified.
13005	0	External Other Scanner	Identified a single scanner over an Other protocol in the external zone.
13005	1	External Other Scanner	Identified a worm attack over an Other protocol in the external zone; the Other histogram threshold was crossed and a scanner over an Other protocol was identified.
13006	0	Illegal TCP Scanner	Identified a single scanner over a TCP protocol in the illegal zone.
13006	1	Illegal TCP Scanner	Identified a worm attack over a TCP protocol in the illegal zone; the TCP histogram threshold was crossed and a scanner over a TCP protocol was identified.
13007	0	Illegal UDP Scanner	Identified a single scanner over a UDP protocol in the illegal zone.
13007	1	Illegal UDP Scanner	Identified a worm attack over a UDP protocol in the illegal zone; the UDP histogram threshold was crossed and a scanner over a UDP protocol was identified.

Table 9-1 Anomaly Detection Worm Signatures (continued)

Signature ID	Subsignature ID	Name	Description
13008	0	Illegal Other Scanner	Identified a single scanner over an Other protocol in the illegal zone.
13008	1	Illegal Other Scanner	Identified a worm attack over an Other protocol in the illegal zone; the Other histogram threshold was crossed and a scanner over an Other protocol was identified.

For More Information

For the procedure for assigning actions to signatures, see [Assigning Actions to Signatures](#), page 7-20.

Enabling Anomaly Detection

To enable anomaly detection, follow these steps:

-
- Step 1** Log in to the CLI using an account with administrator privileges.
- Step 2** Enter analysis engine submode.
- ```
sensor# configure terminal
sensor(config)# service analysis-engine
sensor(config-ana)#
```
- Step 3** Enter the virtual sensor name that contains the anomaly detection policy you want to enable.
- ```
sensor(config-ana)# virtual-sensor vs0
sensor(config-ana-vir)#
```
- Step 4** Enable anomaly detection operational mode.
- ```
sensor(config-ana-vir)# anomaly-detection
sensor(config-ana-vir-ano)# operational-mode detect
sensor(config-ana-vir-ano)#
```
- Step 5** Exit analysis engine submode.
- ```
sensor(config-ana-vir-ano)# exit
sensor(config-ana-vir)# exit
sensor(config-ana-)# exit
Apply Changes:[yes]:
```
- Step 6** Press **Enter** to apply your changes or enter **no** to discard them.
-

Working With Anomaly Detection Policies

Use the **service anomaly-detection name** command in service anomaly detection submode to create an anomaly detection policy. The values of this anomaly detection policy are the same as the default anomaly detection policy, ad0, until you edit them. Or you can use the **copy anomaly-detection source_destination** command in privileged EXEC mode to make a copy of an existing policy and then

edit the values of the new policy as needed. Use the **list anomaly-detection-configurations** command in privileged EXEC mode to list the anomaly detection policies. Use the **no service anomaly-detection name** command in global configuration mode to delete an anomaly detection policy. Use the **default service anomaly-detection name** command in global configuration mode to reset the anomaly detection policy to factory settings.

Working With Anomaly Detection Policies

To create, copy, display, edit, and delete anomaly detection policies, follow these steps:

Step 1 Log in to the CLI using an account with administrator privileges.

Step 2 Create an anomaly detection policy.

```
sensor# configure terminal
sensor(config)# service anomaly-detection MyAnomaly Detection
Editing new instance MyAnomaly Detection.
sensor(config-ano)# exit
Apply Changes?[yes]: yes
sensor(config)# exit
sensor#
```

Step 3 Or copy an existing anomaly detection policy to a new anomaly detection policy.

```
sensor# copy anomaly-detection ad0 ad1
sensor#
```



Note You receive an error if the policy already exists or if there is not enough space available for the new policy.

Step 4 Accept the default anomaly detection policy values or edit the following parameters:

- a. Configure the operational settings.
- b. Configure the zones.
- c. Configure learning accept mode.
- d. Learn how to work with KBs.

Step 5 Display a list of anomaly detection policies on the sensor.

```
sensor# list anomaly-detection-configurations
Anomaly Detection
  Instance   Size   Virtual Sensor
  ad0        255    vs0
  temp       707    N/A
  MyAnomaly Detection 255    N/A
  ad1        141    vs1
sensor#
```

Step 6 Delete an anomaly detection policy.

```
sensor# configure terminal
sensor(config)# no service anomaly-detection MyAnomaly Detection
sensor(config)# exit
sensor#
```



Note You cannot delete the default anomaly detection policy, ad0.

Step 7 Verify that the anomaly detection instance has been deleted.

```
sensor# list anomaly-detection-configurations
Anomaly Detection
  Instance   Size   Virtual Sensor
  ----     -
  ad0        204   vs0
  ad1        141   N/A
sensor#
```

Step 8 Reset an anomaly detection policy to factory settings.

```
sensor# configure terminal
sensor(config)# default service anomaly-detection ad1
sensor(config)#
```

For More Information

- For the procedure for configuring operational settings, see [Configuring Anomaly Detection Operational Settings, page 9-10](#).
- For the procedures for configuring anomaly detection zones, see [Configuring the Internal Zone, page 9-11](#), [Configuring the Illegal Zone, page 9-20](#), and [Configuring the External Zone, page 9-28](#).
- For the procedure for configuring learning accept mode, see [Configuring Learning Accept Mode, page 9-38](#).
- For the procedure for working with KBs, see [Working With KB Files, page 9-40](#).

Configuring Anomaly Detection Operational Settings

Use the **worm-timeout** command in service anomaly detection submode to set the worm detection timeout. After this timeout, the scanner threshold returns to the configured value. Use the **ignore** command in service anomaly detection submode to configure source and destination IP addresses that you want the sensor to ignore when anomaly detection is gathering information for a KB. Anomaly detection does not track these source and destination IP addresses and the KB thresholds are not affected by these IP addresses.

The following commands apply:

- **worm-timeout**—Specifies the amount of time in seconds for the worm termination timeout. The range is 120 to 10,000,000 seconds. The default is 600 seconds.
- **ignore**—Specifies the IP addresses that should be ignored while anomaly detection is processing:
 - **enabled {true | false}**—Enables/disables the list of ignored IP addresses. The default is enabled.
 - **source-ip-address-range**—Specifies the source IP addresses that you want anomaly detection to ignore during processing.
 - **dest-ip-address-range**—Specifies the destination IP addresses that you want anomaly detection to ignore during processing.



Note IP addresses are in the form of <A.B.C.D>-<A.B.C.D>[,<A.B.C.D>-<A.B.C.D>].

Configuring Anomaly Detection Operational Settings

To specify anomaly detection operational settings, follow these steps:

-
- Step 1** Log in to the CLI using an account with administrator privileges.
- Step 2** Enter anomaly detection submode.
- ```
sensor# configure terminal
sensor(config)# service anomaly-detection ad1
```
- Step 3** Specify the worm timeout.
- ```
sensor(config-ano)# worm-timeout 800
```
- Step 4** Verify the setting.
- ```
sensor(config-ano)# show settings
worm-timeout: 800 seconds default: 600
```
- Step 5** Specify the destination IP addresses that you want to be ignored while anomaly detection is processing.
- ```
sensor(config-ano)# ignore
sensor(config-ano-ign)# dest-ip-address-range 10.10.5.5,10.10.2.1-10.10.2.30
```
- Step 6** Specify the source IP addresses that you want to be ignored while anomaly detection is processing.
- ```
sensor(config-ano-ign)# source-ip-address-range 10.20.30.108-10.20.30.191
```
- Step 7** Verify the settings.
- ```
sensor(config-ano-ign)# show settings
ignore
-----
enabled: true default: true
source-ip-address-range: 10.20.30.108-10.20.30.191 default: 0.0.0.0
dest-ip-address-range: 10.10.5.5,10.10.2.1-10.10.2.30 default: 0.0.0.0
-----
sensor(config-ano-ign)#
```
- Step 8** Exit anomaly detection submode.
- ```
sensor(config-ano-ign)# exit
sensor(config-ano)# exit
Apply Changes:[yes]:
```
- Step 9** Press **Enter** to apply your changes or enter **no** to discard them.
- 

## Configuring the Internal Zone

This section describes how to configure the internal zone, and contains the following topics:

- [Understanding the Internal Zone, page 9-12](#)
- [Configuring the Internal Zone, page 9-12](#)
- [Configuring TCP Protocol for the Internal Zone, page 9-13](#)
- [Configuring UDP Protocol for the Internal Zone, page 9-15](#)
- [Configuring Other Protocols for the Internal Zone, page 9-18](#)

## Understanding the Internal Zone

The internal zone should represent your internal network. It should receive all the traffic that comes to your IP address range. If the zone is disabled, packets to this zone are ignored. By default the zone is enabled. You then add the IP addresses that belong to this zone. If you do not configure IP addresses for all zones, all packets are sent to the default zone, the external zone.

You can enable or disable TCP, UDP, and other protocols for the internal zone. You can configure a destination port for the TCP and UDP protocols and a protocol number for the other protocols. You can either use the default thresholds or override the scanner settings and add your own thresholds and histograms.

## Configuring the Internal Zone

Use the **internal-zone {enabled | ip-address-range | tcp | udp | other}** command in service anomaly-detection submode to enable the internal zone, add IP addresses to the internal zone, and specify protocols.

The following commands apply:

- **enabled {false | true}**—Enables/disables the zone.
- **ip-address-range**—Specifies the IP addresses of the subnets in the zone. The valid value is <A.B.C.D>-<A.B.C.D>[,<A.B.C.D>-<A.B.C.D>].



**Note** The second IP address in the range must be greater than or equal to the first IP address.

- **tcp**—Lets you configure TCP protocol.
- **udp**—Lets you configure UDP protocol.
- **other**—Lets you configure other protocols besides TCP and UDP.

### Configuring the Internal Zone

To configure the internal zone, follow these steps:

**Step 1** Log in to the CLI using an account with administrator privileges.

**Step 2** Enter anomaly detection internal zone submode.

```
sensor# configure terminal
sensor(config)# service anomaly-detection ad0
sensor(config-ano)# internal-zone
sensor(config-ano-int)#
```

**Step 3** Enable the internal zone.

```
sensor(config-ano-int)# enabled true
```

**Step 4** Configure the IP addresses to be included in the internal zone.

```
sensor(config-ano-int)# ip-address-range 192.0.2.72-192.0.2.108
```

**Step 5** Configure TCP protocol.

**Step 6** Configure UDP protocol.

**Step 7** Configure the other protocols.

---

**For More Information**

- For the procedure for configuring TCP protocol, see [Configuring TCP Protocol for the Internal Zone, page 9-13](#).
- For the procedure for configuring UDP protocol, see [Configuring UDP Protocol for the Internal Zone, page 9-15](#).
- For the procedure for configuring other protocols, see [Configuring Other Protocols for the Internal Zone, page 9-18](#).

## Configuring TCP Protocol for the Internal Zone

Use the `tcp {enabled | dst-port number | default-thresholds}` command in service anomaly detection internal zone submode to enable and configure the TCP service.

The following commands apply:

- **enabled {false | true}**—Enables/disables TCP protocol.
- **default-thresholds**—Defines thresholds to be used for all ports not specified in the destination port map:
  - **threshold-histogram {low | medium | high} num-source-ips *number***—Sets values in the threshold histogram.
  - **scanner-threshold**—Sets the scanner threshold. The default is 200.
- **dst-port *number***—Defines thresholds for specific destination ports. The valid values are 0 to 65535.
- **enabled {true | false}**—Enables/disables the service.
- **override-scanner-settings {yes | no}**—Lets you override the scanner values:
  - **threshold-histogram {low | medium | high} num-source-ips *number***—Sets values in the threshold histogram.
  - **scanner-threshold**—Sets the scanner threshold. The default is 200.

### Configuring Internal Zone TCP Protocol

To configure TCP protocol for the internal zone, follow these steps:

---

**Step 1** Log in to the CLI using an account with administrator privileges.

**Step 2** Enter anomaly detection internal zone submode.

```
sensor# configure terminal
sensor(config)# service anomaly-detection ad0
sensor(config-ano)# internal-zone
sensor(config-ano-int)#
```

**Step 3** Enable TCP protocol.

```
sensor(config-ano-int)# tcp
sensor(config-ano-int-tcp)# enabled true
```

**Step 4** Associate a specific port with TCP protocol.

```
sensor(config-ano-int-tcp)# dst-port 20
```

```
sensor(config-ano-int-tcp-dst)#
```

**Step 5** Enable the service for that port.

```
sensor(config-ano-int-tcp-dst)# enabled true
```

**Step 6** To override the scanner values for that port. You can use the default scanner values, or you can override them and configure your own scanner values.

```
sensor(config-ano-int-tcp-dst)# override-scanner-settings yes
sensor(config-ano-int-tcp-dst-yes)#
```

**Step 7** To add a histogram for the new scanner settings. Enter the number of destination IP addresses (low, medium, or high) and the number of source IP addresses you want associated with this histogram.

```
sensor(config-ano-int-tcp-dst-yes)# threshold-histogram low num-source-ips 100
```

**Step 8** Set the scanner threshold.

```
sensor(config-ano-int-tcp-dst-yes)# scanner-threshold 100
```

**Step 9** Configure the default thresholds for all other unspecified ports.

```
sensor(config-ano-int-tcp-dst-yes)# exit
sensor(config-ano-int-tcp-dst)# exit
sensor(config-ano-int-tcp)# exit
sensor(config-ano-int-tcp)# default-thresholds
sensor(config-ano-int-tcp-def)# default-thresholds
sensor(config-ano-int-tcp-def)# threshold-histogram medium num-source-ips 120
sensor(config-ano-int-tcp-def)# scanner-threshold 120
```

**Step 10** Verify the TCP configuration settings.

```
sensor(config-ano-int-tcp)# show settings
tcp

dst-port (min: 0, max: 65535, current: 4)

number: 20

override-scanner-settings

yes

scanner-threshold: 120 default: 200
threshold-histogram (min: 0, max: 3, current: 1)

dest-ip-bin: low
num-source-ips: 100

enabled: true default: true

number: 23

override-scanner-settings

no

enabled: true <defaulted>

number: 113
```

```

override-scanner-settings

no

enabled: true <defaulted>

number: 567

override-scanner-settings

no

enabled: true <defaulted>

default-thresholds

scanner-threshold: 120 default: 200
threshold-histogram (min: 0, max: 3, current: 3)

<protected entry>
dest-ip-bin: low <defaulted>
num-source-ips: 10 <defaulted>
<protected entry>
dest-ip-bin: medium
num-source-ips: 120 default: 1
<protected entry>
dest-ip-bin: high <defaulted>
num-source-ips: 1 <defaulted>

enabled: true <defaulted>

sensor(config-ano-int-tcp)#

```

## Configuring UDP Protocol for the Internal Zone

Use the **udp** {**enabled** | **dst-port** *number* | **default-thresholds**} command in service anomaly detection internal zone submode to enable and configure the UDP service.

The following commands apply:

- **enabled** {**false** | **true**}—Enables/disables UDP protocol.
- **default-thresholds**—Defines thresholds to be used for all ports not specified in the destination port map:
  - **threshold-histogram** {**low** | **medium** | **high**} **num-source-ips** *number*—Sets values in the threshold histogram.
  - **scanner-threshold**—Sets the scanner threshold. The default is 200.
- **dst-port** *number*—Defines thresholds for specific destination ports. The valid values are 0 to 65535.
- **enabled** {**true** | **false**}—Enables/disables the service.

- **override-scanner-settings** {yes | no}—Lets you override the scanner values:
  - **threshold-histogram** {low | medium | high} **num-source-ips** *number*—Sets values in the threshold histogram.
  - **scanner-threshold**—Sets the scanner threshold. The default is 200.

### Configuring the Internal Zone UDP Protocol

To configure UDP protocol for a zone, follow these steps:

- 
- Step 1** Log in to the CLI using an account with administrator privileges.
- Step 2** Enter anomaly detection internal zone submode.
- ```
sensor# configure terminal
sensor(config)# service anomaly-detection ad0
sensor(config-ano)# internal-zone
sensor(config-ano-int)#
```
- Step 3** Enable UDP protocol.
- ```
sensor(config-ano-int)# udp
sensor(config-ano-int-udp)# enabled true
```
- Step 4** Associate a specific port with UDP protocol.
- ```
sensor(config-ano-int-udp)# dst-port 20
sensor(config-ano-int-udp-dst)#
```
- Step 5** Enable the service for that port.
- ```
sensor(config-ano-int-udp-dst)# enabled true
```
- Step 6** To override the scanner values for that port. You can use the default scanner values, or you can override them and configure your own scanner values.
- ```
sensor(config-ano-int-udp-dst)# override-scanner-settings yes
sensor(config-ano-int-udp-dst-yes)#
```
- Step 7** To add a histogram for the new scanner settings. Enter the number of destination IP addresses (low, medium, or high) and the number of source IP addresses you want associated with this histogram.
- ```
sensor(config-ano-int-udp-dst-yes)# threshold-histogram low num-source-ips 100
```
- Step 8** Set the scanner threshold.
- ```
sensor(config-ano-int-udp-dst-yes)# scanner-threshold 100
```
- Step 9** Configure the default thresholds for all other unspecified ports.
- ```
sensor(config-ano-int-udp-dst-yes)# exit
sensor(config-ano-int-udp-dst)# exit
sensor(config-ano-int-udp)# default-thresholds
sensor(config-ano-int-udp-def)# default-thresholds
sensor(config-ano-int-udp-def)# threshold-histogram medium num-source-ips 120
sensor(config-ano-int-udp-def)# scanner-threshold 120
```
- Step 10** Verify the UDP configuration settings.
- ```
sensor(config-ano-int-udp)# show settings
udp
-----
dst-port (min: 0, max: 65535, current: 4)
-----
number: 20
```



```

-----
override-scanner-settings
-----
    yes
-----
        scanner-threshold: 100 default: 200
        threshold-histogram (min: 0, max: 3, current: 1)
-----
            dest-ip-bin: low
            num-source-ips: 100
-----
-----
enabled: true default: true
-----
number: 23
-----
override-scanner-settings
-----
    no
-----
-----
enabled: true <defaulted>
-----
number: 113
-----
override-scanner-settings
-----
    no
-----
-----
enabled: true <defaulted>
-----
number: 567
-----
override-scanner-settings
-----
    no
-----
-----
enabled: true <defaulted>
-----
-----
default-thresholds
-----
scanner-threshold: 120 default: 200
threshold-histogram (min: 0, max: 3, current: 3)
-----
    <protected entry>
    dest-ip-bin: low <defaulted>
    num-source-ips: 10 <defaulted>
    <protected entry>
    dest-ip-bin: medium
    num-source-ips: 120 default: 1
    <protected entry>
    dest-ip-bin: high <defaulted>
    num-source-ips: 1 <defaulted>
-----
-----
enabled: true <defaulted>
-----

```

```
sensor(config-ano-int-udp)#
```

Configuring Other Protocols for the Internal Zone

Use the **other** {**enabled** | **protocol** *number* | **default-thresholds**} command in service anomaly detection internal zone submode to enable and configure the other services.

The following commands apply:

- **enabled** {**false** | **true**}—Enables/disables other protocols.
- **default-thresholds**—Defines thresholds to be used for all ports not specified in the destination port map:
 - **threshold-histogram** {**low** | **medium** | **high**} **num-source-ips** *number*—Sets values in the threshold histogram.
 - **scanner-threshold**—Sets the scanner threshold. The default is 200.
- **protocol-number** *number*—Defines thresholds for specific protocols. The valid values are 0 to 255.
- **enabled** {**true** | **false**}—Enables/disables the service.
- **override-scanner-settings** {**yes** | **no**}—Lets you override the scanner values:
 - **threshold-histogram** {**low** | **medium** | **high**} **num-source-ips** *number*—Sets values in the threshold histogram.
 - **scanner-threshold**—Sets the scanner threshold. The default is 200.

Configuring the Internal Zone Other Protocols

To configure other protocols for a zone, follow these steps:

Step 1 Log in to the CLI using an account with administrator privileges.

Step 2 Enter anomaly detection internal zone submode.

```
sensor# configure terminal
sensor(config)# service anomaly-detection ad0
sensor(config-ano)# internal-zone
sensor(config-ano-int)#
```

Step 3 Enable the other protocols.

```
sensor(config-ano-int)# other
sensor(config-ano-int-oth)# enabled true
```

Step 4 Associate a specific number for the other protocols.

```
sensor(config-ano-int-oth)# protocol-number 5
sensor(config-ano-int-oth-pro)#
```

Step 5 Enable the service for that port.

```
sensor(config-ano-int-oth-pro)# enabled true
```

Step 6 To override the scanner values for that protocol. You can use the default scanner values, or you can override them and configure your own scanner values.

```
sensor(config-ano-int-oth-pro)# override-scanner-settings yes
sensor(config-ano-int-oth-pro-yes)#
```

- Step 7** To add a histogram for the new scanner settings. Enter the number of destination IP addresses (low, medium, or high) and the number of source IP addresses you want associated with this histogram.

```
sensor(config-ano-int-oth-pro-yes)# threshold-histogram high num-source-ips 75
```

- Step 8** Set the scanner threshold.

```
sensor(config-ano-int-oth-pro-yes)# scanner-threshold 100
```

- Step 9** Configure the default thresholds for all other unspecified ports.

```
sensor(config-ano-int-oth-pro-yes)# exit
sensor(config-ano-int-oth-pro)# exit
sensor(config-ano-int-oth)# default-thresholds
sensor(config-ano-int-oth-def)# default-thresholds
sensor(config-ano-int-oth-def)# threshold-histogram medium num-source-ips 120
sensor(config-ano-int-oth-def)# scanner-threshold 120
```

- Step 10** Verify the other configuration settings.

```
sensor(config-ano-int-oth)# show settings
other
-----
protocol-number (min: 0, max: 255, current: 1)
-----
number: 5
-----
override-scanner-settings
-----
yes
-----
scanner-threshold: 95 default: 200
threshold-histogram (min: 0, max: 3, current: 1)
-----
dest-ip-bin: high
num-source-ips: 75
-----
-----
enabled: true default: true
-----
default-thresholds
-----
scanner-threshold: 200 <defaulted>
threshold-histogram (min: 0, max: 3, current: 3)
-----
<protected entry>
dest-ip-bin: low <defaulted>
num-source-ips: 10 <defaulted>
<protected entry>
dest-ip-bin: medium <defaulted>
num-source-ips: 1 <defaulted>
<protected entry>
dest-ip-bin: high <defaulted>
num-source-ips: 1 <defaulted>
-----
-----
enabled: true default: true
-----
sensor(config-ano-int-oth)#
```

Configuring the Illegal Zone

This section describes how to configure the illegal zone, and contains the following topics:

- [Understanding the Illegal Zone, page 9-20](#)
- [Configuring the Illegal Zone, page 9-20](#)
- [Configuring TCP Protocol for the Illegal Zone, page 9-21](#)
- [Configuring UDP Protocol for the Illegal Zone, page 9-24](#)
- [Configuring Other Protocols for the Illegal Zone, page 9-26](#)

Understanding the Illegal Zone

The illegal zone should represent IP address ranges that should never be seen in normal traffic, for example, unallocated IP addresses or part of your internal IP address range that is unoccupied. You then add the IP addresses that belong to this zone. If you do not configure IP addresses for all zones, all packets are sent to the default zone, the external zone.

You can enable or disable TCP, UDP, and other protocols for the internal zone. You can configure a destination port for the TCP and UDP protocols and a protocol number for the other protocols. You can either use the default thresholds or override the scanner settings and add your own thresholds and histograms.

Configuring the Illegal Zone

Use the **illegal-zone** {**enabled** | **ip-address-range** | **tcp** | **udp** | **lother**} command in service anomaly detection submode to enable the illegal zone, add IP addresses to the illegal zone, and specify protocols.

The following commands apply:

- **enabled** {**false** | **true**}—Enables/disables the zone.
- **ip-address-range**—Specifies the IP addresses of the subnets in the zone. The valid value is <A.B.C.D>-<A.B.C.D>[,<A.B.C.D>-<A.B.C.D>].



Note The second IP address in the range must be greater than or equal to the first IP address.

- **tcp**—Lets you configure TCP protocol.
- **udp**—Lets you configure UDP protocol.
- **other**—Lets you configure other protocols besides TCP and UDP.

Configuring the Illegal Zone

To configure the illegal zone, follow these steps:

-
- Step 1** Log in to the CLI using an account with administrator privileges.
 - Step 2** Enter anomaly detection illegal zone submode.

```
sensor# configure terminal
sensor(config)# service anomaly-detection ad0
sensor(config-ano)# illegal-zone
```

```
sensor(config-ano-ill)#
```

Step 3 Enable the illegal zone.

```
sensor(config-ano-ill)# enabled true
```

Step 4 Configure the IP addresses to be included in the illegal zone.

```
sensor(config-ano-ill)# ip-address-range 192.0.2.72-192.0.2.108
```

Step 5 Configure TCP protocol.

Step 6 Configure UDP protocol.

Step 7 Configure the other protocols.

For More Information

- For the procedure for configuring TCP protocol, see [Configuring TCP Protocol for the Illegal Zone, page 9-21](#).
- For the procedure for the UDP protocol, see [Configuring UDP Protocol for the Illegal Zone, page 9-24](#).
- For the procedure for configuring other protocols, see [Configuring Other Protocols for the Illegal Zone, page 9-26](#).

Configuring TCP Protocol for the Illegal Zone

Use the `tcp {enabled | dst-port number | default-thresholds}` command in service anomaly detection illegal zone submode to enable and configure the TCP service.

The following commands apply:

- **enabled {false | true}**—Enables/disables TCP protocol.
- **default-thresholds**—Defines thresholds to be used for all ports not specified in the destination port map:
 - **threshold-histogram {low | medium | high} num-source-ips number**—Sets values in the threshold histogram.
 - **scanner-threshold**—Sets the scanner threshold. The default is 200.
- **dst-port number**—Defines thresholds for specific destination ports. The valid values are 0 to 65535.
- **enabled {true | false}**—Enables/disables the service.
- **override-scanner-settings {yes | no}**—Lets you override the scanner values:
 - **threshold-histogram {low | medium | high} num-source-ips number**—Sets values in the threshold histogram.
 - **scanner-threshold**—Sets the scanner threshold. The default is 200.

Configuring the Illegal Zone TCP Protocol

To configure TCP protocol for illegal zone, follow these steps:

Step 1 Log in to the CLI using an account with administrator privileges.

Step 2 Enter anomaly detection illegal zone submode.

```
sensor# configure terminal
sensor(config)# service anomaly-detection ad0
sensor(config-ano)# illegal-zone
sensor(config-ano-ill)#
```

Step 3 Enable TCP protocol.

```
sensor(config-ano-ill)# tcp
sensor(config-ano-ill-tcp)# enabled true
```

Step 4 Associate a specific port with TCP protocol.

```
sensor(config-ano-ill-tcp)# dst-port 20
sensor(config-ano-ill-tcp-dst)#
```

Step 5 Enable the service for that port.

```
sensor(config-ano-ill-tcp-dst)# enabled true
```

Step 6 Override the scanner values for that port. You can use the default scanner values, or you can override them and configure your own scanner values.

```
sensor(config-ano-ill-tcp-dst)# override-scanner-settings yes
sensor(config-ano-ill-tcp-dst-yes)#
```

Step 7 Add a histogram for the new scanner settings. Enter the number of destination IP addresses (low, medium, or high) and the number of source IP addresses you want associated with this histogram.

```
sensor(config-ano-ill-tcp-dst-yes)# threshold-histogram low num-source-ips 100
```

Step 8 Set the scanner threshold.

```
sensor(config-ano-ill-tcp-dst-yes)# scanner-threshold 100
```

Step 9 Configure the default thresholds for all other unspecified ports.

```
sensor(config-ano-ill-tcp-dst-yes)# exit
sensor(config-ano-ill-tcp-dst)# exit
sensor(config-ano-ill-tcp)# exit
sensor(config-ano-ill-tcp)# default-thresholds
sensor(config-ano-ill-tcp-def)# default-thresholds
sensor(config-ano-ill-tcp-def)# threshold-histogram medium num-source-ips 120
sensor(config-ano-ill-tcp-def)# scanner-threshold 120
```

Step 10 Verify the TCP configuration settings.

```
sensor(config-ano-ill-tcp)# show settings
tcp
-----
dst-port (min: 0, max: 65535, current: 4)
-----
number: 20
-----
override-scanner-settings
-----
yes
-----
scanner-threshold: 100 default: 200
```

```

threshold-histogram (min: 0, max: 3, current: 1)
-----
    dest-ip-bin: low
    num-source-ips: 100
-----
-----
enabled: true default: true
-----
number: 23
-----
override-scanner-settings
-----
    no
-----
-----
enabled: true <defaulted>
-----
number: 113
-----
override-scanner-settings
-----
    no
-----
-----
enabled: true <defaulted>
-----
number: 567
-----
override-scanner-settings
-----
    no
-----
-----
enabled: true <defaulted>
-----
-----
default-thresholds
-----
scanner-threshold: 120 default: 200
threshold-histogram (min: 0, max: 3, current: 3)
-----
    <protected entry>
    dest-ip-bin: low <defaulted>
    num-source-ips: 10 <defaulted>
    <protected entry>
    dest-ip-bin: medium
    num-source-ips: 120 default: 1
    <protected entry>
    dest-ip-bin: high <defaulted>
    num-source-ips: 1 <defaulted>
-----
-----
enabled: true <defaulted>
-----
sensor(config-ano-ill-tcp)#

```

Configuring UDP Protocol for the Illegal Zone

Use the `udp {enabled | dst-port number | default-thresholds}` command in service anomaly detection illegal zone submode to enable and configure the UDP service.

The following commands apply:

- **enabled {false | true}**—Enables/disables UDP protocol.
- **default-thresholds**—Defines thresholds to be used for all ports not specified in the destination port map:
 - **threshold-histogram {low | medium | high} num-source-ips number**—Sets values in the threshold histogram.
 - **scanner-threshold**—Sets the scanner threshold. The default is 200.
- **dst-port number**—Defines thresholds for specific destination ports. The valid values are 0 to 65535.
- **enabled {true | false}**—Enables/disables the service.
- **override-scanner-settings {yes | no}**—Lets you override the scanner values:
 - **threshold-histogram {low | medium | high} num-source-ips number**—Sets values in the threshold histogram.
 - **scanner-threshold**—Sets the scanner threshold. The default is 200.

Configuring the Illegal Zone UDP Protocol

To configure UDP protocol for a zone, follow these steps:

Step 1 Log in to the CLI using an account with administrator privileges.

Step 2 Enter anomaly detection illegal zone submode.

```
sensor# configure terminal
sensor(config)# service anomaly-detection ad0
sensor(config-ano)# illegal-zone
sensor(config-ano-ill)#
```

Step 3 Enable UDP protocol.

```
sensor(config-ano-ill)# udp
sensor(config-ano-ill-udp)# enabled true
```

Step 4 Associate a specific port with UDP protocol.

```
sensor(config-ano-ill-udp)# dst-port 20
sensor(config-ano-ill-udp-dst)#
```

Step 5 Enable the service for that port.

```
sensor(config-ano-ill-udp-dst)# enabled true
```

Step 6 Override the scanner values for that port. You can use the default scanner values, or you can override them and configure your own scanner values.

```
sensor(config-ano-ill-udp-dst)# override-scanner-settings yes
sensor(config-ano-ill-udp-dst-yes)#
```

Step 7 Add a histogram for the new scanner settings. Enter the number of destination IP addresses (low, medium, or high) and the number of source IP addresses you want associated with this histogram.

```
sensor(config-ano-ill-udp-dst-yes)# threshold-histogram low num-source-ips 100
```


Step 8 Set the scanner threshold.

```
sensor(config-ano-ill-udp-dst-yes)# scanner-threshold 100
```

Step 9 Configure the default thresholds for all other unspecified ports.

```
sensor(config-ano-ill-udp-dst-yes)# exit
sensor(config-ano-ill-udp-dst)# exit
sensor(config-ano-ill-udp)# exit
sensor(config-ano-ill-udp)# default-thresholds
sensor(config-ano-ill-udp-def)# default-thresholds
sensor(config-ano-ill-udp-def)# threshold-histogram medium num-source-ips 120
sensor(config-ano-ill-udp-def)# scanner-threshold 120
```

Step 10 Verify the UDP configuration settings.

```
sensor(config-ano-ill-udp)# show settings
udp
-----
dst-port (min: 0, max: 65535, current: 4)
-----
number: 20
-----
override-scanner-settings
-----
yes
-----
scanner-threshold: 100 default: 200
threshold-histogram (min: 0, max: 3, current: 1)
-----
dest-ip-bin: low
num-source-ips: 100
-----
-----
enabled: true default: true
-----
number: 23
-----
override-scanner-settings
-----
no
-----
-----
enabled: true <defaulted>
-----
number: 113
-----
override-scanner-settings
-----
no
-----
-----
enabled: true <defaulted>
-----
number: 567
-----
override-scanner-settings
-----
no
-----
-----
-----
```

```

        enabled: true <defaulted>
-----
default-thresholds
-----
scanner-threshold: 120 default: 200
threshold-histogram (min: 0, max: 3, current: 3)
-----
<protected entry>
dest-ip-bin: low <defaulted>
num-source-ips: 10 <defaulted>
<protected entry>
dest-ip-bin: medium
num-source-ips: 120 default: 1
<protected entry>
dest-ip-bin: high <defaulted>
num-source-ips: 1 <defaulted>
-----
enabled: true <defaulted>
-----
sensor(config-ano-ill-udp)#

```

Configuring Other Protocols for the Illegal Zone

Use the **other** {**enabled** | **protocol** *number* | **default-thresholds**} command in service anomaly detection illegal zone submode to enable and configure the other services.

The following commands apply:

- **enabled** {**false** | **true**}—Enables/disables other protocols.
- **default-thresholds**—Defines thresholds to be used for all ports not specified in the destination port map:
 - **threshold-histogram** {**low** | **medium** | **high**} **num-source-ips** *number*—Sets values in the threshold histogram.
 - **scanner-threshold**—Sets the scanner threshold. The default is 200.
- **protocol-number** *number*—Defines thresholds for specific protocols. The valid values are 0 to 255.
- **enabled** {**true** | **false**}—Enables/disables the service.
- **override-scanner-settings** {**yes** | **no**}—Lets you override the scanner values:
 - **threshold-histogram** {**low** | **medium** | **high**} **num-source-ips** *number*—Sets values in the threshold histogram.
 - **scanner-threshold**—Sets the scanner threshold. The default is 200.

Configuring the Illegal Zone Other Protocols

To configure other protocols for a zone, follow these steps:

- Step 1** Log in to the CLI using an account with administrator privileges.
- Step 2** Enter anomaly detection illegal zone submode.

```

sensor# configure terminal
sensor(config)# service anomaly-detection ad0

```

```
sensor(config-ano)# illegal-zone
sensor(config-ano-ill)#
```

Step 3 Enable the other protocols.

```
sensor(config-ano-ill)# other
sensor(config-ano-ill-oth)# enabled true
```

Step 4 Associate a specific number for the other protocols.

```
sensor(config-ano-ill-oth)# protocol-number 5
sensor(config-ano-ill-oth-pro)#
```

Step 5 Enable the service for that port.

```
sensor(config-ano-ill-oth-pro)# enabled true
```

Step 6 Override the scanner values for that protocol. You can use the default scanner values, or you can override them and configure your own scanner values.

```
sensor(config-ano-ill-oth-pro)# override-scanner-settings yes
sensor(config-ano-ill-oth-pro-yes)#
```

Step 7 Add a histogram for the new scanner settings. Enter the number of destination IP addresses (low, medium, or high) and the number of source IP addresses you want associated with this histogram.

```
sensor(config-ano-ill-oth-pro-yes)# threshold-histogram high num-source-ips 75
```

Step 8 Set the scanner threshold.

```
sensor(config-ano-ill-oth-pro-yes)# scanner-threshold 100
```

Step 9 Configure the default thresholds for all other unspecified ports.

```
sensor(config-ano-ill-oth-pro-yes)# exit
sensor(config-ano-ill-oth-pro)# exit
sensor(config-ano-ill-oth)# default-thresholds
sensor(config-ano-ill-oth-def)# default-thresholds
sensor(config-ano-ill-oth-def)# threshold-histogram medium num-source-ips 120
sensor(config-ano-ill-oth-def)# scanner-threshold 120
```

Step 10 Verify the other protocols configuration settings.

```
sensor(config-ano-ill-oth)# show settings
other
-----
protocol-number (min: 0, max: 255, current: 1)
-----
number: 5
-----
override-scanner-settings
-----
yes
-----
scanner-threshold: 95 default: 200
threshold-histogram (min: 0, max: 3, current: 1)
-----
dest-ip-bin: high
num-source-ips: 75
-----
-----
enabled: true default: true
-----
-----
default-thresholds
```

```

-----
scanner-threshold: 200 <defaulted>
threshold-histogram (min: 0, max: 3, current: 3)
-----
<protected entry>
dest-ip-bin: low <defaulted>
num-source-ips: 10 <defaulted>
<protected entry>
dest-ip-bin: medium <defaulted>
num-source-ips: 1 <defaulted>
<protected entry>
dest-ip-bin: high <defaulted>
num-source-ips: 1 <defaulted>
-----
enabled: true default: true
-----
sensor(config-ano-ill-oth)#

```

Configuring the External Zone

This section describes how to configure the external zone, and contains the following topics:

- [Understanding the External Zone, page 9-28](#)
- [Configuring the External Zone, page 9-28](#)
- [Configuring TCP Protocol for the External Zone, page 9-29](#)
- [Configuring UDP Protocol for the External Zone, page 9-32](#)
- [Configuring Other Protocols for the External Zone, page 9-34](#)

Understanding the External Zone

The external zone is the default zone with the default Internet range of 0.0.0.0-255.255.255.255. By default, the internal and illegal zones contain no IP addresses. Packets that do not match the set of IP addresses in the internal or illegal zone are handled by the external zone.

You can enable or disable TCP, UDP, and other protocols for the external zone. You can configure a destination port for the TCP and UDP protocols and a protocol number for the other protocols. You can either use the default thresholds or override the scanner settings and add your own thresholds and histograms.

Configuring the External Zone

Use the **external-zone {enabled | tcp | udp | other}** command in service anomaly detection submode to enable the external zone and specify protocols.

The following commands apply:

- **enabled {false | true}**—Enables/disables the zone.
- **tcp**—Lets you configure TCP protocol.
- **udp**—Lets you configure UDP protocol.

- **other**—Lets you configure other protocols besides TCP and UDP.

Configuring the External Zone

To configure the external zone, follow these steps:

-
- Step 1** Log in to the CLI using an account with administrator privileges.
- Step 2** Enter anomaly detection external zone submode.
- ```
sensor# configure terminal
sensor(config)# service anomaly-detection ad0
sensor(config-ano)# external-zone
sensor(config-ano-ext)#
```
- Step 3** Enable the external zone.
- ```
sensor(config-ano-ext)# enabled true
```
- Step 4** Configure TCP protocol.
- Step 5** Configure UDP protocol.
- Step 6** Configure the other protocols.
-

For More Information

- For the procedure for configuring TCP protocol, see [Configuring TCP Protocol for the External Zone, page 9-29](#).
- For the procedure for configuring UDP protocol, see [Configuring UDP Protocol for the External Zone, page 9-32](#).
- For the procedure for configuring other protocols, see [Configuring Other Protocols for the External Zone, page 9-34](#).

Configuring TCP Protocol for the External Zone

Use the `tcp {enabled | dst-port number | default-thresholds}` command in service anomaly detection external zone submode to enable and configure the TCP service.

The following commands apply:

- **enabled {false | true}**—Enables/disables TCP protocol.
- **default-thresholds**—Defines thresholds to be used for all ports not specified in the destination port map:
 - **threshold-histogram {low | medium | high} num-source-ips number**—Sets values in the threshold histogram.
 - **scanner-threshold**—Sets the scanner threshold. The default is 200.
- **dst-port number**—Defines thresholds for specific destination ports. The valid values are 0 to 65535.
- **enabled {true | false}**—Enables/disables the service.
- **override-scanner-settings {yes | no}**—Lets you override the scanner values:
 - **threshold-histogram {low | medium | high} num-source-ips number**—Sets values in the threshold histogram.

- **scanner-threshold**—Sets the scanner threshold. The default is 200.

Configuring the External Zone TCP Protocol

To configure TCP protocol for the external zone, follow these steps:

Step 1 Log in to the CLI using an account with administrator privileges.

Step 2 Enter anomaly detection external zone submode.

```
sensor# configure terminal
sensor(config)# service anomaly-detection ad0
sensor(config-ano)# external-zone
sensor(config-ano-ext)#
```

Step 3 Enable TCP protocol.

```
sensor(config-ano-ext)# tcp
sensor(config-ano-ext-tcp)# enabled true
```

Step 4 Associate a specific port with TCP protocol.

```
sensor(config-ano-ext-tcp)# dst-port 20
sensor(config-ano-ext-tcp-dst)#
```

Step 5 Enable the service for that port.

```
sensor(config-ano-ext-tcp-dst)# enabled true
```

Step 6 Override the scanner values for that port. You can use the default scanner values, or you can override them and configure your own scanner values.

```
sensor(config-ano-ext-tcp-dst)# override-scanner-settings yes
sensor(config-ano-ext-tcp-dst-yes)#
```

Step 7 Add a histogram for the new scanner settings. Enter the number of destination IP addresses (low, medium, or high) and the number of source IP addresses you want associated with this histogram.

```
sensor(config-ano-ext-tcp-dst-yes)# threshold-histogram low num-source-ips 100
```

Step 8 Set the scanner threshold.

```
sensor(config-ano-ext-tcp-dst-yes)# scanner-threshold 100
```

Step 9 Configure the default thresholds for all other unspecified ports.

```
sensor(config-ano-ext-tcp-dst-yes)# exit
sensor(config-ano-ext-tcp-dst)# exit
sensor(config-ano-ext-tcp)# exit
sensor(config-ano-ext-tcp)# default-thresholds
sensor(config-ano-ext-tcp-def)# default-thresholds
sensor(config-ano-ext-tcp-def)# threshold-histogram medium num-source-ips 120
sensor(config-ano-ext-tcp-def)# scanner-threshold 120
```

Step 10 Verify the TCP configuration settings.

```
sensor(config-ano-ext-tcp)# show settings
tcp
-----
dst-port (min: 0, max: 65535, current: 4)
-----
number: 20
-----
override-scanner-settings
-----
```

```

yes
-----
scanner-threshold: 100 default: 200
threshold-histogram (min: 0, max: 3, current: 1)
-----
dest-ip-bin: low
num-source-ips: 100
-----
-----
enabled: true default: true
-----
number: 23
-----
override-scanner-settings
-----
no
-----
-----
enabled: true <defaulted>
-----
number: 113
-----
override-scanner-settings
-----
no
-----
-----
enabled: true <defaulted>
-----
number: 567
-----
override-scanner-settings
-----
no
-----
-----
enabled: true <defaulted>
-----
-----
default-thresholds
-----
scanner-threshold: 120 default: 200
threshold-histogram (min: 0, max: 3, current: 3)
-----
<protected entry>
dest-ip-bin: low <defaulted>
num-source-ips: 10 <defaulted>
<protected entry>
dest-ip-bin: medium
num-source-ips: 120 default: 1
<protected entry>
dest-ip-bin: high <defaulted>
num-source-ips: 1 <defaulted>
-----
-----

```

```

        enabled: true <defaulted>
-----
sensor(config-ano-ext-tcp)#

```

Configuring UDP Protocol for the External Zone

Use the **udp** {**enabled** | **dst-port** *number* | **default-thresholds**} command in service anomaly detection external zone submode to enable and configure the UDP service.

The following commands apply:

- **enabled** {**false** | **true**}—Enables/disables UDP protocol.
- **default-thresholds**—Defines thresholds to be used for all ports not specified in the destination port map:
 - **threshold-histogram** {**low** | **medium** | **high**} **num-source-ips** *number*—Sets values in the threshold histogram.
 - **scanner-threshold**—Sets the scanner threshold. The default is 200.
- **dst-port** *number*—Defines thresholds for specific destination ports. The valid values are 0 to 65535.
- **enabled** {**true** | **false**}—Enables/disables the service.
- **override-scanner-settings** {**yes** | **no**}—Lets you override the scanner values:
 - **threshold-histogram** {**low** | **medium** | **high**} **num-source-ips** *number*—Sets values in the threshold histogram.
 - **scanner-threshold**—Sets the scanner threshold. The default is 200.

Configuring the External Zone UDP Protocol

To configure UDP protocol for a zone, follow these steps:

Step 1 Log in to the CLI using an account with administrator privileges.

Step 2 Enter anomaly detection external zone submode.

```

sensor# configure terminal
sensor(config)# service anomaly-detection ad0
sensor(config-ano)# external-zone
sensor(config-ano-ext)#

```

Step 3 Enable UDP protocol.

```

sensor(config-ano-ext)# udp
sensor(config-ano-ext-udp)# enabled true

```

Step 4 Associate a specific port with UDP protocol.

```

sensor(config-ano-ext-udp)# dst-port 20
sensor(config-ano-ext-udp-dst)#

```

Step 5 Enable the service for that port.

```

sensor(config-ano-ext-udp-dst)# enabled true

```


- Step 6** Override the scanner values for that port. You can use the default scanner values, or you can override them and configure your own scanner values.

```
sensor(config-ano-ext-udp-dst)# override-scanner-settings yes
sensor(config-ano-ext-udp-dst-yes)#
```

- Step 7** Add a histogram for the new scanner settings. Enter the number of destination IP addresses (low, medium, or high) and the number of source IP addresses you want associated with this histogram.

```
sensor(config-ano-ext-udp-dst-yes)# threshold-histogram low num-source-ips 100
```

- Step 8** Set the scanner threshold.

```
sensor(config-ano-ext-udp-dst-yes)# scanner-threshold 100
```

- Step 9** Configure the default thresholds for all other unspecified ports.

```
sensor(config-ano-ext-udp-dst-yes)# exit
sensor(config-ano-ext-udp-dst)# exit
sensor(config-ano-ext-udp)# default-thresholds
sensor(config-ano-ext-udp-def)# default-thresholds
sensor(config-ano-ext-udp-def)# threshold-histogram medium num-source-ips 120
sensor(config-ano-ext-udp-def)# scanner-threshold 120
```

- Step 10** Verify the UDP configuration settings.

```
sensor(config-ano-ext-udp)# show settings
udp
-----
dst-port (min: 0, max: 65535, current: 4)
-----
number: 20
-----
override-scanner-settings
-----
yes
-----
scanner-threshold: 100 default: 200
threshold-histogram (min: 0, max: 3, current: 1)
-----
dest-ip-bin: low
num-source-ips: 100
-----
-----
enabled: true default: true
-----
number: 23
-----
override-scanner-settings
-----
no
-----
-----
enabled: true <defaulted>
-----
number: 113
-----
override-scanner-settings
-----
no
-----
-----
-----
```

```

        enabled: true <defaulted>
-----
number: 567
-----
override-scanner-settings
-----
        no
-----
-----
        enabled: true <defaulted>
-----
default-thresholds
-----
scanner-threshold: 120 default: 200
threshold-histogram (min: 0, max: 3, current: 3)
-----
<protected entry>
dest-ip-bin: low <defaulted>
num-source-ips: 10 <defaulted>
<protected entry>
dest-ip-bin: medium
num-source-ips: 120 default: 1
<protected entry>
dest-ip-bin: high <defaulted>
num-source-ips: 1 <defaulted>
-----
enabled: true <defaulted>
-----
sensor (config-ano-ext-udp) #

```

Configuring Other Protocols for the External Zone

Use the **other** { **enabled** | **protocol** *number* | **default-thresholds** } command in service anomaly detection external zone submode to enable and configure the other services.

The following commands apply:

- **enabled** { **false** | **true** }—Enables/disables other protocols.
- **default-thresholds**—Defines thresholds to be used for all ports not specified in the destination port map:
 - **threshold-histogram** { **low** | **medium** | **high** } **num-source-ips** *number*—Sets values in the threshold histogram.
 - **scanner-threshold**—Sets the scanner threshold. The default is 200.
- **protocol-number** *number*—Defines thresholds for specific protocols. The valid values are 0 to 255.
- **enabled** { **true** | **false** }—Enables/disables the service.
- **override-scanner-settings** { **yes** | **no** }—Lets you override the scanner values:
 - **threshold-histogram** { **low** | **medium** | **high** } **num-source-ips** *number*—Sets values in the threshold histogram.
 - **scanner-threshold**—Sets the scanner threshold. The default is 200.

Configuring the External Zone Other Protocols

To configure other protocols for a zone, follow these steps:

-
- Step 1** Log in to the CLI using an account with administrator privileges.
- Step 2** Enter anomaly detection external zone submode.
- ```
sensor# configure terminal
sensor(config)# service anomaly-detection ad0
sensor(config-ano)# external-zone
sensor(config-ano-ext)#
```
- Step 3** Enable the other protocols.
- ```
sensor(config-ano-ext)# other
sensor(config-ano-ext-oth)# enabled true
```
- Step 4** Associate a specific number for the other protocols.
- ```
sensor(config-ano-ext-oth)# protocol-number 5
sensor(config-ano-ext-oth-pro)#
```
- Step 5** Enable the service for that port.
- ```
sensor(config-ano-ext-oth-pro)# enabled true
```
- Step 6** Override the scanner values for that protocol. You can use the default scanner values, or you can override them and configure your own scanner values.
- ```
sensor(config-ano-ext-oth-pro)# override-scanner-settings yes
sensor(config-ano-ext-oth-pro-yes)#
```
- Step 7** Add a histogram for the new scanner settings. Enter the number of destination IP addresses (low, medium, or high) and the number of source IP addresses you want associated with this histogram.
- ```
sensor(config-ano-ext-oth-pro-yes)# threshold-histogram high num-source-ips 75
```
- Step 8** Set the scanner threshold.
- ```
sensor(config-ano-ext-oth-pro-yes)# scanner-threshold 100
```
- Step 9** Configure the default thresholds for all other unspecified ports.
- ```
sensor(config-ano-ext-oth-pro-yes)# exit
sensor(config-ano-ext-oth-pro)# exit
sensor(config-ano-ext-oth)# default-thresholds
sensor(config-ano-ext-oth-def)# default-thresholds
sensor(config-ano-ext-oth-def)# threshold-histogram medium num-source-ips 120
sensor(config-ano-ext-oth-def)# scanner-threshold 120
```
- Step 10** Verify the other protocols configuration settings.
- ```
sensor(config-ano-ext-oth)# show settings
other

protocol-number (min: 0, max: 255, current: 1)

number: 5

override-scanner-settings

yes

scanner-threshold: 95 default: 200
threshold-histogram (min: 0, max: 3, current: 1)
```

```

dest-ip-bin: high
num-source-ips: 75

enabled: true default: true

default-thresholds

scanner-threshold: 200 <defaulted>
threshold-histogram (min: 0, max: 3, current: 3)

<protected entry>
dest-ip-bin: low <defaulted>
num-source-ips: 10 <defaulted>
<protected entry>
dest-ip-bin: medium <defaulted>
num-source-ips: 1 <defaulted>
<protected entry>
dest-ip-bin: high <defaulted>
num-source-ips: 1 <defaulted>

enabled: true default: true

sensor(config-ano-ext-oth)#

```

## Configuring Learning Accept Mode

This section describes KBs and histograms and how to configure learning accept mode. It contains the following topics:

- [The KB and Histograms, page 9-36](#)
- [Configuring Learning Accept Mode, page 9-38](#)

## The KB and Histograms

The KB has a tree structure, and contains the following information:

- KB name
- Zone name
- Protocol
- Service

The KB holds a scanner threshold and a histogram for each service. If you have learning accept mode set to auto and the action set to rotate, a new KB is created every 24 hours and used in the next 24 hours. If you have learning accept mode set to auto and the action is set to save only, a new KB is created, but the current KB is used. If you do not have learning accept mode set to auto, no KB is created.

**Note**


---

Learning accept mode uses the sensor local time.

---

The scanner threshold defines the maximum number of zone IP addresses that a single source IP address can scan. The histogram threshold defines the maximum number of source IP addresses that can scan more than the specified numbers of zone IP addresses.

Anomaly detection identifies a worm attack when there is a deviation from the histogram that it has learned when no attack was in progress (that is, when the number of source IP addresses that concurrently scan more than the defined zone destination IP address is exceeded). For example, if the scanning threshold is 300 and the histogram for port 445, if anomaly detection identifies a scanner that scans 350 zone destination IP addresses, it produces an action indicating that a mass scanner was detected. However, this scanner does not yet verify that a worm attack is in progress. [Table 9-2](#) describes this example.

**Table 9-2 Example Histogram**

|                                    |    |    |     |
|------------------------------------|----|----|-----|
| Number of source IP addresses      | 10 | 5  | 2   |
| Number of destination IP addresses | 5  | 20 | 100 |

When anomaly detection identifies six concurrent source IP addresses that scan more than 20 zone destination IP addresses on port 445, it produces an action with an unspecified source IP address that indicates anomaly detection has identified a worm attack on port 445. The dynamic filter threshold, 20, specifies the new internal scanning threshold and causes anomaly detection to lower the threshold definition of a scanner so that anomaly detection produces additional dynamic filters for each source IP address that scans more than the new scanning threshold (20).

You can override what the KB learned per anomaly detection policy and per zone. If you understand your network traffic, you may want to use overrides to limit false positives.

#### Triggering the High Category Histogram Before the Single-Scanner Threshold

Based on the default histogram (nonlearned knowledge base [KB]) values, histogram-based detection can occur before single-scanner detection.

Single scanner detection is based on the scanner threshold settings. The scanner threshold setting is a single number for that port or protocol and zone. Any single IP address scanning more than that number of hosts of that port or protocol in that zone is alerted as a scanner.

There is a histogram for that port or protocol and zone that tracks how many systems normally scan a smaller number of hosts (10 hosts, 20 hosts, or 100 hosts). When more than that normal number of scanners are seen, then a worm is declared and all IPs scanning more than the associated number of hosts are alerted on as being a worm scanner.

**Note**


---

An IP source address can be alerted on as being a worm scanner without ever reaching the scanner threshold. The scanner threshold is used to detect single systems scanning a large number of hosts and is tracked separately from the algorithms for detecting worms.

---

## Configuring Learning Accept Mode

Use the **learning-accept-mode** command in service anomaly detection submode to configure whether you want the sensor to create a new KB every so many hours. You can configure whether the KB is created and loaded (rotate) or saved (save only). You can schedule how often and when the KB is loaded or saved.

The new updated KB file name is the current date and time, *YYYY-Mon-dd-hh\_mm\_ss*, where *Mon* is the three-letter abbreviation of the month.



### Note

Anomaly detection learning accept mode uses the sensor local time.

The following commands apply:

- **learning-accept-mode**—Specifies if and when the KB is saved and loaded:
  - **auto**— Configures the sensor to automatically accept the KB.
  - **manual**—Does not save the KB.



### Note

You can save and load the KB using the **anomaly-detection {load | save}** commands.

- **action**—Specifies whether to rotate or save the KB:
  - **save-only**—Saves the new KB. You can examine it and decide whether to load it into anomaly detection.
- **rotate**—Saves the new KB and loads it as the current KB according to the schedule you define.
- **schedule**— Configures a schedule to accept the KB:
  - **calendar-schedule {days-of-week} {times-of-day}**—Starts learning accept mode at specific times on specific days.
  - **periodic-schedule {interval} {start-time}**—Starts learning accept mode at specific periodic intervals.



### Note

You can load the KB using the **anomaly-detection load** command.

### Configuring Learning Accept Mode

The first saving begins after a full interval between configuration time and start time. For example, if the time is now 16:00 and you configure start time at 16:30 with an interval of one hour, the first KB is saved at 17:30, because there was no one-hour interval between 16:00 and 16:30.

To configure learning accept mode, follow these steps:

**Step 1** Log in to the CLI using an account with administrator privileges.

**Step 2** Enter anomaly detection submode.

```
sensor# configure terminal
sensor(config)# service anomaly-detection ad1
```

**Step 3** Specify how the KB is saved and loaded:

- a. Specify that the KB is automatically saved and loaded. Go to Step 4.

```
sensor(config-ano)# learning-accept-mode auto
sensor(config-ano-aut)#
```

- b. Specify that the KB is going to be manually saved and loaded. Go to Step 6.

```
sensor(config-ano)# learning-accept-mode manual
sensor(config-ano-man)#
```

**Step 4** Specify how you want the KB automatically accepted:

- a. Save the KB so that you can inspect it and decide whether to load it. Go to Step 6.

```
sensor(config-ano-aut)# action save-only
```

- b. Have the KB saved and loaded as the current KB according to the schedule you define. Continue with Step 5.

```
sensor(config-ano-aut)# action rotate
```

**Step 5** Schedule the automatic KB saves and loads:

- Calendar schedule—With this schedule the KB is saved and loaded every Monday at midnight.

```
sensor(config-ano-aut)# schedule calendar-schedule
sensor(config-ano-aut-cal)# days-of-week monday
sensor(config-ano-aut-cal)# times-of-day time 24:00:00
```

- Periodic schedule—With this schedule the KB is saved and loaded every 24 hours at midnight.

```
sensor(config-ano-aut)# schedule periodic-schedule
sensor(config-ano-aut-per)# start-time 24:00:00
sensor(config-ano-aut-per)# interval 24
```

**Step 6** Verify the settings.

```
sensor(config-ano-aut-per)# exit
sensor(config-ano-aut)# show settings
auto

 action: rotate default: rotate
 schedule

 periodic-schedule

 start-time: 12:00:00 default: 10:00:00
 interval: 24 hours default: 24


```

**Step 7** Exit anomaly detection submode.

```
sensor(config-ano-aut)# exit
sensor(config-ano)# exit
Apply Changes:[yes]:
```

**Step 8** Press **Enter** to apply your changes or enter **no** to discard them.

---

**For More Information**

For the procedures for saving and loading anomaly detection KBs manually, see [Saving and Loading KBs Manually, page 9-41](#).

## Working With KB Files

This section describes how to display, load, save, copy, rename and delete KB files. It also provides the procedures for comparing two KB files and for displaying the thresholds of a KB file. It contains the following topics:

- [Displaying KB Files, page 9-40](#)
- [Saving and Loading KBs Manually, page 9-41](#)
- [Copying, Renaming, and Erasing KBs, page 9-42](#)
- [Displaying the Differences Between Two KBs, page 9-44](#)
- [Displaying the Thresholds for a KB, page 9-45](#)

## Displaying KB Files

Use the **show ad-knowledge-base [virtual-sensor] files** command in privileged EXEC mode to display the available KB files for a virtual sensor.

**Note**

The \* before the file name indicates that this KB file is the currently loaded KB file.

To display KB files, follow these steps:

- 
- Step 1** Log in to the CLI.
- Step 2** Display the KB files for all virtual sensors.

```
sensor# show ad-knowledge-base files
Virtual Sensor vs0
 Filename Size Created
 initial 84 04:27:07 CDT Wed Jan 29 2003
* 2003-Jan-28-10_00_01 84 04:27:07 CDT Wed Jan 29 2003
Virtual Sensor vs1
 Filename Size Created
 initial 84 14:35:38 CDT Tue Mar 14 2006
 2006-Mar-16-10_00_00 84 10:00:00 CDT Thu Mar 16 2006
 2006-Mar-17-10_00_00 84 10:00:00 CDT Fri Mar 17 2006
 2006-Mar-18-10_00_00 84 10:00:00 CDT Sat Mar 18 2006
 2006-Mar-19-10_00_00 84 10:00:00 CDT Sun Mar 19 2006
 2006-Mar-20-10_00_00 84 10:00:00 CDT Mon Mar 20 2006
 2006-Mar-21-10_00_00 84 10:00:00 CDT Tue Mar 21 2006
 2006-Mar-22-10_00_00 84 10:00:00 CDT Wed Mar 22 2006
 2006-Mar-23-10_00_00 84 10:00:00 CDT Thu Mar 23 2006
 2006-Mar-24-10_00_00 84 10:00:00 CDT Fri Mar 24 2006
 2006-Mar-25-10_00_00 84 10:00:00 CDT Sat Mar 25 2006
 2006-Mar-26-10_00_00 84 10:00:00 CDT Sun Mar 26 2006
 2006-Mar-27-10_00_00 84 10:00:00 CDT Mon Mar 27 2006
 2003-Jan-02-10_00_00 84 10:00:00 CDT Thu Jan 02 2003
 2003-Jan-03-10_00_00 84 10:00:00 CDT Fri Jan 03 2003
 2003-Jan-04-10_00_00 84 10:00:00 CDT Sat Jan 04 2003
```



```

2003-Jan-05-10_00_00 84 10:00:00 CDT Sun Jan 05 2003
2003-Jan-06-10_00_00 84 10:00:00 CDT Mon Jan 06 2003
sensor#

```

**Step 3** Display the KB files for a specific virtual sensor.

```

sensor# show ad-knowledge-base vs0 files
Virtual Sensor vs0
 Filename Size Created
 initial 84 10:24:58 CDT Tue Mar 14 2006
 2006-Mar-16-10_00_00 84 10:00:00 CDT Thu Mar 16 2006
 2006-Mar-17-10_00_00 84 10:00:00 CDT Fri Mar 17 2006
 2006-Mar-18-10_00_00 84 10:00:00 CDT Sat Mar 18 2006
 2006-Mar-19-10_00_00 84 10:00:00 CDT Sun Mar 19 2006
 2006-Mar-20-10_00_00 84 10:00:00 CDT Mon Mar 20 2006

```

## Saving and Loading KBs Manually

Use these commands in privileged EXEC mode to manually save and load KBs.

The following commands apply:

- **show ad-knowledge-base virtual-sensor files**—Displays the available KB files per virtual sensor.
- **anomaly-detection virtual-sensor load {initial | file name}**—Sets the KB file as the current KB for the specified virtual sensor. If AD is active, the file is loaded as the current KB.
- **anomaly-detection virtual-sensor save [new-name]**—Retrieves the current KB file and saves it locally.

### Manually Saving and Loading KBs

To manually save and load a KB, follow these steps:

**Step 1** Log in to the CLI using an account with administrator privileges.

**Step 2** Locate the KB you want to load.

```

sensor# show ad-knowledge-base vs0 files
Virtual Sensor vs0
 Filename Size Created
 initial 84 10:24:58 CDT Tue Mar 14 2006
 2006-Mar-16-10_00_00 84 10:00:00 CDT Thu Mar 16 2006
 2006-Mar-17-10_00_00 84 10:00:00 CDT Fri Mar 17 2006
 2006-Mar-18-10_00_00 84 10:00:00 CDT Sat Mar 18 2006
 2006-Mar-19-10_00_00 84 10:00:00 CDT Sun Mar 19 2006
 2006-Mar-20-10_00_00 84 10:00:00 CDT Mon Mar 20 2006

```

**Step 3** Load the KB file as the current KB file for a specific virtual sensor.

```

sensor# anomaly-detection vs0 load file 2006-Mar-16-10_00_00
sensor#

```

**Step 4** Save the current KB file and store it as a new name.

```

sensor# anomaly-detection vs0 save my-KB
sensor#

```




---

**Note** An error is generated if anomaly detection is not active when you enter this command. You cannot overwrite the initial file.

---

## Copying, Renaming, and Erasing KBs

Use these commands in privileged EXEC mode to manually copy, rename, and erase KB files.

The following commands apply:

- **copy ad-knowledge-base** *virtual-sensor* { **current** | **initial** | **file name** } *destination-url*—Copies the KB file (current, initial, or the file name you enter) to a specified destination URL.




---

**Note** Copying a file to a name that already exists overwrites it.

---

- **copy ad-knowledge-base** *virtual-sensor source-url new-name*—Copies a KB with a new file name to the source URL you specify.




---

**Note** You cannot use the **current** keyword as a *new-name*. A new current KB file is created with the **load** command.

---

- **rename ad-knowledge-base** *virtual-sensor* { **current** | **file name** } *new-name*—Renames an existing KB file.
- **erase ad-knowledge-base** [*virtual-sensor [name]*]—Removes all KB files from a virtual sensor, or just one KB file if you use the *name* option.

You cannot erase the initial KB file or the KB file loaded as the current KB. The exact format of the source and destination URLs varies according to the file. Here are the valid types:

- **ftp:**—Source or destination URL for an FTP network server. The syntax for this prefix is:  
 ftp://[[username@] location]/relativeDirectory/filename  
 ftp://[[username@]location]//absoluteDirectory/filename
- **scp:**—Source or destination URL for the SCP network server. The syntax for this prefix is:  
 scp://[[username@] location]/relativeDirectory/filename  
 scp://[[username@] location]//absoluteDirectory/filename




---

**Note** If you use FTP or SCP protocol, you are prompted for a password. If you use SCP protocol, you must add the remote host to the SSH known hosts list.

---

- **http:**—Source URL for the web server. The syntax for this prefix is:  
 http://[[username@]location]/directory/filename
- **https:**—Source URL for the web server. The syntax for this prefix is:  
 https://[[username@]location]/directory/filename



**Note** If you use HTTPS protocol, the remote host must be a TLS trusted host.

### Copying, Renaming, and Removing KB Files

To copy, rename, and remove KB files, follow these steps:

**Step 1** Log in to the CLI using an account with administrator privileges.

**Step 2** Locate the KB file you want to copy.

```
sensor# show ad-knowledge-base vs0 files
Virtual Sensor vs0
 Filename Size Created
 ----- -
 initial 84 10:24:58 CDT Tue Mar 14 2006
 2006-Mar-16-10_00_00 84 10:00:00 CDT Thu Mar 16 2006
 2006-Mar-17-10_00_00 84 10:00:00 CDT Fri Mar 17 2006
 2006-Mar-18-10_00_00 84 10:00:00 CDT Sat Mar 18 2006
 2006-Mar-19-10_00_00 84 10:00:00 CDT Sun Mar 19 2006
 2006-Mar-20-10_00_00 84 10:00:00 CDT Mon Mar 20 2006
```

**Step 3** Copy the KB file to a user on a computer with the IP address 10.1.1.1.

```
sensor# copy ad-knowledge-base vs0 file 2006-Mar-16-10_00_00
scp://cidsuser@10.1.1.1/AD/my-KB
password: *****
sensor#
```

**Step 4** Rename a KB file.

```
sensor# rename ad-knowledge-base vs0 2006-Mar-16-10_00_00 My-KB
sensor#
```

**Step 5** Remove a KB file from a specific virtual sensor.

```
sensor# erase ad-knowledge-base vs0 2006-Mar-16-10_00_00
sensor#
```

**Step 6** Remove all KB files except the file loaded as current and the initial KB file from a virtual sensor.

```
sensor# erase ad-knowledge-base vs0
Warning: Executing this command will delete all virtual sensor 'vs0' knowledge bases
except the file loaded as current and the initial knowledge base.
Continue with erase? [yes]: yes
sensor#
```

**Step 7** Remove all KB files except the file loaded as current and the initial KB file from all virtual sensors.

```
sensor# erase ad-knowledge-base
Warning: Executing this command will delete all virtual sensor knowledge bases except the
file loaded as current and the initial knowledge base.
Continue with erase? [yes]: yes
sensor#
```

### For More Information

- For the procedure for creating a new KB using the **load** command, see [Saving and Loading KBs Manually, page 9-41](#).
- For the procedure for adding hosts to the SSH known hosts list, see [Adding Hosts to the SSH Known Hosts List, page 4-47](#).

- For the procedure for adding TLS trusted hosts, see [Adding TLS Trusted Hosts, page 4-53](#).

## Displaying the Differences Between Two KBs

Use the `show ad-knowledge-base virtual-sensor diff {current | initial | file name1} {current | initial | file name2} [diff-percentage]` command in privileged EXEC mode to display the differences between two KBs.

The following commands apply:

- *virtual-sensor*—Specifies the name of the virtual sensor that contains the KB files you want to compare.
- *name1*—Specifies the name of the first existing KB file to compare.
- *name2*—Specifies the name of the second existing KB file to compare.
- **current**—Specifies the currently loaded KB.
- **initial**—Specifies the initial KB.
- **file**—Specifies the name of an existing KB file.
- *diff-percentage*—(Optional) Displays the services where the thresholds differ more than the specified percentage. The valid values are 1 to 100. The default is 10%.

### Comparing Two KBs

To compare two KBs, follow these steps:

---

**Step 1** Log in to the CLI.

**Step 2** Locate the file you want to compare.

```
sensor# show ad-knowledge-base vs0 files
Virtual Sensor vs0
 Filename Size Created
 initial 84 04:27:07 CDT Wed Jan 29 2003
* 2006-Jun-28-10_00_01 84 04:27:07 CDT Thu Jun 29 2006
sensor#
```

**Step 3** Compare the currently loaded file (the file with the \*) with the initial KB for virtual sensor vs0.

```
sensor# show ad-knowledge-base vs0 diff initial file 2006-Jun-28-10_00_01
Initial Only Services/Protocols
 External Zone
 TCP Services
 Service = 30
 Service = 20
 UDP Services
 None
 Other Protocols
 Protocol = 1
 Illegal Zone
 None
 Internal Zone
 None
2006-Jun-28-10_00_01 Only Services/Protocols
 External Zone
 None
 Illegal Zone
 None
 Internal Zone
```

```

None
Thresholds differ more than 10%
External Zone
None
Illegal Zone
TCP Services
 Service = 31
 Service = 22
UDP Services
None
Other Protocols
 Protocol = 3
Internal Zone
None
sensor#

```

---

## Displaying the Thresholds for a KB

Use the **show ad-knowledge-base virtual-sensor thresholds** {**current** | **initial** | **file name**} [**zone** {**external** | **illegal** | **internal**}] {[**protocol** {**tcp** | **udp**}] [**dst-port port**] | [**protocol other**] [**number protocol-number**]} command in privileged EXEC mode to display the thresholds in a KB.

The following commands apply:

- **virtual-sensor**—Specifies the name of the virtual sensor that contains the KB files you want to compare.
- **name**—Specifies the name of the existing KB file.
- **current**—Specifies the currently loaded KB.
- **initial**—Specifies the initial KB.
- **file**—Specifies the name of an existing KB file.
- **zone**—(Optional) Displays the thresholds for the specified zone. The default displays information for all zones.
- **external**—Displays the thresholds for the external zone.
- **illegal**—Displays the thresholds for the illegal zone.
- **internal**—Displays the thresholds for the internal zone.
- **protocol**—(Optional) Displays the thresholds for the specified protocol. The default displays information about all protocols.
- **tcp**—Displays the thresholds for the TCP protocol.
- **udp**—Displays the thresholds for the UDP protocol.
- **other**—Displays the thresholds for the other protocols besides TCP or UDP.
- **dst-port**—(Optional) Displays thresholds for the specified port. The default displays information about all TCP and/or UDP ports.
- **port**—Specifies the port number. The valid values are 0 to 65535.
- **number**—(Optional) Displays thresholds for the specified other protocol number. The default displays information for all other protocols.
- **protocol-number**—Specifies the protocol number. The valid values are 0 to 255.

### Displaying KB Thresholds

To display the KB thresholds, follow these steps:

**Step 1** Log in to the CLI.

**Step 2** Locate the file for which you want to display thresholds:

```
sensor# show ad-knowledge-base vs1 files
Virtual Sensor vs1
 Filename Size Created
 initial 84 10:24:58 CDT Tue Mar 14 2006
 2006-Mar-16-10_00_00 84 10:00:00 CDT Thu Mar 16 2006
 2006-Mar-17-10_00_00 84 10:00:00 CDT Fri Mar 17 2006
 2006-Mar-18-10_00_00 84 10:00:00 CDT Sat Mar 18 2006
 2006-Mar-19-10_00_00 84 10:00:00 CDT Sun Mar 19 2006
 2006-Mar-27-10_00_00 84 10:00:00 CDT Mon Mar 27 2006
 2006-Apr-24-05_00_00 88 05:00:00 CDT Mon Apr 24 2006
 * 2006-Apr-25-05_00_00 88 05:00:00 CDT Tue Apr 25 2006
```

**Step 3** Display thresholds contained in a specific file for the illegal zone.

```
sensor# show ad-knowledge-base vs0 thresholds file 2006-Nov-11-10_00_00 zone illegal

AD Thresholds
 Creation Date = 2006-Nov-11-10_00_00
 KB = 2006-Nov-11-10_00_00
 Illegal Zone
 TCP Services
 Default
 Scanner Threshold
 User Configuration = 200
 Threshold Histogram - User Configuration
 Low = 10
 Medium = 3
 High = 1
 UDP Services
 Default
 Scanner Threshold
 User Configuration = 200
 Threshold Histogram - User Configuration
 Low = 10
 Medium = 3
 High = 1
 Other Services
 Default
 Scanner Threshold
 User Configuration = 200
 Threshold Histogram - User Configuration
 Low = 10
 Medium = 3
 High = 1
sensor#
```

**Step 4** Display thresholds contained in the current KB illegal zone, protocol TCP, and destination port 20.

```
sensor# show ad-knowledge-base vs0 thresholds current zone illegal protocol tcp dst-port 20

AD Thresholds
 Creation Date = 2006-Nov-14-10_00_00
 KB = 2006-Nov-14-10_00_00
 Illegal Zone
 TCP Services
```

```

Default
Scanner Threshold
 User Configuration = 200
Threshold Histogram - User Configuration
 Low = 10
 Medium = 3
 High = 1
sensor#

```

**Step 5** Display thresholds contained in the current KB illegal zone, and protocol other.

```

sensor# show ad-knowledge-base vs0 thresholds current zone illegal protocol other

AD Thresholds
Creation Date = 2006-Nov-14-10_00_00
KB = 2006-Nov-14-10_00_00
Illegal Zone
 Other Services
 Default
 Scanner Threshold
 User Configuration = 200
 Threshold Histogram - User Configuration
 Low = 10
 Medium = 3
 High = 1
sensor#

```

## Displaying Anomaly Detection Statistics

Use the **show statistics anomaly-detection** [*virtual-sensor-name*] command in privileged EXEC mode to display the statistics for anomaly detection. You can see if an attack is in progress (*Attack in progress* or *No attack*). You can also see when the next KB will be saved (*Next KB rotation at 10:00:00 UTC Wed Apr 26 2006*).



### Note

The **clear** command is not available for anomaly detection statistics.

To display anomaly detection statistics, follow these steps:

**Step 1** Log in to the CLI.

**Step 2** Display the anomaly detection statistics for a specific virtual sensor.

```

sensor# show statistics anomaly-detection vs0
Statistics for Virtual Sensor vs0
No attack
Detection - ON
Learning - ON
Next KB rotation at 10:00:00 UTC Wed Apr 26 2006
Internal Zone
 TCP Protocol
 UDP Protocol
 Other Protocol
External Zone
 TCP Protocol
 UDP Protocol
 Other Protocol
Illegal Zone

```

```

 TCP Protocol
 UDP Protocol
 Other Protocol
sensor#

```

**Step 3** Display the statistics for all virtual sensors.

```

sensor# show statistics anomaly-detection
Statistics for Virtual Sensor vs0
 No attack
 Detection - ON
 Learning - ON
 Next KB rotation at 10:00:01 UTC Wed Jun 29 2006
 Internal Zone
 TCP Protocol
 UDP Protocol
 Other Protocol
 External Zone
 TCP Protocol
 UDP Protocol
 Other Protocol
 Illegal Zone
 TCP Protocol
 UDP Protocol
 Other Protocol
Statistics for Virtual Sensor vs1
 No attack
 Detection - ON
 Learning - ON
 Next KB rotation at 10:00:00 UTC Wed Jul 29 2006
 Internal Zone
 TCP Protocol
 UDP Protocol
 Other Protocol
 External Zone
 TCP Protocol
 UDP Protocol
 Other Protocol
 Illegal Zone
 TCP Protocol
 UDP Protocol
 Other Protocol
sensor#

```

---

## Disabling Anomaly Detection

If you have anomaly detection enabled and you have your sensor configured to see only one direction of traffic, you should disable anomaly detection. Otherwise, you will receive many alerts, because anomaly detection sees asymmetric traffic as having incomplete connections, that is, like worm scanners, and fires alerts.



To disable anomaly detection, follow these steps:

---

**Step 1** Log in to the CLI using an account with administrator privileges.

**Step 2** Enter analysis engine submode.

```
sensor# configure terminal
sensor(config)# service analysis-engine
sensor(config-ana)#
```

**Step 3** Enter the virtual sensor name that contains the anomaly detection policy you want to disable.

```
sensor(config-ana)# virtual-sensor vs0
sensor(config-ana-vir)#
```

**Step 4** Disable anomaly detection operational mode.

```
sensor(config-ana-vir)# anomaly-detection
sensor(config-ana-vir-ano)# operational-mode inactive
sensor(config-ana-vir-ano)#
```

**Step 5** Exit analysis engine submode.

```
sensor(config-ana-vir-ano)# exit
sensor(config-ana-vir)# exit
sensor(config-ana-)# exit
Apply Changes:[yes]:
```

**Step 6** Press **Enter** to apply your changes or enter **no** to discard them.

---

#### For More Information

For more information about how worms operate, see [Understanding Worms](#), page 9-2.





## Configuring Global Correlation

---

This chapter provides information for configuring global correlation. It contains the following sections:

- [Global Correlation Notes and Caveats, page 10-1](#)
- [Understanding Global Correlation, page 10-2](#)
- [Participating in the SensorBase Network, page 10-2](#)
- [Understanding Reputation, page 10-3](#)
- [Understanding Network Participation, page 10-4](#)
- [Understanding Efficacy, page 10-5](#)
- [Understanding Reputation and Risk Rating, page 10-6](#)
- [Global Correlation Features and Goals, page 10-6](#)
- [Global Correlation Requirements, page 10-7](#)
- [Understanding Global Correlation Sensor Health Metrics, page 10-8](#)
- [Configuring Global Correlation Inspection and Reputation Filtering, page 10-8](#)
- [Configuring Network Participation, page 10-11](#)
- [Troubleshooting Global Correlation, page 10-13](#)
- [Disabling Global Correlation, page 10-13](#)
- [Displaying Global Correlation Statistics, page 10-14](#)

### Global Correlation Notes and Caveats

The following notes and caveats apply to configuring global correlation:

- As with signature updates, when the sensor applies a global correlation update, it may trigger bypass. Whether or not bypass is triggered depends on the traffic load of the sensor and the size of the signature/global correlation update. If bypass mode is turned off, an inline sensor stops passing traffic while the update is being applied.
- Network participation requires a network connection to the Internet.
- Valid license—You must have a valid sensor license for automatic signature updates and global correlation features to function. You can still configure and display statistics for the global correlation features, but the global correlation databases are cleared and no updates are attempted. Once you install a valid license, the global correlation features are reactivated.

- Global correlation inspection and the reputation filtering deny features do not support IPv6 addresses. For global correlation inspection, the sensor does not receive or process reputation data for IPv6 addresses. The risk rating for IPv6 addresses is not modified for global correlation inspection. Similarly, network participation does not include event data for attacks from IPv6 addresses. And finally, IPv6 addresses do not appear in the deny list.
- The sensor must operate in inline mode so that the global correlation features can increase efficacy by being able to use the inline deny actions.
- For global correlation to function, you must have either a DNS server or an HTTP proxy server configured at all times.

## Understanding Global Correlation

You can configure global correlation so that your sensors are aware of network devices with a reputation for malicious activity, and can take action against them. Participating IPS devices in a centralized Cisco threat database, the SensorBase Network, receive and absorb global correlation updates. The reputation information contained in the global correlation updates is factored in to the analysis of network traffic, which increases IPS efficacy, since traffic is denied or allowed based on the reputation of the source IP address. The participating IPS devices send data back to the Cisco SensorBase Network, which results in a feedback loop that keeps the updates current and global.

You can configure the sensor to participate in the global correlation updates and/or in sending telemetry data or you can turn both services off. You can view reputation scores in events and see the reputation score of the attacker. You can also view statistics from the reputation filter.

## Participating in the SensorBase Network

The Cisco IPS contains a security capability, Cisco Global Correlation, which uses the immense security intelligence that we have amassed over the years. At regular intervals, the Cisco IPS receives threat updates from the Cisco SensorBase Network, which contain detailed information about known threats on the Internet, including serial attackers, Botnet harvesters, Malware outbreaks, and dark nets. The IPS uses this information to filter out the worst attackers before they have a chance to attack critical assets. It then incorporates the global threat data in to its system to detect and prevent malicious activity even earlier.

If you agree to participate in the SensorBase Network, Cisco will collect aggregated statistics about traffic sent to your IPS. This includes summary data on the Cisco IPS network traffic properties and how this traffic was handled by the Cisco appliances. We do not collect the data content of traffic or other confidential business or personal information. All data is aggregated and sent by secure HTTP to the Cisco SensorBase Network servers in periodic intervals. All data shared with Cisco will be anonymous and treated as strictly confidential.

Table 10-1 shows how we use the data.

**Table 10-1 Cisco Network Participation Data Use**

| Participation Level | Type of Data                                                                                    | Purpose                                                              |
|---------------------|-------------------------------------------------------------------------------------------------|----------------------------------------------------------------------|
| Partial             | Protocol attributes<br>(TCP maximum segment size and options string, for example)               | Tracks potential threats and helps us to understand threat exposure. |
|                     | Attack type<br>(signature fired and risk rating, for example)                                   | Used to understand current attacks and attack severity.              |
|                     | Connecting IP address and port                                                                  | Identifies attack source.                                            |
|                     | Summary IPS performance<br>(CPU utilization, memory usage, inline vs. promiscuous, for example) | Tracks product efficacy.                                             |
| Full                | Victim IP address and port                                                                      | Detects threat behavioral patterns.                                  |

When you enable Partial or Full Network Participation, the Network Participation Disclaimer appears. You must enter **yes** to participate. If you do not have a license installed, you receive a warning telling you that global correlation inspection and reputation filtering are disabled until the sensor is licensed. You can obtain a license at <http://www.cisco.com/go/license>.

#### For More Information

For information on how to obtain and install a sensor license, see [Installing the License Key, page 4-58](#).

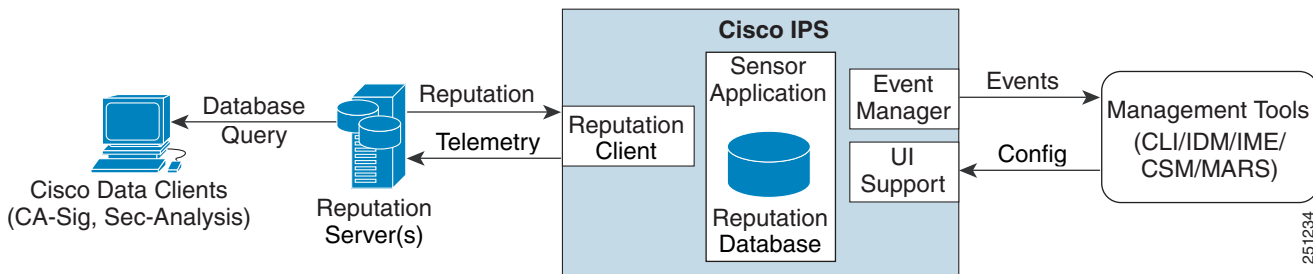
## Understanding Reputation

Similar to human social interaction, reputation is an opinion toward a device on the Internet. It enables the installed base of IPS sensors in the field to collaborate using the existing network infrastructure. A network device with reputation is most likely either malicious or infected. You can view reputation information and statistics in the IDM, IME, or the CLI.

The IPS sensor collaborates with the global correlation servers (also known as reputation servers) to improve the efficacy of the sensor.

Figure 10-1 shows the role of the sensor and the global correlation servers.

**Figure 10-1** IPS Management and Global Correlation Server Interaction



The global correlation servers provide information to the sensor about certain IP addresses that may identify malicious or infected hosts. The sensor uses this information to determine which actions, if any, to perform when potentially harmful traffic is received from a host with known reputation. Because the global correlation database changes rapidly, the sensor must periodically download global correlation updates from the global correlation servers.



#### Caution

As with signature updates, when the sensor applies a global correlation update, it may trigger bypass. Whether or not bypass is triggered depends on the traffic load of the sensor and the size of the signature/global correlation update. If bypass mode is turned off, an inline sensor stops passing traffic while the update is being applied.

#### For More Information

For more information about viewing global correlation statistics, see [Displaying Statistics](#), page 17-30.

## Understanding Network Participation

Network participation lets us collect nearly real-time data from sensors around the world. Sensors installed at customer sites can send data to the SensorBase Network. These data feed in to the global correlation database to increase reputation fidelity. Communication between sensors and the SensorBase Network involves an HTTPS request and response over TCP/IP. Network participation gathers the following data:

- Signature ID
- Attacker IP address
- Attacker port
- Maximum segment size
- Victim IP address
- Victim port
- Signature version
- TCP options string
- Reputation score
- Risk rating

- Data gathered from the sensor health metrics

The statistics for network participation show the hits and misses for alerts, the reputation actions, and the counters of packets that have been denied.

**Note**

---

Network participation requires a network connection to the Internet.

---

There are three modes for network participation:

- Off—The network participation server does not collect data, track statistics, or try to contact the Cisco SensorBase Network.
- Partial Participation—The network participation server collects data, tracks statistics, and communicates with the SensorBase Network. Data considered to be potentially sensitive is filtered out and never sent.
- Full Participation—The network participation server collects data, tracks statistics, and communicates with the SensorBase Network. All data collected is sent except the IP addresses that you exclude from the network participation data.

**Caution**

---

As with signature updates, when the sensor applies a global correlation update, it may trigger bypass. Whether or not bypass is triggered depends on the traffic load of the sensor and the size of the signature/global correlation update. If bypass mode is turned off, an inline sensor stops passing traffic while the update is being applied.

---

**For More Information**

- For more information on network participation, see [Configuring Network Participation, page 10-11](#).
- For more information on bypass mode, see [Configuring Inline Bypass Mode, page 5-42](#).

## Understanding Efficacy

Obtaining data from participating IPS clients and using that in conjunction with the existing corpus of threat knowledge improves the efficacy of the IPS. We measure efficacy based on the following:

- False positives as a percentage of actionable events
- False negatives as a percentage of threats that do not result in actionable events
- Actionable events as a percentage of all events

The IPS signature team uses the data to improve signature fidelity and the IPS engineering team uses the data to better understand the various types of sensor deployment.

**For More Information**

For more information about reputation and risk rating, see [Understanding Reputation and Risk Rating, page 10-6](#).

# Understanding Reputation and Risk Rating

Risk rating is the concept of the probability that a network event is malicious. You assign a numerical quantification of the risk associated with a particular event on the network. By default, an alert with an extreme risk rating shuts down traffic. Reputation indicates the probability that a particular attacker IP address will initiate malicious behavior based on its known past activity. A certain score is computed for this reputation by the Alarm Channel and added to risk rating, thus improving the efficacy of the IPS. When the attacker has a bad reputation score, an incremental risk is added to the risk rating to make it more aggressive.

The Alarm Channel handles signature events from the data path. The alert processing units have multiple aggregation techniques, action overrides, action filters, attacker reputation, and per-action custom handling methods. We use the large reputation data from the reputation participation client to score attackers in the Alarm Channel and then use this score to influence the risk rating and actions of the alert.

## For More Information

- For a detailed description of risk rating, see [Calculating the Risk Rating, page 8-13](#).
- For a detailed description of threat rating, see [Understanding Threat Rating, page 8-15](#).
- For a detailed description of event action filters, see [Configuring Event Action Filters, page 8-20](#).
- For a detailed description of the Alarm Channel, see [Understanding the SensorApp, page A-23](#).
- For a detailed description of event action aggregation, see [Understanding Event Action Aggregation, page 8-34](#).

# Global Correlation Features and Goals

There are three main features of global correlation:

- **Global Correlation Inspection**—We use the global correlation reputation knowledge of attackers to influence alert handling and deny actions when attackers with a bad score are seen on the sensor.
- **Reputation Filtering**—Applies automatic deny actions to packets from known malicious sites.
- **Network Reputation**—Sensor sends alert and TCP fingerprint data to the SensorBase Network.

Global correlation has the following goals:

- Dealing intelligently with alerts thus improving efficacy.
- Improving protection against known malicious sites.
- Sharing telemetry data with the SensorBase Network to improve visibility of alerts and sensor actions on a global scale.
- Simplifying configuration settings.
- Automatic handling of the uploads and downloads of the information.



# Global Correlation Requirements

Global correlation has the following requirements:

- Valid license—You must have a valid sensor license for automatic signature updates and global correlation features to function. You can still configure and display statistics for the global correlation features, but the global correlation databases are cleared and no updates are attempted. Once you install a valid license, the global correlation features are reactivated.
- Network Participation disclaimer—You must agree to the disclaimer to participate.
- External connectivity for the sensor and a DNS server—The global correlation features of Cisco IPS require the sensor to connect to the Cisco SensorBase Network. Domain name resolution is also required for these features to function. You can either configure the sensor to connect through an HTTP proxy server that has a DNS client running on it, or you can assign an Internet routeable address to the management interface of the sensor and configure the sensor to use a DNS server. In Cisco IPS the HTTP proxy and DNS servers are used only by the global correlation features.
- If you are connecting through an HTTP proxy, make sure you have the following configuration:
  - The proxy must allow HTTP requests from the IPS systems to `http://updates.ironport.com/ibrs/` on port 80.
  - The proxy must allow HTTPS requests from the IPS systems to `update-manifests.ironport.com` on port 443.
  - The firewall must allow access from the proxy to the internet (any destination address) on ports 80 and 443.
- If you are NOT connecting through the HTTP proxy:
  - The firewall must allow access from each IPS to the Internet (any destination address) on ports 80 and 443.



---

**Note** The IPS does not support the use of authenticated proxies.

---

- Sensors deployed in an environment with a slow command and control connection will be slow to download global correlation updates.
- No IPv6 address support—Global correlation inspection and the reputation filtering deny features do not support IPv6 addresses. For global correlation inspection, the sensor does not receive or process reputation data for IPv6 addresses. The risk rating for IPv6 addresses is not modified for global correlation inspection. Similarly, network participation does not include event data for attacks from IPv6 addresses. And finally, IPv6 addresses do not appear in the deny list.
- Sensor in inline mode—The sensor must operate in inline mode so that the global correlation features can increase efficacy by being able to use the inline deny actions.
- Sensor that supports the global correlation features
- IPS version that supports the global correlation features

## For More Information

- For information on how to obtain and install a sensor license, see [Installing the License Key, page 4-58](#).
- For information about the Network Participation disclaimer, see [Participating in the SensorBase Network, page 10-2](#).

- For information about configuring an HTTP proxy or DNS server to support global correlation, see [Configuring the DNS and Proxy Servers for Global Correlation and Automatic Update, page 4-10](#).

## Understanding Global Correlation Sensor Health Metrics

For global correlation, the following metrics are added to sensor health monitoring:

- Green indicates that the last update was successful.
- Yellow indicates that there has not been a successful update within the past day (86,400 seconds).
- Red indicates that there has not been a successful update within the last three days (259,200 seconds).

For network participation, the following metrics are added to sensor health monitoring:

- Green indicates that the last connection was successful.
- Yellow indicates that less than 6 connections failed in a row.
- Red indicates that more than 6 connections failed in a row.

Use the **health-monitor** command in service submode to configure the health statistics for the sensor. Use the **show health** command to see the results of the **health-monitor** command.

Global correlation health status defaults to red and changes to green after a successful global correlation update. Successful global correlation updates require a DNS server or an HTTP proxy server. Because DNS and HTTP proxy server configuration features are beginning with IPS 7.0(1)E3, they are unconfigured after an upgrade from 6.x to 7.0(1)E3 or higher. As a result, global correlation health and overall sensor health status are red until you configure a DNS or HTTP proxy server on the sensor. If the sensor is deployed in an environment where a DNS or HTTP proxy server is not available, you can address the red global correlation health and overall sensor health status by disabling global correlation and configuring sensor health status not to include global correlation health status.

### For More Information

- For more information about the sensor health metrics and how to enable/disable Global Correlation health status, see [Configuring Health Status Information, page 17-14](#).
- For the procedure to view sensor health metrics, see [Showing Sensor Overall Health Status, page 17-19](#).
- For information about configuring an HTTP proxy or DNS server to support global correlation, see [Configuring the DNS and Proxy Servers for Global Correlation and Automatic Update, page 4-10](#).
- For the procedure to disable global correlation, see [Configuring Global Correlation Inspection and Reputation Filtering, page 10-10](#).

## Configuring Global Correlation Inspection and Reputation Filtering

This section describes global correlation inspection and reputation filtering, and how to configure them. It contains the following topics:

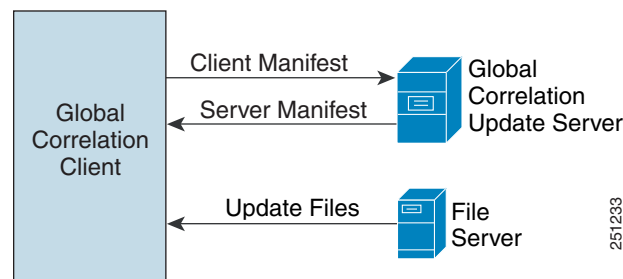
- [Understanding Global Correlation Inspection and Reputation Filtering, page 10-9](#)
- [Configuring Global Correlation Inspection and Reputation Filtering, page 10-10](#)

## Understanding Global Correlation Inspection and Reputation Filtering

You can configure the sensor to use updates from the SensorBase Network to adjust the risk rating. The client determines which updates are available and applicable to the sensor by communicating with the global correlation update server and a file server, which is a two-phase process. In the first phase the sensor sends a client manifest to the global correlation update server via an HTTPS POST request. The server then returns the server manifest document in the HTTPS response. In the next phase the sensor identifies the updates that are available and how to obtain them from a file server. The sensor downloads the encrypted update files via HTTP from the file server using the information in the server manifest. The integrity of these update files has been verified by comparing its MD5 hash with the hash value specified in the server manifest.

Figure 10-2 demonstrates how the global correlation update client obtains the files.

**Figure 10-2 Global Correlation Update Client**



### Caution

You must have a valid sensor license for automatic signature updates and global correlation features to function. You can still configure and display statistics for the global correlation features, but the global correlation databases are cleared and no updates are attempted. Once you install a valid license, the global correlation features are reactivated.

Once you configure global correlation, updates are automatic and happen at regular intervals, approximately every five minutes by default, but this interval may be modified by the global correlation server. The sensor gets a full update and then applies an incremental update periodically.

You configure an HTTP proxy or a DNS server in the service network-setting submode. If you turn on global correlation, you can choose how aggressively you want the deny actions to be enforced against malicious hosts. You can then enable reputation filtering to deny access to known malicious hosts. If you only want a report of what could have happened, you can enable **test-global-correlation**. This puts the sensor in audit mode, and actions the sensor would have performed are generated in the events.

Use the **show health** command in privileged EXEC mode to display the overall health status information of the sensor. The health status categories are rated by red and green with red being critical.



### Caution

As with signature updates, when the sensor applies a global correlation update, it may trigger bypass. Whether or not bypass is triggered depends on the traffic load of the sensor and the size of the signature/global correlation update. If bypass mode is turned off, an inline sensor stops passing traffic while the update is being applied.

**For More Information**

- For the procedure for configuring global correlation features, see [Configuring Global Correlation Inspection and Reputation Filtering, page 10-10](#).
- For the procedure to view sensor health metrics, see [Showing Sensor Overall Health Status, page 17-19](#).
- For information on the CollaborationApp, see [CollaborationApp, page A-27](#).
- For more information on bypass mode, see [Configuring Inline Bypass Mode, page 5-42](#).

## Configuring Global Correlation Inspection and Reputation Filtering

**Caution**

For automatic and global correlation updates to function, you must have either a DNS server or an HTTP proxy server configured at all times.

The following commands apply:

- **global-correlation-inspection {on | off}**—Turns global correlation inspection on or off. When turned on, the sensor uses updates from the SensorBase network to adjust the risk rating. The default is on.
- **global-correlation-inspection-influence {permissive | standard | aggressive}**—Lets you choose the level of global correlation inspection. The default is standard.
  - **permissive**—Global correlation data has little influence in the decision to deny traffic.
  - **standard**—Global correlation moderately influences the decision to deny traffic.
  - **aggressive**—Global correlation data heavily influences the decision to deny traffic.
- **reputation-filtering {on | off}**—Turns reputation filtering on or off. When turned on, the sensor denies access to malicious hosts that are listed in the global correlation database. The default is on.
- **test-global-correlation {on | off}**—Enables reporting of deny actions that are affected by global correlation. Allows you to test the global correlation features without actually denying any hosts. The default is off.

**Configuring Global Correlation**

To configure global correlation features, follow these steps:

**Step 1** Log in to the CLI using an account with administrator privileges.

**Step 2** Enter global correlation submode.

```
sensor# configure terminal
sensor(config)# service global-correlation
sensor(config-glo)#
```

**Step 3** Turn on global correlation inspection.

```
sensor(config-glo)# global-correlation-inspection on
sensor(config-glo)#
```

**Step 4** Specify the level of global correlation inspection.

```
sensor(config-glo)# global-correlation-inspection-influence aggressive
sensor(config-glo)#
```

**Step 5** Turn on reputation filtering.

```
sensor(config-glo)# reputation-filtering on
sensor(config-glo)#
```

**Step 6** Test global correlation data, but do not actually deny traffic.

```
sensor(config-glo)# test-global-correlation on
sensor(config-glo)#
```

**Step 7** Verify the settings.

```
sensor(config-glo)# show settings
 global-correlation-inspection: on default: on
 global-correlation-inspection-influence: aggressive default: standard
 reputation-filtering: on default: on
 test-global-correlation: on default: off
sensor(config-glo)#
```

**Step 8** Exit global correlation submode.

```
sensor(config-glo)# exit
Apply Changes:[yes]:
```

**Step 9** Press **Enter** to apply your changes or enter **no** to discard them.

#### For More Information

- For information about configuring a proxy or DNS server to support global correlation, see [Configuring the DNS and Proxy Servers for Global Correlation and Automatic Update, page 4-10](#).
- For information on how to obtain and install a sensor license, see [Installing the License Key, page 4-58](#).
- For more information about the sensor health metrics, see [Showing Sensor Overall Health Status, page 17-19](#).

## Configuring Network Participation

You can configure the sensor to send data to the SensorBase Network. You can configure the sensor to fully participate and send all data to the SensorBase Network. Or you can configure the sensor to collect the data but to omit potentially sensitive data, such as the destination IP address of trigger packets.



#### Note

Configuring the sensor for partial network participation limits a third party from extracting reconnaissance information about your internal network from the global correlation database.

The following option applies:

- **network-participation**—Sets the level of network participation. The default is off.
  - **off**—No data is contributed to the SensorBase network.
  - **partial**—Data is contributed to the SensorBase network but potentially sensitive information is withheld.
  - **full**—All data is contributed to the SensorBase network.



**Note** You must accept the network participation disclaimer to turn on network participation.

### Turning on Network Participation

To turn on network participation, follow these steps:

**Step 1** Log in to the CLI using an account with administrator privileges.

**Step 2** Enter global correlation submode.

```
sensor# configure terminal
sensor(config)# service global-correlation
sensor(config-glo)#
```

**Step 3** Turn on network participation.

```
sensor(config-glo)# network-participation [full | partial]
sensor(config-glo)# exit
```

**Step 4** Enter **yes** to agree to participate in the SensorBase Network.

If you agree to participate in the SensorBase Network, Cisco will collect aggregated statistics about traffic sent to your IPS. This includes summary data on the Cisco IPS network traffic properties and how this traffic was handled by the Cisco appliances. We do not collect the data content of traffic or other sensitive business or personal information. All data is aggregated and sent via secure HTTP to the Cisco SensorBase Network servers in periodic intervals. All data shared with Cisco will be anonymous and treated as strictly confidential.

The table below describes how the data will be used by Cisco.

Participation Level = "Partial":

- \* Type of Data: Protocol Attributes (e.g. TCP max segment size and options string)  
Purpose: Track potential threats and understand threat exposure
- \* Type of Data: Attack Type (e.g. Signature Fired and Risk Rating)  
Purpose: Used to understand current attacks and attack severity
- \* Type of Data: Connecting IP Address and port  
Purpose: Identifies attack source
- \* Type of Data: Summary IPS performance (CPU utilization memory usage, inline vs. promiscuous, etc)  
Purpose: Tracks product efficacy

Participation Level = "Full" additionally includes:

- \* Type of Data: Victim IP Address and port  
Purpose: Detect threat behavioral patterns

Do you agree to participate in the SensorBase Network?[no]:

**Step 5** Verify the settings.

```
sensor(config-glo)# show settings
network-participation: full default: off
global-correlation-inspection: on default: on
global-correlation-inspection-influence: aggressive default: standard
reputation-filtering: on default: on
test-global-correlation: on default: off
sensor(config-glo)#
```

**Step 6** Exit global correlation submode.

```
sensor(config-glo)# exit
Apply Changes:[yes]:
```

**Step 7** Press **Enter** to apply your changes or enter **no** to discard them.

---

#### For More Information

For more information about participating in the SensorBase Network, see [Participating in the SensorBase Network, page 10-2](#).

## Troubleshooting Global Correlation

Make sure you observe the following when configuring global correlation:

- Because global correlation updates occur through the sensor management interface, firewalls must allow port 443/80 traffic.
- You must have an HTTP proxy server or a DNS server configured to allow global correlation features to function.
- If you have an HTTP proxy server configured, the proxy must allow port 443/80 traffic from IPS systems.
- You must have a valid IPS license to allow global correlation features to function.
- Global correlation features only contain external IP addresses, so if you position a sensor in an internal lab, you may never receive global correlation information.
- Make sure your sensor supports the global correlation features.
- Make sure your IPS version supports the global correlation features.

#### For More Information

- For the procedure for configuring a DNS or HTTP proxy server, see [Configuring the DNS and Proxy Servers for Global Correlation and Automatic Update, page 4-10](#).
- For the procedure for obtaining an IPS license, see [Installing the License Key, page 4-58](#).

## Disabling Global Correlation

If your sensor is deployed in an environment where a DNS server or HTTP proxy server is not available, you may want to disable global correlation so that global correlation health does not appear as red in the overall sensor health, thus indicating a problem. You can also configure sensor health to exclude global correlation status.

The following commands apply:

- **global-correlation-inspection {on | off}**—Turns global correlation inspection on or off. When turned on, the sensor uses updates from the SensorBase network to adjust the risk rating. The default is on.
- **reputation-filtering {on | off}**—Turns reputation filtering on or off. When turned on, the sensor denies access to malicious hosts that are listed in the global correlation database. The default is on.
- **network-participation**—Sets the level of network participation. The default is off.
  - **off**—No data is contributed to the SensorBase network.
  - **partial**—Data is contributed to the SensorBase network but potentially sensitive information is withheld.

- **full**—All data is contributed to the SensorBase network.

### Disabling Global Correlation

To disable global correlation features, follow these steps:

---

**Step 1** Log in to the CLI using an account with administrator privileges.

**Step 2** Enter global correlation submode.

```
sensor# configure terminal
sensor(config)# service global-correlation
sensor(config-glo)#
```

**Step 3** Turn off global correlation inspection.

```
sensor(config-glo)# global-correlation-inspection off
sensor(config-glo)#
```

**Step 4** Turn off reputation filtering.

```
sensor(config-glo)# reputation-filtering off
sensor(config-glo)#
```

**Step 5** Turn off network participation.

```
sensor(config-glo)# network-participation off
sensor(config-glo)# exit
```

**Step 6** Verify the settings.

```
sensor(config-glo)# show settings
 network-participation: full default: off
 global-correlation-inspection: on default: off
 reputation-filtering: on default: off
sensor(config-glo)#
```

**Step 7** Exit global correlation submode.

```
sensor(config-glo)# exit
Apply Changes:[yes]:
```

**Step 8** Press **Enter** to apply your changes or enter **no** to discard them.

---

## Displaying Global Correlation Statistics

Use the **show statistics global-correlation** command to display global correlation statistics. Use the **show statistics global-correlation [name | clear]** command to display statistics for these components for all virtual sensors. If you provide the virtual sensor name, the statistics for that virtual sensor only are displayed.

To display and clear global correlation statistics for the sensor, follow these steps:

---

**Step 1** Log in to the CLI.

**Step 2** Display the statistics for global correlation.

```
sensor# show statistics global-correlation
```



```

Network Participation:
 Counters:
 Total Connection Attempts = 4347
 Total Connection Failures = 155
 Connection Failures Since Last Success = 0
 Connection History:
 Connection Attempt on June 17 2012, at 21:57:19 UTC = Successful
 Connection Attempt on June 17 2012, at 21:54:18 UTC = Successful
 Connection Attempt on June 17 2012, at 21:51:17 UTC = Successful
 Connection Attempt on June 17 2012, at 21:48:17 UTC = Successful
 Connection Attempt on June 17 2012, at 21:45:16 UTC = Successful
Updates:
 Status Of Last Update Attempt = Disabled
 Time Since Last Successful Update = never
 Counters:
 Update Failures Since Last Success = 0
 Total Update Attempts = 0
 Total Update Failures = 0
 Update Interval In Seconds = 300
 Update Server = update-manifests.ironport.com
 Update Server Address = Unknown
 Current Versions:
Warnings:
Details:
 Last fail log =
sensor#

```

### Step 3 Clear the statistics for global correlation:

```

sensor# show statistics global-correlation clear
Network Participation:
 Counters:
 Total Connection Attempts = 0
 Total Connection Failures = 0
 Connection Failures Since Last Success = 0
 Connection History:
 Connection Attempt on June 17 2012, at 22:03:20 UTC = Successful
 Connection Attempt on June 17 2012, at 22:00:19 UTC = Successful
 Connection Attempt on June 17 2012, at 21:57:19 UTC = Successful
 Connection Attempt on June 17 2012, at 21:54:18 UTC = Successful
 Connection Attempt on June 17 2012, at 21:51:17 UTC = Successful
Updates:
 Status Of Last Update Attempt = Disabled
 Time Since Last Successful Update = never
 Counters:
 Update Failures Since Last Success = 0
 Total Update Attempts = 0
 Total Update Failures = 0
 Update Interval In Seconds = 300
 Update Server = update-manifests.ironport.com
 Update Server Address = Unknown
 Current Versions:
Warnings:
Details:
 Last fail log =
sensor#

```

---





# Configuring External Product Interfaces

---

This chapter explains how to configure external product interfaces. It contains the following sections:

- [External Product Interface Notes and Caveats, page 11-1](#)
- [Understanding External Product Interfaces, page 11-1](#)
- [Understanding the CSA MC, page 11-2](#)
- [External Product Interface Issues, page 11-3](#)
- [Configuring the CSA MC to Support the IPS Interface, page 11-4](#)
- [Adding External Product Interfaces and Posture ACLs, page 11-4](#)
- [Troubleshooting External Product Interfaces, page 11-8](#)

## External Product Interface Notes and Caveats

The following notes and caveats apply to external product interfaces:

- In Cisco IPS, you can only add interfaces to the CSA MC.
- You can only enable two CSA MC interfaces.
- You must add the CSA MC as a trusted host so the sensor can communicate with it.

## Understanding External Product Interfaces



**Note**

---

In Cisco IPS, you can only add interfaces to the CSA MC.

---

The external product interface is designed to receive and process information from external security and management products. These external security and management products collect information that can be used to automatically enhance the sensor configuration information. For example, the types of information that can be received from external products include host profiles (the host OS configuration, application configuration, and security posture) and IP addresses that have been identified as causing malicious network activity.

# Understanding the CSA MC

The CSA MC enforces a security policy on network hosts. It has two components:

- Agents that reside on and protect network hosts.
- Management Console (MC)—An application that manages agents. It downloads security policy updates to agents and uploads operational information from agents.

The CSA MC receives host posture information from the CSA agents it manages. It also maintains a watch list of IP addresses that it has determined should be quarantined from the network. The CSA MC sends two types of events to the sensor—host posture events and quarantined IP address events.

Host posture events (called imported OS identifications in IPS) contain the following information:

- Unique host ID assigned by the CSA MC
- CSA agent status
- Host system hostname
- Set of IP addresses enabled on the host
- CSA software version
- CSA polling status
- CSA test mode status
- NAC posture

For example, when an OS-specific signature fires whose target is running that OS, the attack is highly relevant and the response should be greater. If the target OS is different, then the attack is less relevant and the response may be less critical. The signature attack relevance rating is adjusted for this host.

The quarantined host events (called the watch list in IPS) contain the following information:

- IP address
- Reason for the quarantine
- Protocol associated with a rule violation (TCP, UDP, or ICMP)
- Indicator of whether a rule-based violation was associated with an established session or a UDP packet.

For example, if a signature fires that lists one of these hosts as the attacker, it is presumed to be that much more serious. The risk rating is increased for this host. The magnitude of the increase depends on what caused the host to be quarantined.

The sensor uses the information from these events to determine the risk rating increase based on the information in the event and the risk rating configuration settings for host postures and quarantined IP addresses.

**Note**

The host posture and watch list IP address information is not associated with a virtual sensor, but is treated as global information.

Secure communications between the CSA MC and the IPS sensor are maintained through SSL/TLS. The sensor initiates SSL/TLS communications with the CSA MC. This communication is mutually authenticated. The CSA MC authenticates by providing X.509 certificates. The sensor uses username/password authentication.

**Note**

You can only enable two CSA MC interfaces.

**Caution**

You must add the CSA MC as a trusted host so the sensor can communicate with it.

**For More Information**

For the procedure for adding trusted hosts, see [Adding TLS Trusted Hosts, page 4-53](#).

## External Product Interface Issues

When the external product interface receives host posture and quarantine events, the following issues can arise:

- The sensor can store only a certain number of host records:
  - If the number of records exceeds 10,000, subsequent records are dropped.
  - If the 10,000 limit is reached and then it drops to below 9900, new records are no longer dropped.
- Hosts can change an IP address or appear to use another host IP address, for example, because of DHCP lease expiration or movement in a wireless network. In the case of an IP address conflict, the sensor presumes the most recent host posture event to be the most accurate.
- A network can include overlapping IP address ranges in different VLANs, but host postures do not include VLAN ID information. You can configure the sensor to ignore specified address ranges.
- A host can be unreachable from the CSA MC because it is behind a firewall. You can exclude unreachable hosts.
- The CSA MC event server allows up to ten open subscriptions by default. You can change this value. You must have an administrative account and password to open subscriptions.
- CSA data is not virtualized; it is treated globally by the sensor.
- Host posture OS and IP addresses are integrated into passive OS fingerprinting storage. You can view them as imported OS profiles.
- You cannot see the quarantined hosts.
- The sensor must recognize each CSA MC host X.509 certificate. You must add them as a trusted host.
- You can configure a maximum of two external product devices.

**For More Information**

- For more information on working with OS maps and identifications, see [Adding, Editing, Deleting, and Moving Configured OS Maps, page 8-28](#) and [Displaying and Clearing OS Identifications, page 8-32](#).
- For the procedure for adding trusted hosts, see [Adding TLS Trusted Hosts, page 4-53](#).

## Configuring the CSA MC to Support the IPS Interface


**Note**

For more detailed information about host posture events and quarantined IP address events, refer to [Using Management Center for Cisco Security Agents 5.1](#).

You must configure the CSA MC to send host posture events and quarantined IP address events to the sensor. To configure the CSA MC to support IPS interfaces, follow these steps:

**Step 1** Choose **Events > Status Summary**.

**Step 2** In the Network Status section, click **No** beside **Host history collection enabled**, and then click **Enable** in the popup window.


**Note**

Host history collection is enabled globally for the system. This feature is disabled by default because the MC log file tends to fill quickly when it is turned on.

**Step 3** Choose **Systems > Groups** to create a new group (with no hosts) to use in conjunction with administrator account you will next create.

**Step 4** Choose **Maintenance > Administrators > Account Management** to create a new CSA MC administrator account to provide IPS access to the MC system.

**Step 5** Create a new administrator account with the role of **Monitor**. This maintains the security of the MC by not allowing this new account to have configure privileges.


**Note**

Remember the username and password for this administrator account because you need them to configure external product interfaces on the sensor.

**Step 6** Choose **Maintenance > Administrators > Access Control** to further limit this administrator account.

**Step 7** In the Access Control window, select the administrator you created and select the group you created.


**Note**

When you save this configuration, you further limit the MC access of this new administrator account with the purpose of maintaining security on the CSA MC.

## Adding External Product Interfaces and Posture ACLs


**Caution**

In the Cisco IPS, the only external product interfaces you can add are CSA MC interfaces. The Cisco IPS supports two CSA MC interfaces.

Use the `cisco-security-agents-mc-settings ip-address` command in service external product interfaces submode to add the CSA MC as an external product interface.

The following commands apply:

- **enabled {yes | no}**—Enables/disables the receipt of information from the CSA MC.
- **host-posture-settings**—Specifies how host postures received from the CSA MC are handled:
  - **allow-unreachable-postures {yes | no}**—Allows postures for hosts that are not reachable by the CSA MC.
 

A host is not reachable if the CSA MC cannot establish a connection with the host on any IP addresses in the posture of the host. This option is useful in filtering the postures whose IP addresses may not be visible to the IPS or may be duplicated across the network. This filter is most applicable in network topologies where hosts that are not reachable by the CSA MC are also not reachable by the IPS, for example if the IPS and the CSA MC are on the same network segment.
  - **enabled {yes | no}**—Enables/disables receipt of host postures from the CSA MC.
  - **posture-acls {edit | insert | move} name1 {begin | end | inactive | before | after}**—Specifies the list of permitted or denied posture addresses. This command provides a mechanism for filtering postures that have IP addresses that may not be visible to the IPS or may be duplicated across the network.
  - **action {permit | deny}**—Specifies the permit or deny postures that match the specified network address.
  - **network-address address**—Specifies the network address, in the form x.x.x.x/nn, for postures to be permitted or denied.
- **password**—Specifies the password used to log in to the CSA MC.
- **port**—Specifies the TCP port to connect to on the CSA MC. The valid range is 1 to 65535. The default is 443.
- **username**—Specifies the username used to log in to the CSA MC.
- **watchlist-address-settings**—Specifies how watch listed addresses received from the CSA MC are handled:
  - **enabled {yes | no}**—Enables/disables receipt of watch list addresses from the CSA MC.
  - **manual-rr-increase**—Specifies the number added to an event RR because the attacker has been manually watch-listed by the CSA MC. The valid range is 0 to 35. The default is 25.
  - **packet-rr-increase**—Specifies the number added to an event risk rating because the attacker has been watch listed by the CSA MC because of a sessionless packet-based policy violation. The valid range is 0 to 35. The default is 10.
  - **session-rr-increase**—Specifies the number added to an event risk rating because the attacker has been watch-listed by the CSA MC because of a session-based policy violation. The valid range is 0 to 35. The default is 25.



#### Note

Make sure you add the external product as a trusted host so the sensor can communicate with it.

### Adding External Product Interfaces

To add external product interfaces, follow these steps:

**Step 1** Log in to the CLI using an account with administrator privileges.

**Step 2** Enter external product interfaces submenu.

```
sensor# configure terminal
```

```
sensor(config)# service external-product-interface
```

**Step 3** Add the CSA MC interface.

```
sensor(config-ext)# cisco-security-agents-mc-settings 209.165.200.225
sensor(config-ext-cis)#
```

**Step 4** Enable receipt of information from the CSA MC.

```
sensor(config-ext-cis)# enabled yes
```

**Step 5** Change the default port setting.

```
sensor(config-ext-cis)# port 80
```

**Step 6** Configure the login settings:

a. Enter the username.

```
sensor(config-ext-cis)# username jsmith
```

b. Enter and confirm the password.

```
sensor(config-ext-cis)# password
Enter password[]: *****
Re-enter password: *****
sensor(config-ext-cis)#
```




---

**Note** Steps 7 through 10 are optional. If you do not perform Steps 7 through 10, the default values are used to receive all the CSA MC information with no filters applied.

---

**Step 7** (Optional) Configure the watch list settings:

a. Allow the watch list information to be passed from the external product to the sensor.

```
sensor(config-ext-cis-wat)# enabled yes
```




---

**Note** If you do not enable the watch list, the watch list information received from a CSA MC is deleted.

---

b. Change the percentage of the manual watch list RR from the default of 25.

```
sensor(config-ext-cis-wat)# manual-rr-increase 30
```

c. Change the percentage of the session-based watch list RR from the default of 25.

```
sensor(config-ext-cis-wat)# session-rr-increase 30
```

d. Change the percentage of the packet-based watch list RR from the default of 10.

```
sensor(config-ext-cis-wat)# packet-rr-increase 20
```

**Step 8** (Optional) Allow the host posture information to be passed from the external product to the sensor.

```
sensor(config-ext-cis)# host-posture-settings
sensor(config-ext-cis-hos)# enabled yes
```




---

**Note** If you do not enable the host posture information, the host posture information received from a CSA MC is deleted.

---



- Step 9** (Optional) Allow the host posture information from unreachable hosts to be passed from the external product to the sensor.

```
sensor(config-ext-cis-hos)# allow-unreachable-postures yes
```



**Note** A host is not reachable if the CSA MC cannot establish a connection with the host on any of the IP addresses in the host's posture. This option is useful in filtering the postures whose IP addresses may not be visible to the IPS or may be duplicated across the network. This filter is most applicable in network topologies where hosts that are not reachable by the CSA MC are also not reachable by the IPS, for example if the IPS and the CSA MC are on the same network segment.

- Step 10** Configure a posture ACL:

- a. Add the posture ACL into the ACL list.

```
sensor(config-ext-cis-hos)# posture-acls insert name1 begin
sensor(config-ext-cis-hos-pos)#
```



**Note** Posture ACLs are network address ranges for which host postures are allowed or denied. Use posture ACLs to filter postures that have IP addresses that may not be visible to the IPS or may be duplicated across the network.

- b. Enter the network address the posture ACL will use.

```
sensor(config-ext-cis-hos-pos)# network-address 192.0.2.0/24
```

- c. Choose the action (deny or permit) the posture ACL will take.

```
sensor(config-ext-cis-hos-pos)# action permit
```

- Step 11** Verify the settings.

```
sensor(config-ext-cis-hos-pos)# exit
sensor(config-ext-cis-hos)# exit
sensor(config-ext-cis)# exit
sensor(config-ext)# show settings
cisco-security-agents-mc-settings (min: 0, max: 2, current: 1)

ip-address: 209.165.200.225

interface-type: extended-sdee <protected>
enabled: yes default: yes
url: /csamc50/sdee-server <protected>
port: 80 default: 443
use-ssl

always-yes: yes <protected>

username: jsmith
password: <hidden>
host-posture-settings

enabled: yes default: yes
allow-unreachable-postures: yes default: yes
posture-acls (ordered min: 0, max: 10, current: 1 - 1 active, 0 inactive)

ACTIVE list-contents
```

```

NAME: name1

network-address: 192.0.2.0/24
action: permit

watchlist-address-settings

enabled: yes default: yes
manual-rr-increase: 30 default: 25
session-rr-increase: 30 default: 25
packet-rr-increase: 20 default: 10

sensor(config-ext)#

```

**Step 12** Exit external product interface submode.

```

sensor(config-ext)# exit
Apply Changes?[yes]:

```

**Step 13** Press **Enter** to apply the changes or enter **no** to discard them.

#### For More Information

For the procedure for adding trusted hosts, see [Adding TLS Trusted Hosts, page 4-53](#).

## Troubleshooting External Product Interfaces

To troubleshoot external product interfaces, check the following:

- Make sure the interface is active by checking the output from the **show statistics external-product-interface** command in the CLI, or choose **Monitoring > Sensor Monitoring > Support Information > Statistics** in the IDM and check the Interface state line in the response, or choose **Configuration > sensor\_name > Sensor Monitoring > Support Information > Statistics** in the IME, and check the Interface state line in the response.
- Make sure you have added the CSA MC IP address to the trusted hosts. If you forgot to add it, add it, wait a few minutes and then check again.
- Confirm subscription login information by opening and closing a subscription on the CSA MC using the browser.
- Check the Event Store for the CSA MC subscription errors.

#### For More Information

- For the procedure for adding trusted hosts, see [Adding TLS Trusted Hosts, page 4-53](#).
- For the procedure for displaying events, see [Clearing Events from Event Store, page 8-42](#).



## Configuring IP Logging

---

This chapter describes how to configure IP logging on the sensor. It contains the following sections:

- [Understanding IP Logging, page 12-2](#)
- [Configuring Automatic IP Logging, page 12-2](#)
- [Configuring Manual IP Logging for a Specific IP Address, page 12-3](#)
- [Displaying the Contents of IP Logs, page 12-5](#)
- [Stopping Active IP Logs, page 12-6](#)
- [Copying IP Log Files to Be Viewed, page 12-7](#)

## IP Logging Notes and Caveats

The following notes and caveats apply to IP logging:

- Enabling IP logging slows down system performance.
- IP logging allows a maximum limit of 20 concurrent IP log files. Once the limit of 20 is reached, you receive the following message in `main.log`:  
`Cid/W errWarnIpLogProcessor::addIpLog: Ran out of file descriptors`
- You cannot delete or manage IP log files. The **no iplog** command does not delete IP logs, it only stops more packets from being recorded for that IP log. IP logs are stored in a circular buffer that is never filled because new IP logs overwrite old ones.
- You can configure IP logging restrictions using the **permit-packet-logging true | false** command.
- On IPS sensors with multiple processors, packets may be captured out of order in the IP logs and by the **packet** command. Because the packets are not processed using a single processor, the packets can become out of sync when received from multiple processors.

### For More Information

For detailed information about the packet-related command restrictions, see [Configuring Packet Command Restriction, page 4-26](#).

# Understanding IP Logging

You can manually configure the sensor to capture all IP traffic associated with a host you specify by IP address. You can specify how long you want the IP traffic to be logged, how many packets you want logged, and how many bytes you want logged. The sensor stops logging IP traffic at the first parameter you specify.

You can also have the sensor log IP packets every time a particular signature is fired. You can specify how long you want the sensor to log IP traffic and how many packets and bytes you want logged.

You can copy the IP logs from the sensor and have them analyzed by a tool that can read packet files in a libpcap format, such as Wireshark or TCPDUMP.

**Note**

Each alert references IP logs that are created because of that alert. If multiple alerts create IP logs for the same IP address, only one IP log is created for all the alerts. Each alert references the same IP log. However, the output of the IP log status only shows the event ID of the first alert triggering the IP log.

**Note**

IP logging allows a maximum limit of 20 concurrent IP log files. Once the limit of 20 is reached, you receive the following message in main.log: `Cid/W errWarnIpLogProcessor::addIpLog: Ran out of file descriptors`

# Configuring Automatic IP Logging

Use the **ip-log-packets** *number*, **ip-log-time** *number*, and **ip-log-bytes** *number* commands to configure automatic IP logging parameters on the sensor.

The following commands apply:

- **ip-log-packets**—Identifies the number of packets you want logged. The valid value is 0 to 65535. The default is 0.
- **ip-log-time**—Identifies the duration in seconds you want the sensor to log packets. The valid value is 30 to 300 seconds. The default is 30 seconds.
- **ip-log-bytes** —Identifies the maximum number of bytes you want logged. The valid value is 0 to 2147483647. The default is 0.
- **default**—Resets the parameters.

**Note**

An automatic IP log continues capturing packets until one of these parameters is reached.

Automatic IP logging is configured on a per signature basis or as an event action override. The following actions trigger automatic IP logging:

- log-attacker-packets
- log-victim-packets
- log-pair-packets

### Configuring Automatic IP Logging

To configure automatic IP logging parameters, follow these steps:

- 
- Step 1** Log in to the CLI using an account with administrator or operator privileges.
- Step 2** Enter signature definition IP log configuration submode.
- ```
sensor# configure terminal
sensor(config)# service signature-definition sig0
sensor(config-sig)# ip-log
```
- Step 3** Specify the number of packets you want the sensor to log. The range is 0 to 65535.
- ```
sensor(config-sig-ip)# ip-log-packets 200
```
- Step 4** Specify the duration in seconds you want the sensor to log packets. The range is 30 to 300 seconds.
- ```
sensor(config-sig-ip)# ip-log-time 60
```
- Step 5** Specify the number of bytes you want logged. The range is 0 to 2147483647.
- ```
sensor(config-sig-ip)# ip-log-bytes 5024
```
- Step 6** Verify the settings.
- ```
sensor(config-sig-ip)# show settings
ip-log
-----
ip-log-packets: 200 default: 0
ip-log-time: 60 default: 30
ip-log-bytes: 5024 default: 0
-----
sensor(config-sig-ip)#
```
- Step 7** Exit IP logging submode.
- ```
sensor(config-sig-ip)# exit
sensor(config-sig)# exit
Apply Changes?:[yes]:
```
- Step 8** Press **Enter** to apply the changes or type **no** to discard the changes.
- 

#### For More Information

- To copy and view an IP log file, see [Copying IP Log Files to Be Viewed](#), page 12-7.
- For more information on event actions, see [Assigning Actions to Signatures](#), page 7-20 and [Configuring Event Action Overrides](#), page 8-17.

## Configuring Manual IP Logging for a Specific IP Address

Use the **iplog name ip\_address [duration minutes] [packets numPackets] [bytes numBytes]** command to log IP packets manually on a virtual sensor for a specific IP address.

The following commands apply:

- *name*—Specifies the virtual sensor on which to begin and end logging.
- *ip\_address*—Logs packets containing the specified source and/or destination IP address.

- *minutes*—Specifies the duration the logging should be active. The valid range is 1 to 60 minutes. The default is 10 minutes.
- *numPackets*—Specifies the maximum number of packets to log. The valid range is 0 to 4294967295. The default is 1000 packets.
- *numBytes*—Specifies the maximum number of bytes to log. The valid range is 0 to 4294967295. A value of 0 indicates unlimited bytes.



**Note** The *minutes*, *numPackets*, and *numBytes* parameters are optional, you do not have to specify all three. However, if you include more than one parameter, the sensor continues logging only until the first threshold is reached. For example, if you set the duration to 5 minutes and the number of packets to 1000, the sensor stops logging after the 1000th packet is captured, even if only 2 minutes have passed.

### Configuring Manual IP Logging

To manually log packets on a virtual sensor for a specific IP address, follow these steps:

- Step 1** Log in to the CLI using an account with administrator or operator privileges.
- Step 2** Start IP logging for a specific IP address. The range is 1 to 60 minutes.

```
sensor# iplog vs0 10.16.0.0 duration 5
Logging started for virtual sensor vs0, IP address 192.0.2.1, Log ID 1
Warning: IP Logging will affect system performance.
sensor#
```

The example shows the sensor logging all IP packets for 5 minutes to and from the IP address 192.0.2.1.



**Note** Make note of the Log ID for future reference.

- Step 3** Monitor the IP log status with the **iplog-status** command.

```
sensor# iplog-status
Log ID: 1
IP Address 1: 192.0.2.1
Virtual Sensor: vs0
Status: added
Event ID: 0
Bytes Captured: 0
Packets Captured: 0
sensor
```



**Note** Each alert references IP logs that are created because of that alert. If multiple alerts create IP logs for the same IP address, only one IP log is created for all the alerts. Each alert references the same IP log. However, the output of the IP log status only shows the event ID of the first alert triggering the IP log.

**For More Information**

- To stop logging IP packets for a specific IP address, see [Stopping Active IP Logs, page 12-6](#).
- To log IP packets as an event associated with a signature, see [Configuring Automatic IP Logging, page 12-2](#).
- To copy and view an IP log file, see [Copying IP Log Files to Be Viewed, page 12-7](#).

## Displaying the Contents of IP Logs

Use the `iplog-status [log-id log_id] [brief] [reverse] [ | {begin regular_expression | exclude regular_expression | include regular_expression } ]` command to display the description of the available IP log contents.

When the log is created, the status reads `added`. If and when the first entry is inserted in the log, the status changes to `started`. When the log is completed, because it reaches the packet count limit, for example, the status changes to `completed`.

The following commands apply:

- `log_id`—(Optional) Specifies the log ID of the file for which you want to see the status.
- `brief`—(Optional) Displays a summary of IP log status information for each log.
- `reverse`—(Optional) Displays the list in reverse chronological order (newest log first).
- `|`—(Optional) Indicates that an output processing specification follows.
- `regular_expression`—Specifies any regular expression found in the IP log status output.
- `begin`—Searches the output of the `more` command and displays the output from the first instance of a specified string.
- `exclude`—Filters the IP log status output so that it excludes lines that contain a particular regular expression.
- `include`—Filters the IP log status output so that it includes lines that contain a particular regular expression.

**Displaying IP Logs**

To view the contents of IP logs, follow these steps:

- 
- Step 1** Log in to the CLI.
- Step 2** Display the status of all IP logs.

```

sensor# iplog-status
Log ID: 2425
IP Address 1: 192.0.2.1
Virtual Sensor: vs0
Status: started
Start Time: 2003/07/30 18:24:18 2002/07/30 12:24:18 CST
Packets Captured: 1039438

Log ID: 2342
IP Address 1: 192.0.2.10
IP Address 2: 192.0.2.20
Virtual Sensor: vs0
Status: completed
Event ID: 209348

```

```

Start Time: 2003/07/30 18:24:18 2002/07/30 12:24:18 CST
End Time: 2003/07/30 18:34:18 2002/07/30 12:34:18 CST
sensor#

```

**Step 3** Display a brief list of all IP logs.

```

sensor# iplog-status brief
Log ID VS IP Address1 Status Event ID Start Date
2425 vs0 192.0.2.10 started N/A 2003/07/30
2342 vs0 192.0.2.20 completed 209348 2003/07/30
sensor#

```

## Stopping Active IP Logs

Use the **no iplog** [**log-id** *log\_id* | **name** *name*] command to stop logging for the logs that are in the `started` state and to remove logs that are in the `added` state. The **no iplog** command does not remove or delete the IP log. It only signals to the sensor to stop capturing additional packets on that IP log.



### Note

Using the **no iplog** command on an `added` state IP log stops the IP log. The `added` state means that the IP log is still empty (no packets). Stopping it when there are no packets means you are stopping an empty IP log. An empty log is removed when it is stopped.

The following commands apply:

- *log\_id*—Specifies the log ID of the logging session to stop. Use the **iplog-status** command to find the log ID.
- *name*—Specifies the virtual sensor on which to begin or end logging.

### Disabling IP Logging Sessions

To disable one or all IP logging sessions, follow these steps:

**Step 1** Log in to the CLI using an account with administrator or operator privileges.

**Step 2** Stop a particular IP logging session:

- Find the log ID of the session you want to stop.

```

sensor# iplog-status
Log ID: 1
IP Address 1: 192.0.2.1
Virtual Sensor: vs0
Status: added
Event ID: 0
Bytes Captured: 0
Packets Captured: 0
sensor#

```



### Note

Each alert references IP logs that are created because of that alert. If multiple alerts create IP logs for the same IP address, only one IP log is created for all the alerts. Each alert references the same IP log. However, the output of the IP log status only shows the event ID of the first alert triggering the IP log.



- b. Stop the IP log session.

```
sensor# no iplog log-id 137857512
```

- Step 3** Stop all IP logging sessions on a virtual sensor.

```
sensor# no iplog name vs0
```

- Step 4** Verify that IP logging has been stopped. When the logs are stopped, the status shows them as completed.

```
sensor# iplog-status
Log ID: 1
IP Address 1: 192.0.2.1
Virtual Sensor: vs0
Status: completed
Event ID: 0
Bytes Captured: 0
Packets Captured: 0
sensor#
```

## Copying IP Log Files to Be Viewed

Use the **copy iplog log\_id destination\_url** command to copy IP log files to an FTP or SCP server so that you can view them with a sniffing tool such as Ethereal or tcpdump.

The following commands apply:

- *log\_id*—Specifies the log ID of the logging session. You can retrieve the log ID using the **iplog-status** command.
- *destination\_url*—Specifies the location of the destination file to be copied. It can be a URL or a keyword.

The exact format of the source and destination URLs varies according to the file. Here are the valid types:

- **ftp:**—Destination URL for an FTP network server. The syntax for this prefix is:  
ftp://[[username@]location][[/relativeDirectory]/filename  
ftp://[[username@]location][[/absoluteDirectory]/filename
- **scp:**—Destination URL for the SCP network server. The syntax for this prefix is:  
scp://[[username@]location][[/relativeDirectory]/filename  
scp://[[username@]location][[/absoluteDirectory]/filename

When you use FTP or SCP protocol, you are prompted for a password.

### Copying IP Log Files

To copy IP log files to an FTP or SCP server, follow these steps:

- Step 1** Log in to the CLI.

- Step 2** Monitor the IP log status with the **iplog-status** command until you see that the status reads completed for the log ID of the log file that you want to copy.

```
sensor# iplog-status
Log ID: 2425
```

```

IP Address: 192.0.2.1
Virtual Sensor: vs0
Status: started
Start Time: 2003/07/30 18:24:18 2002/07/30 12:24:18 CST
Packets Captured: 1039438

```

```

Log ID: 2342
IP Address: 192.0.2.2
Virtual Sensor: vs0
Status: completed
Event ID: 209348
Start Time: 2003/07/30 18:24:18 2002/07/30 12:24:18 CST
End Time: 2003/07/30 18:34:18 2002/07/30 12:34:18 CST
sensor#

```

**Step 3** Copy the IP log to your FTP or SCP server.

```

sensor# copy iplog 2342 ftp://root@209.165.200.225/user/iplog1
Password: ***** Connected to 209.165.200.225 (209.165.200.225). 220 linux.machine.com
FTP server (Version wu-2.6.0(1) Mon Feb 28 10:30 :36 EST 2000) ready. ftp> user (username)
root 331 Password required for root. Password:230 User root logged in. ftp> 200 Type set
to I. ftp> put iplog.8518.tmp iplog1 local: iplog.8518.tmp remote: iplog1 227 Entering
Passive Mode (2,4,6,8,179,125) 150 Opening BINARY mode data connection for iplog1. 226
Transfer complete. 30650 bytes sent in 0.00246 secs (1.2e+04 Kbytes/sec) ftp>

```

**Step 4** Open the IP log using a sniffer program such as Wireshark or TCPDUMP. For more information on Wireshark, go to <http://www.wireshark.org>. For more information on TCPDUMP, go to <http://www.tcpdump.org/>.

---



## Displaying and Capturing Live Traffic on an Interface

---

This chapter describes how to display, capture, copy, and erase packet files. It contains the following sections:

- [Packet Display And Capture Notes and Caveats, page 13-1](#)
- [Understanding Packet Display and Capture, page 13-2](#)
- [Displaying Live Traffic on an Interface, page 13-2](#)
- [Capturing Live Traffic on an Interface, page 13-4](#)
- [Copying the Packet File, page 13-6](#)
- [Erasing the Packet File, page 13-7](#)

### Packet Display And Capture Notes and Caveats

The following notes and caveats apply to capturing packet files:

- Although capturing live traffic off the interface does not disrupt any of the functionality of the sensor, it does cause significant performance degradation.
- Changing the interface configuration results in abnormal termination of any **packet** command running on that interface.
- You can configure packet capture/display restrictions using the **permit-packet-logging true | false** command.
- On IPS sensors with multiple processors, packets may be captured out of order in the IP logs and by the **packet** command. Because the packets are not processed using a single processor, the packets can become out of sync when received from multiple processors.
- When the IPS 4500 series sensors are configured in VLAN pairs, the **packet display** command does not work without the VLAN option if the **expression** keyword is also used.

#### For More Information

For detailed information about the packet-related command restrictions, see [Configuring Packet Command Restriction, page 4-26](#).

# Understanding Packet Display and Capture

You can display or capture live traffic from an interface and have the live traffic or a previously captured file put directly on the screen. Storage is available for one local file only, subsequent capture requests overwrites an existing file. The size of the storage file varies depending on the platform. A message may be displayed if the maximum file size is reached before the requested packet count is captured.

## Displaying Live Traffic on an Interface

Use the **packet display** *interface\_name* [**snaplen** *length*] [**count** *count*] [**verbose**] [**expression** *expression*] command to display live traffic from an interface directly on your screen. Use the **packet display iplog** *id* [**verbose**] [**expression** *expression*] to display iplogs.



### Note

To terminate the live display, press **Ctrl-C**.

The following commands apply:

- **interface\_name**—Specifies the interface name, interface type (GigabitEthernet, FastEthernet, Management, PortChannel) followed by slot/port. You can only use an interface name that exists in the system.
- **snaplen**—(Optional) Specifies the maximum number of bytes captured for each packet. The valid range is 68 to 1600. The default is 0. A value of 0 means use the required length to catch whole packets.
- **count**—(Optional) Specifies the maximum number of packets to capture. The valid range is 1 to 10000.



### Note

If you do not specify this option, the capture terminates after the maximum file size is captured.

- **verbose**—(Optional) Displays the protocol tree for each packet rather than a one-line summary.
- **expression**—Specifies the packet-display filter expression. This expression is passed directly to TCPDUMP and must meet the TCPDUMP expression syntax.



### Note

The expression syntax is described in the TCPDUMP man page.



### Note

If you use the **expression** option when monitoring packets with VLAN headers, the expression does not match properly unless **vlan and** is added to the beginning of the expression. For example, **packet display iplog 926299444 verbose expression icmp** Will NOT show ICMP packets; **packet display iplog 926299444 verbose expression vlan and icmp** WILL show ICMP packets. It is often necessary to use **expression vlan and** on IPS appliance interfaces connected to trunk ports.

- **file-info**—Displays information about the stored packet file. **File-info** displays the following information:

Captured by: *user:id*, Cmd: *cliCmd*

Start: yyyy/mm/dd hh:mm:ss zone, End: yyyy/mm/dd hh:mm:ss zone or in-progress.

Where *user* = the username of user initiating capture, *id* = the CLI ID of the user, and *cliCmd* = the command entered to perform the capture.



### Caution

Executing the **packet display** command causes significant performance degradation.

### Displaying Live Traffic From an Interface

To configure the sensor to display live traffic from an interface on the screen, follow these steps:

- Step 1** Log in to the sensor using an account with administrator or operator privileges.
- Step 2** Display the live traffic on the interface you are interested in, for example, GigabitEthernet0/1.

```

sensor# packet display GigabitEthernet0/1
Warning: This command will cause significant performance degradation
tcpdump: listening on ge0_1, link-type EN10MB (Ethernet), capture size 65535 bytes
03:43:05.691883 IP (tos 0x10, ttl 64, id 55460, offset 0, flags [DF], length: 100)
10.89.147.31.22 > 10.89.147.50.41805: P [tcp sum ok] 4233955485:4233955533(48) ack
1495691730 win 8576 <nop,nop,timestamp 44085169 226014949>
03:43:05.691975 IP (tos 0x10, ttl 64, id 55461, offset 0, flags [DF], length: 164)
10.89.147.31.22 > 10.89.147.50.41805: P [tcp sum ok] 48:160(112) ack 1 win 8576
<nop,nop,timestamp 44085169 226014949>
03:43:05.691998 IP (tos 0x10, ttl 64, id 53735, offset 0, flags [DF], length: 52)
10.89.147.50.41805 > 10.89.147.31.22: . [tcp sum ok] 1:1(0) ack 48 win 11704
<nop,nop,timestamp 226014949 44085169>
03:43:05.693165 IP (tos 0x10, ttl 64, id 53736, offset 0, flags [DF], length: 52)
10.89.147.50.41805 > 10.89.147.31.22: . [tcp sum ok] 1:1(0) ack 160 win 11704
<nop,nop,timestamp 226014949 44085169>
03:43:05.693351 IP (tos 0x10, ttl 64, id 55462, offset 0, flags [DF], length: 316)
10.89.147.31.22 > 10.89.147.50.41805: P [tcp sum ok] 160:424(264) ack 1 win 8576
<nop,nop,timestamp 44085169 226014949>
03:43:05.693493 IP (tos 0x10, ttl 64, id 55463, offset 0, flags [DF], length: 292)
10.89.147.31.22 > 10.89.147.50.41805: P [tcp sum ok] 424:664(240) ack 1 win 8576
<nop,nop,timestamp 44085169 226014949>
03:43:05.693612 IP (tos 0x10, ttl 64, id 55464, offset 0, flags [DF], length: 292)
10.89.147.31.22 > 10.89.147.50.41805: P [tcp sum ok] 664:904(240) ack 1 win 8576
<nop,nop,timestamp 44085169 226014949>
03:43:05.693628 IP (tos 0x10, ttl 64, id 53737, offset 0, flags [DF], length: 52)
10.89.147.50.41805 > 10.89.147.31.22: . [tcp sum ok] 1:1(0) ack 424 win 11704
<nop,nop,timestamp 226014949 44085169>
03:43:05.693654 IP (tos 0x10, ttl 64, id 53738, offset 0, flags [DF], length: 52)
10.89.147.50.41805 > 10.89.147.31.22: . [tcp sum ok] 1:1(0) ack 664 win 11704
<nop,nop,timestamp 226014949 44085169>
03:43:05.693926 IP (tos 0x10, ttl 64, id 55465, offset 0, flags [DF], length: 292)
10.89.147.31.22 > 10.89.147.50.41805: P [tcp sum ok] 904:1144(240) ack 1 win 8576
<nop,nop,timestamp 44085169 226014949>
03:43:05.694043 IP (tos 0x10, ttl 64, id 55466, offset 0, flags [DF], length: 292)
10.89.147.31.22 > 10.89.147.50.41805: P [tcp sum ok] 1144:1384(240) ack 1 win 8576
<nop,nop,timestamp 44085169 226014949>
03:43:05.694163 IP (tos 0x10, ttl 64, id 55467, offset 0, flags [DF], length: 292)
10.89.147.31.22 > 10.89.147.50.41805: P [tcp sum ok] 1384:1624(240) ack 1 win 8576
<nop,nop,timestamp 44085169 226014949>
03:43:05.694209 IP (tos 0x10, ttl 64, id 53739, offset 0, flags [DF], length: 52)
10.89.147.50.41805 > 10.89.147.31.22: . [tcp sum ok] 1:1(0) ack 1384 win 11704
<nop,nop,timestamp 226014950 44085169>
03:43:05.694283 IP (tos 0x10, ttl 64, id 55468, offset 0, flags [DF], length: 292)
10.89.147.31.22 > 10.89.147.50.41805: P [tcp sum ok] 1624:1864(240) ack 1 win 8576
<nop,nop,timestamp 44085169 226014950>

```

```

03:43:05.694402 IP (tos 0x10, ttl 64, id 55469, offset 0, flags [DF], length: 292)
10.89.147.31.22 > 10.89.147.50.41805: P [tcp sum ok] 1864:2104(240) ack 1 win 8576
<nop,nop,timestamp 44085169 226014950>
03:43:05.694521 IP (tos 0x10, ttl 64, id 55470, offset 0, flags [DF], length: 292)
10.89.147.31.22 > 10.89.147.50.41805: P [tcp sum ok] 2104:2344(240) ack 1 win 8576
<nop,nop,timestamp 44085169 226014950>
03:43:05.694690 IP (tos 0x10, ttl 64, id 53740, offset 0, flags [DF], length: 52)
10.89.147.50.41805 > 10.89.147.31.22: . [tcp sum ok] 1:1(0) ack 2344 win 11704
<nop,nop,timestamp 226014950 44085169>
03:43:05.694808 IP (tos 0x10, ttl 64, id 55471, offset 0, flags [DF], length: 300)
10.89.147.31.22 > 10.89.147.50.41805: P [tcp sum ok] 2344:2592(248) ack 1 win 8576
<nop,nop,timestamp 44085169 226014950>

```

**Step 3** You can use the **expression** option to limit what you display, for example, only TCP packets.



**Note** As described in the TCPDUMP man page, the protocol identifiers tcp, udp, and icmp are also keywords and must be escaped by using two back slashes (\).

```

sensor# packet display GigabitEthernet0/1 verbose expression ip proto \\tcp
Warning: This command will cause significant performance degradation
tcpdump: listening on ge0_1, link-type EN10MB (Ethernet), capture size 65535 bytes
03:42:02.509738 IP (tos 0x10, ttl 64, id 27743, offset 0, flags [DF], length: 88)
10.89.147.31.22 > 64.101.182.54.47039: P [tcp sum ok] 3449098782:3449098830(48) ack
3009767154 win 8704
03:42:02.509834 IP (tos 0x10, ttl 64, id 27744, offset 0, flags [DF], length: 152)
10.89.147.31.22 > 64.101.182.54.47039: P [tcp sum ok] 48:160(112) ack 1 win 8704
03:42:02.510248 IP (tos 0x0, ttl 252, id 55922, offset 0, flags [none], length: 40)
64.101.182.54.47039 > 10.89.147.31.22: . [tcp sum ok] 1:1(0) ack 160 win 8760
03:42:02.511262 IP (tos 0x10, ttl 64, id 27745, offset 0, flags [DF], length: 264)
10.89.147.31.22 > 64.101.182.54.47039: P [tcp sum ok] 160:384(224) ack 1 win 8704
03:42:02.511408 IP (tos 0x10, ttl 64, id 27746, offset 0, flags [DF], length: 248)
10.89.147.31.22 > 64.101.182.54.47039: P [tcp sum ok] 384:592(208) ack 1 win 8704
03:42:02.511545 IP (tos 0x10, ttl 64, id 27747, offset 0, flags [DF], length: 240)
10.89.147.31.22 > 64.101.182.54.47039: P [tcp sum ok] 592:792(200) ack 1 win 8704

```

**Step 4** Display information about the packet file.

```

sensor# packet display file-info
Captured by: cisco:25579, Cmd: packet capture GigabitEthernet0/1
Start: 2003/02/03 02:56:48 UTC, End: 2003/02/03 02:56:51 UTC
sensor#

```

## Capturing Live Traffic on an Interface

Use the **packet capture** *interface\_name* [*snaplen length*] [*count count*] [*expression expression*] command to capture live traffic on an interface. Only one user can use the **packet capture** command at a time. A second user request results in an error message containing information about the user currently executing the capture.



### Caution

Executing the **packet capture** command causes significant performance degradation.

The **packet capture** command captures the libpcap output into a local file. Use the **packet display packet-file [verbose] [expression *expression*]** command to view the local file. Use the **packet display file-info** to display information about the local file, if any.

The following commands apply:

- *interface\_name*—Specifies the logical interface name. You can only use an interface name that exists in the system.
- **snaptlen**—Specifies the maximum number of bytes captured for each packet (optional). The valid range is 68 to 1600. The default is 0.
- **count**—Specifies the maximum number of packets to capture (optional). The valid range is 1 to 10000.



**Note** If you do not specify this option, the capture terminates after the maximum file size is captured.

- **expression**—Specifies the packet-capture filter expression. This expression is passed directly to TCPDUMP and must meet the TCPDUMP expression syntax.
- **file-info**—Displays information about the stored packet file.

**File-info** displays the following information:

```
Captured by: user:id, Cmd: cliCmd
Start: yyyy/mm/dd hh:mm:ss zone, End: yyyy/mm/dd hh:mm:ss zone or in-progress
```

Where *user* = username of user initiating capture, *id* = CLI ID of the user, and *cliCmd* = command entered to perform the capture.

- **verbose**—Displays the protocol tree for each packet rather than a one-line summary. This parameter is optional.

### Capturing Live Traffic on an Interface

To configure the sensor to capture live traffic on an interface, follow these steps:

- Step 1** Log in to the sensor using an account with administrator or operator privileges.
- Step 2** Capture the live traffic on the interface you are interested in, for example, GigabitEthernet0/1.

```
sensor# packet capture GigabitEthernet0/1
Warning: This command will cause significant performance degradation
tcpdump: WARNING: ge0_1: no IPv4 address assigned
tcpdump: listening on ge0_1, link-type EN10MB (Ethernet), capture size 65535 bytes
125 packets captured
126 packets received by filter
0 packets dropped by kernel
```

- Step 3** View the captured packet file.

```
sensor# packet display packet-file
reading from file /usr/cids/idsRoot/var/packet-file, link-type EN10MB (Ethernet)
03:03:13.216768 802.1d config TOP_CHANGE 8000.00:04:9a:66:35:01.8025 root 8000.0
0:04:6d:f9:e8:82 pathcost 8 age 2 max 20 hello 2 fdelay 15
03:03:13.232881 IP 64.101.182.244.1978 > 10.89.130.108.23: . ack 3266153791 win
64328
03:03:13.232895 IP 10.89.130.108.23 > 64.101.182.244.1978: P 1:157(156) ack 0 wi
n 5840
03:03:13.433136 IP 64.101.182.244.1978 > 10.89.130.108.23: . ack 157 win 65535
03:03:13.518335 IP 10.89.130.134.42342 > 255.255.255.255.42342: UDP, length: 76
```

```

03:03:15.218814 802.1d config TOP_CHANGE 8000.00:04:9a:66:35:01.8025 root 8000.0
0:04:6d:f9:e8:82 pathcost 8 age 2 max 20 hello 2 fdelay 15
03:03:15.546866 IP 64.101.182.244.1978 > 10.89.130.108.23: P 0:2(2) ack 157 win
65535
03:03:15.546923 IP 10.89.130.108.23 > 64.101.182.244.1978: P 157:159(2) ack 2 wi
n 5840
03:03:15.736377 IP 64.101.182.244.1978 > 10.89.130.108.23: . ack 159 win 65533
03:03:17.219612 802.1d config TOP_CHANGE 8000.00:04:9a:66:35:01.8025 root 8000.0
0:04:6d:f9:e8:82 pathcost 8 age 2 max 20 hello 2 fdelay 15
03:03:19.218535 802.1d config TOP_CHANGE 8000.00:04:9a:66:35:01.8025 root 8000.0
0:04:6d:f9:e8:82 pathcost 8 age 2 max 20 hello 2 fdelay 15
03:03:19.843658 IP 64.101.182.143.3262 > 10.89.130.23.445: P 3749577803:37495778
56(53) ack 3040953472 win 64407
03:03:20.174835 IP 161.44.55.250.1720 > 10.89.130.60.445: S 3147454533:314745453
3(0) win 65520 <mss 1260,nop,nop,sackOK>
03:03:21.219958 802.1d config TOP_CHANGE 8000.00:04:9a:66:35:01.8025 root 8000.0
0:04:6d:f9:e8:82 pathcost 8 age 2 max 20 hello 2 fdelay 15
03:03:21.508907 IP 161.44.55.250.1809 > 10.89.130.61.445: S 3152179859:315217985
9(0) win 65520 <mss 1260,nop,nop,sackOK>
03:03:23.221004 802.1d config TOP_CHANGE 8000.00:04:9a:66:35:01.8025 root 8000.0
0:04:6d:f9:e8:82 pathcost 8 age 2 max 20 hello 2 fdelay 15
03:03:23.688099 IP 161.44.55.250.1975 > 10.89.130.63.445: S 3160484670:316048467
0(0) win 65520 <mss 1260,nop,nop,sackOK>
03:03:25.219054 802.1d config TOP_CHANGE 8000.00:04:9a:66:35:01.8025 root 8000.0
0:04:6d:f9:e8:82 pathcost 8 age 2 max 20 hello 2 fdelay 15
03:03:25.846552 IP 172.20.12.10.2984 > 10.89.130.127.445: S 1345848756:134584875
6(0) win 64240 <mss 1460,nop,nop,sackOK>
03:03:26.195342 IP 161.44.55.250.2178 > 10.89.130.65.445: S 3170518052:317051805
2(0) win 65520 <mss 1260,nop,nop,sackOK>
03:03:27.222725 802.1d config TOP_CHANGE 8000.00:04:9a:66:35:01.8025 root 8000.0
0:04:6d:f9:e8:82 pathcost 8 age 2 max 20 hello 2 fdelay 15
03:03:27.299178 IP 161.44.55.250.2269 > 10.89.130.66.445: S 3174717959:317471795
9(0) win 65520 <mss 1260,nop,nop,sackOK>
03:03:27.308798 arp who-has 161.44.55.250 tell 10.89.130.66
03:03:28.383028 IP 161.44.55.250.2349 > 10.89.130.67.445: S 3178636061:317863606
1(0) win 65520 <mss 1260,nop,nop,sackOK>
--MORE--

```

#### Step 4 View any information about the packet file.

```

sensor# packet display file-info
Captured by: cisco:8874, Cmd: packet capture GigabitEthernet0/1
Start: 2003/01/07 00:12:50 UTC, End: 2003/01/07 00:15:30 UTC
sensor#

```

## Copying the Packet File

Use the **copy packet-file destination\_url** command to copy the packet file to an FTP or SCP server for saving or further analysis with another tool, such as Wireshark or TCPDUMP.

The following commands apply:

- **packet-file**—Specifies the locally stored libpcap file that you captured using the **packet capture** command.
- **destination\_url**—Specifies the location of the destination file to be copied. It can be a URL or a keyword.






---

**Note** The exact format of the source and destination URLs varies according to the file.

---

- **ftp:**—Destination URL for an FTP network server. The syntax for this prefix is:  
`ftp://[[username@]location][relativeDirectory]/filename`  
`ftp://[[username@]location][absoluteDirectory]/filename`
- **scp:**—Destination URL for the SCP network server. The syntax for this prefix is:  
`scp://[[username@]location][relativeDirectory]/filename`  
`scp://[[username@]location][absoluteDirectory]/filename`




---

**Note** When you use FTP or SCP protocol, you are prompted for a password.

---

To copy packets files to an FTP or SCP server, follow these steps:

---

**Step 1** Log in to the CLI using an account with administrator privileges.

**Step 2** Copy the packet-file to an FTP or SCP server.

```
sensor# copy packet-file scp://jbrown@209.165.200.225/work/
Password: *****
packet-file 100% 1670 0.0KB/s 00:00
sensor#
```

**Step 3** View the packet file with Wireshark or TCPDUMP.

---

## Erasing the Packet File

Use the **erase packet-file** command to erase the packet file. There is only one packet file. It is 16 MB and is over-written each time you use the **packet capture** command. To erase the packet file, follow these steps:

---

**Step 1** Display information about the current captured packet file.

```
sensor# packet display file-info
Captured by: cisco:1514, Cmd: packet capture GigabitEthernet0/1
Start: 2005/02/15 03:55:00 CST, End: 2005/02/15 03:55:05 CST
sensor#
```

**Step 2** Erase the packet file.

```
sensor# erase packet-file
sensor#
```

**Step 3** Verify that you have erased the packet file.

```
sensor# packet display file-info
No packet-file available.
sensor#
```

---





# Configuring Attack Response Controller for Blocking and Rate Limiting

---

This chapter provides information for setting up the ARC to perform blocking and rate limiting on the sensor. It the following sections:

- [Blocking Notes and Caveats, page 14-1](#)
- [Understanding Blocking, page 14-2](#)
- [Understanding Rate Limiting, page 14-4](#)
- [Understanding Service Policies for Rate Limiting, page 14-5](#)
- [Before Configuring ARC, page 14-5](#)
- [Supported Devices, page 14-6](#)
- [Configuring Blocking Properties, page 14-7](#)
- [Configuring User Profiles, page 14-20](#)
- [Configuring Blocking and Rate Limiting Devices, page 14-21](#)
- [Configuring the Sensor to be a Master Blocking Sensor, page 14-28](#)
- [Configuring Host Blocking, page 14-31](#)
- [Configuring Network Blocking, page 14-32](#)
- [Configuring Connection Blocking, page 14-33](#)
- [Obtaining a List of Blocked Hosts and Connections, page 14-34](#)

## Blocking Notes and Caveats

The following notes and caveats apply to blocking:

- The ARC is formerly known as Network Access Controller. Although the name has been changed, the IDM, the IME, and the CLI contain references to Network Access Controller, **nac**, and **network-access**.
- Blocking is not supported on the FWSM in multiple mode admin context.
- Connection blocks and network blocks are not supported on adaptive security appliances. Adaptive security appliances only support host blocks with additional connection information.

- Do not confuse blocking with the ability of the sensor to drop packets. The sensor can drop packets when the following actions are configured for a sensor in inline mode: deny packet inline, deny connection inline, and deny attacker inline.
- The ACLs that ARC makes should never be modified by you or any other system. These ACLs are temporary and new ACLs are constantly being created by the sensor. The only modifications that you can make are to the Pre- and Post-Block ACLs.
- Rate limiting and blocking are not supported for IPv6 traffic. If a signature is configured with a block or rate limit event action and is triggered by IPv6 traffic, an alert is generated but the action is not carried out.
- Two sensors cannot control blocking or rate limiting on the same device. If this situation is needed, configure one sensor as the master blocking sensor to manage the devices and the other sensors can forward their requests to the master blocking sensor.
- Pre-Block and Post-Block ACLS do not apply to rate limiting.
- When you add a master blocking sensor, you reduce the number of blocking devices per sensor. For example, if you want to block on 10 security appliances and 10 routers with one blocking interface/direction each, you can assign 10 to the sensor and assign the other 10 to a master blocking sensor.
- While blocking is disabled, the ARC continues to receive blocks and track the time on active blocks, but will not apply new blocks or remove blocks from the managed devices. After blocking is reenabled, the blocks on the devices are updated.
- We recommend that you do not permit the sensor to block itself, because it may stop communicating with the blocking device. You can configure this option if you can ensure that if the sensor creates a rule to block its own IP address, it will not prevent the sensor from accessing the blocking device.
- You **MUST** create a user profile before configuring the blocking device.

## Understanding Blocking

The ARC is responsible for managing network devices in response to suspicious events by blocking access from attacking hosts and networks. The ARC blocks the IP address on the devices it is managing. It sends the same block to all the devices it is managing, including any other master blocking sensors. The ARC monitors the time for the block and removes the block after the time has expired.

The ARC completes the action response for a new block in no more than 7 seconds. In most cases, it completes the action response in less time. To meet this performance goal, you should not configure the sensor to perform blocks at too high a rate or to manage too many blocking devices and interfaces. We recommend that the maximum number of blocks not exceed 250 and the maximum number of blocking items not exceed 10. To calculate the maximum number of blocking items, a security appliance counts as one blocking item per blocking context. A router counts as one blocking item per blocking interface/direction. A switch running Catalyst software counts as one blocking item per blocking VLAN. If the recommended limits are exceeded, the ARC may not apply blocks in a timely manner or may not be able to apply blocks at all.



### Caution

---

Blocking is not supported on the FWSM in multiple mode admin context.

---

For security appliances configured in multi-mode, Cisco IPS does not include VLAN information in the block request. Therefore you must make sure the IP addresses being blocked are correct for each security appliance. For example, the sensor is monitoring packets on a security appliance customer context that

is configured for VLAN A, but is blocking on a different security appliance customer context that is configured for VLAN B. Addresses that trigger blocks on VLAN A may refer to a different host on VLAN B.

There are three types of blocks:

- Host block—Blocks all traffic from a given IP address.
- Connection block—Blocks traffic from a given source IP address to a given destination IP address and destination port. Multiple connection blocks from the same source IP address to either a different destination IP address or destination port automatically switch the block from a connection block to a host block.
- Network block—Blocks all traffic from a given network. You can initiate host and connection blocks manually or automatically when a signature is triggered. You can only initiate network blocks manually.

**Note**

Connection blocks and network blocks are not supported on adaptive security appliances. Adaptive security appliances only support host blocks with additional connection information.

**Caution**

Do not confuse blocking with the ability of the sensor to drop packets. The sensor can drop packets when the following actions are configured for a sensor in inline mode: deny packet inline, deny connection inline, and deny attacker inline.

For automatic blocks, you must configure **request-block-host** or **request-block-connection** as the event action for particular signatures, and add them to any event action overrides you have configured, so that the SensorApp sends a block request to the ARC when the signature is triggered. When the ARC receives the block request from the SensorApp, it updates the device configurations to block the host or connection.

On Cisco routers and Catalyst 6500 series switches, ARC creates blocks by applying ACLs or VACLs. ACLs and VACLs permit or deny passage of data packets through interface directions or VLANs. Each ACL or VACL contains permit and deny conditions that apply to IP addresses. The security appliances do not use ACLs or VACLs. The built-in **shun** and **no shun** command is used.

**Caution**

The ACLs that ARC makes should never be modified by you or any other system. These ACLs are temporary and new ACLs are constantly being created by the sensor. The only modifications that you can make are to the Pre- and Post-Block ACLs.

You need the following information for the ARC to manage a device:

- Login user ID (if the device is configured with AAA).
- Login password.
- Enable password (not needed if the user has enable privileges).
- Interfaces to be managed (for example, ethernet0, vlan100).
- Any existing ACL or VACL information you want applied at the beginning (Pre-Block ACL or VACL) or end (Post-Block ACL or VACL) of the ACL or VACL that will be created. This does not apply to the security appliances because they do not use ACLs to block.
- Whether you are using Telnet or SSH to communicate with the device.
- IP addresses (host or range of hosts) you never want blocked.

- How long you want the blocks to last.

**Tip**

To check the status of the ARC, type **show statistics network-access** at the `sensor#`. The output shows the devices you are managing, any active blocks and rate limits, and the status of all devices..

**Note**

Rate limiting and blocking are not supported for IPv6 traffic. If a signature is configured with a block or rate limit event action and is triggered by IPv6 traffic, an alert is generated but the action is not carried out.

**For More Information**

- For the procedure to add request-block-host or request-block-connection event actions to a signature, see [Assigning Actions to Signatures, page 7-20](#).
- For the procedure for configuring overrides that add the **request-block-host** or **request-block-connection** event actions to alerts of a specific risk rating, see [Adding, Editing, Enabling, and Disabling Event Action Overrides, page 8-18](#).
- For more information on Pre- and Post-Block ACLs, see [How the Sensor Manages Devices, page 14-21](#).

## Understanding Rate Limiting

The ARC is responsible for rate limiting traffic in protected networks. Rate limiting lets sensors restrict the rate of specified traffic classes on network devices. Rate limit responses are supported for the Host Flood and Net Flood engines, and the TCP half-open SYN signature. The ARC can configure rate limits on network devices running Cisco IOS 12.3 or later. Master blocking sensors can also forward rate limit requests to blocking forwarding sensors.

To add a rate limit, you specify the following:

- Source address and/or destination address for any rate limit
- Source port and/or destination port for rate limits with TCP or UDP protocol

You can also tune rate limiting signatures. You must also set the action to **request-rate-limit** and set the percentage for these signatures.

**Note**

Rate limiting and blocking are not supported for IPv6 traffic. If a signature is configured with a block or rate limit event action and is triggered by IPv6 traffic, an alert is generated but the action is not carried out.

[Table 14-1](#) lists the supported rate limiting signatures and parameters.

**Table 14-1** Rate Limiting Signatures

| Signature ID | Signature Name    | Protocol | Destination IP Address Allowed | Data         |
|--------------|-------------------|----------|--------------------------------|--------------|
| 2152         | ICMP Flood Host   | ICMP     | Yes                            | echo-request |
| 2153         | ICMP Smurf Attack | ICMP     | Yes                            | echo-reply   |

Table 14-1 Rate Limiting Signatures (continued)

| Signature ID | Signature Name         | Protocol | Destination IP Address Allowed | Data         |
|--------------|------------------------|----------|--------------------------------|--------------|
| 4002         | UDP Flood Host         | UDP      | Yes                            | none         |
| 6901         | Net Flood ICMP Reply   | ICMP     | No                             | echo-reply   |
| 6902         | Net Flood ICMP Request | ICMP     | No                             | echo-request |
| 6903         | Net Flood ICMP Any     | ICMP     | No                             | None         |
| 6910         | Net Flood UDP          | UDP      | No                             | None         |
| 6920         | Net Flood TCP          | TCP      | No                             | None         |
| 3050         | TCP HalfOpenSyn        | TCP      | No                             | halfOpenSyn  |



Tip

To check the status of the ARC, type **show statistics network-access** at the `sensor#`. The output shows the devices you are managing, any active blocks and rate limits, and the status of all devices..

#### For More Information

- For the procedure for configuring rate limiting on a router, see [Configuring Blocking and Rate Limiting Devices, page 14-21](#).
- For the procedure for configuring a sensor to be a master blocking sensor, see [Configuring the Sensor to be a Master Blocking Sensor, page 14-28](#).

## Understanding Service Policies for Rate Limiting

You must not apply a service policy to an interface/direction that is configured for rate limiting. If you do so, the rate limit action will fail. Before configuring rate limits, confirm that there is no service policy on the interface/direction, and remove it if one exists. The ARC does not remove the existing rate limit unless it is one that the ARC had previously added.

Rate limits use ACLs, but not in the same way as blocks. Rate limits use **acls** and **class-map** entries to identify traffic, and **policy-map** and **service-policy** entries to police the traffic.

## Before Configuring ARC



Caution

Two sensors cannot control blocking or rate limiting on the same device. If this situation is needed, configure one sensor as the master blocking sensor to manage the devices and the other sensors can forward their requests to the master blocking sensor.



Note

When you add a master blocking sensor, you reduce the number of blocking devices per sensor. For example, if you want to block on 10 security appliances and 10 routers with one blocking interface/direction each, you can assign 10 to the sensor and assign the other 10 to a master blocking sensor.

Before you configure the ARC for blocking or rate limiting, make sure you do the following:

- Analyze your network topology to understand which devices should be blocked by which sensor, and which addresses should never be blocked.
- Gather the usernames, device passwords, enable passwords, and connections types (Telnet or SSH) needed to log in to each device.
- Know the interface names on the devices.
- Know the names of the Pre-Block ACL or VACL and the Post-Block ACL or VACL if needed.
- Understand which interfaces should and should not be blocked and in which direction (in or out). You do not want to accidentally shut down an entire network.

#### For More Information

For the procedure for configuring the master blocking sensor, see [Configuring the Sensor to be a Master Blocking Sensor, page 14-28](#).

## Supported Devices



#### Caution

If the recommended limits are exceeded, the ARC may not apply blocks in a timely manner or may not be able to apply blocks at all.

By default, the ARC supports up to 250 devices in any combination. The following devices are supported for blocking by the ARC:

- Cisco series routers using Cisco IOS 11.2 or later (ACLs):
  - Cisco 1600 series router
  - Cisco 1700 series router
  - Cisco 2500 series router
  - Cisco 2600 series router
  - Cisco 2800 series router
  - Cisco 3600 series router
  - Cisco 3800 series router
  - Cisco 7200 series router
  - Cisco 7500 series router
- Catalyst 5000 switches with RSM with IOS 11.2(9)P or later (ACLs)
- Catalyst 6500 switches and 7600 routers with IOS 12.1(13)E or later (ACLs)
- Catalyst 6500 switches 7600 routers with Catalyst software version 7.5(1) or later (VACLs)
  - Supervisor Engine 1A with PFC
  - Supervisor Engine 1A with MSFC1
  - Supervisor Engine 1A with MFSC2
  - Supervisor Engine 2 with MSFC2
  - Supervisor Engine 720 with MSFC3



**Note**

We support VACL blocking on the Supervisor Engine and ACL blocking on the MSFC.

- PIX Firewall with version 6.0 or later (**shun** command)
  - 501
  - 506E
  - 515E
  - 525
  - 535
- ASA with version 7.0 or later (**shun** command)
  - ASA 5510
  - ASA 5520
  - ASA 5540
- FWSM 1.1 or later (**shun** command)

You configure blocking using either ACLs, VACLs, or the **shun** command. All firewall and ASA models support the **shun** command.

The following devices are supported for rate limiting by the ARC:

- Cisco series routers using Cisco IOS 12.3 or later:
  - Cisco 1700 series router
  - Cisco 2500 series router
  - Cisco 2600 series router
  - Cisco 2800 series router
  - Cisco 3600 series router
  - Cisco 3800 series router
  - Cisco 7200 series router
  - Cisco 7500 series router

**Caution**

The ARC cannot perform rate limits on 7500 routers with VIP. The ARC reports the error but cannot rate limit.

## Configuring Blocking Properties

You can change the default blocking properties. It is best to use the default properties, but if you need to change them, use the following procedures:

- [Allowing the Sensor to Block Itself, page 14-8](#)
- [Disabling Blocking, page 14-9](#)
- [Specifying Maximum Block Entries, page 14-11](#)
- [Specifying the Block Time, page 14-13](#)
- [Enabling ACL Logging, page 14-14](#)

- [Enabling Writing to NVRAM, page 14-15](#)
- [Logging All Blocking Events and Errors, page 14-16](#)
- [Configuring the Maximum Number of Blocking Interfaces, page 14-17](#)
- [Configuring Addresses Never to Block, page 14-19](#)

## Allowing the Sensor to Block Itself



### Caution

We recommend that you do not permit the sensor to block itself, because it may stop communicating with the blocking device. You can configure this option if you can ensure that if the sensor creates a rule to block its own IP address, it will not prevent the sensor from accessing the blocking device.

Use the **allow-sensor-block {true | false}** command in the service network access submode to configure the sensor to block itself. To allow the sensor to block itself, follow these steps:

**Step 1** Log in to the CLI using an account with administrator privileges.

**Step 2** Enter network access submode.

```
sensor# configure terminal
sensor(config)# service network-access
```

**Step 3** Enter general submode.

```
sensor(config-net)# general
```

**Step 4** Configure the sensor to block itself. By default, this value is false.

```
sensor(config-net-gen)# allow-sensor-block true
```

**Step 5** Verify the settings.

```
sensor(config-net-gen)# show settings
general

log-all-block-events-and-errors: true <defaulted>
enable-nvram-write: false <defaulted>
enable-acl-logging: false <defaulted>
allow-sensor-block: true default: false
block-enable: true default: true
block-max-entries: 100 default: 250
max-interfaces: 250 <defaulted>
master-blocking-sensors (min: 0, max: 100, current: 0)

never-block-hosts (min: 0, max: 250, current: 1)

ip-address: 192.0.2.1

never-block-networks (min: 0, max: 250, current: 1)

ip-address: 209.165.200.224/27

block-hosts (min: 0, max: 250, current: 0)

--MORE--
```

**Step 6** Configure the sensor not to block itself.

```
sensor(config-net-gen)# allow-sensor-block false
```

**Step 7** Verify the setting.

```
sensor(config-net-gen)# show settings
general

log-all-block-events-and-errors: true <defaulted>
enable-nvram-write: false <defaulted>
enable-acl-logging: false <defaulted>
allow-sensor-block: false default: false
block-enable: true default: true
block-max-entries: 100 default: 250
max-interfaces: 250 <defaulted>
master-blocking-sensors (min: 0, max: 100, current: 0)

never-block-hosts (min: 0, max: 250, current: 1)

ip-address: 192.0.2.1

never-block-networks (min: 0, max: 250, current: 1)

ip-address: 209.165.200.224/27

block-hosts (min: 0, max: 250, current: 0)

--MORE--
```

**Step 8** Exit network access submode.

```
sensor(config-net-gen)# exit
sensor(config-net)# exit
Apply Changes?[yes]:
```

**Step 9** Press **Enter** to apply the changes or enter **no** to discard them.

## Disabling Blocking



### Note

For blocking to operate, you must set up devices to do the blocking.

Use the **block-enable {true | false}** command in the service network access submode to enable or disable blocking on the sensor. By default, blocking is enabled on the sensor. If the ARC is managing a device and you need to manually configure something on that device, you should disable blocking first. You want to avoid a situation in which both you and the ARC could be making a change at the same time on the same device. This could cause the device and/or the ARC to crash.



### Caution

If you disable blocking for maintenance on the devices, make sure you enable it after the maintenance is complete or the network will be vulnerable to attacks that would otherwise be blocked.

**Note**

While blocking is disabled, the ARC continues to receive blocks and track the time on active blocks, but will not apply new blocks or remove blocks from the managed devices. After blocking is reenabled, the blocks on the devices are updated.

To disable blocking or rate limiting, follow these steps:

**Step 1** Log in to the CLI using an account with administrator privileges.

**Step 2** Enter network access submode.

```
sensor# configure terminal
sensor(config)# service network-access
sensor(config-net)#
```

**Step 3** Enter general submode.

```
sensor(config-net)# general
```

**Step 4** Disable blocking on the sensor. By default, this value is set to true.

```
sensor(config-net-gen)# block-enable false
```

**Step 5** Verify the settings.

```
sensor(config-net-gen)# show settings
general

log-all-block-events-and-errors: true <defaulted>
enable-nvram-write: false <defaulted>
enable-acl-logging: false <defaulted>
allow-sensor-block: false default: false
block-enable: false default: true
block-max-entries: 100 default: 250
max-interfaces: 250 <defaulted>
master-blocking-sensors (min: 0, max: 100, current: 0)

never-block-hosts (min: 0, max: 250, current: 1)

ip-address: 192.0.2.1

never-block-networks (min: 0, max: 250, current: 1)

ip-address: 209.165.200.224/27

block-hosts (min: 0, max: 250, current: 0)

--MORE--
```

**Step 6** Enable blocking on the sensor.

```
sensor(config-net-gen)# block-enable true
```

**Step 7** Verify that the setting has been returned to the default.

```
sensor(config-net-gen)# show settings
general

log-all-block-events-and-errors: true <defaulted>
enable-nvram-write: false <defaulted>
```

```

enable-acl-logging: false <defaulted>
allow-sensor-block: false default: false
block-enable: true default: true
block-max-entries: 100 default: 250
max-interfaces: 250 <defaulted>
master-blocking-sensors (min: 0, max: 100, current: 0)

never-block-hosts (min: 0, max: 250, current: 1)

 ip-address: 192.0.2.1

never-block-networks (min: 0, max: 250, current: 1)

 ip-address: 209.165.200.224/27

block-hosts (min: 0, max: 250, current: 0)

--MORE--

```

**Step 8** Exit network access submode.

```

sensor(config-net-gen)# exit
sensor(config-net)# exit
Apply Changes?[yes]:

```

**Step 9** Press **Enter** to apply the changes or enter **no** to discard them.

#### For More Information

- For the procedure for configuring the sensor to manage Cisco routers, see [Configuring the Sensor to Manage Cisco Routers, page 14-23](#).
- For the procedure for configuring the sensor to manage Cisco routers and switches, see [Configuring the Sensor to Manage Catalyst 6500 Series Switches and Cisco 7600 Series Routers, page 14-25](#).

## Specifying Maximum Block Entries



#### Caution

We do not recommend setting the maximum block entries higher than 250. Some devices have problems with larger numbers of ACL or shun entries. Refer to the documentation for each device to determine its limits before increasing this number.



#### Note

The number of blocks will not exceed the maximum block entries. If the maximum is reached, new blocks will not occur until existing blocks time out and are removed.

Use the **block-max-entries** command in the service network access submode to configure the maximum block entries. You can set how many blocks are to be maintained simultaneously (1 to 65535). The default value is 250.

To change the maximum number of block entries, follow these steps:

**Step 1** Log in to the CLI using an account with administrator privileges.

**Step 2** Enter network access submode.

```
sensor# configure terminal
sensor(config)# service network-access
sensor(config-net)#
```

**Step 3** Enter general submode.

```
sensor(config-net)# general
```

**Step 4** Change the maximum number of block entries.

```
sensor(config-net-gen)# block-max-entries 100
```

**Step 5** Verify the setting.

```
sensor(config-net-gen)# show settings
general

log-all-block-events-and-errors: true <defaulted>
enable-nvram-write: false <defaulted>
enable-acl-logging: false <defaulted>
allow-sensor-block: false default: false
block-enable: true <defaulted>
block-max-entries: 100 default: 250
max-interfaces: 250 <defaulted>
master-blocking-sensors (min: 0, max: 100, current: 0)

never-block-hosts (min: 0, max: 250, current: 1)

 ip-address: 192.0.2.1

never-block-networks (min: 0, max: 250, current: 1)

 ip-address: 209.165.200.224/27

block-hosts (min: 0, max: 250, current: 0)

--MORE--
```

**Step 6** Return to the default value of 250 blocks.

```
sensor(config-net-gen)# default block-max-entries
```

**Step 7** Verify the setting.

```
sensor(config-net-gen)# show settings
general

log-all-block-events-and-errors: true <defaulted>
enable-nvram-write: false <defaulted>
enable-acl-logging: false <defaulted>
allow-sensor-block: false default: false
block-enable: true <defaulted>
block-max-entries: 250 <defaulted>
max-interfaces: 250 <defaulted>
master-blocking-sensors (min: 0, max: 100, current: 0)

```

```

never-block-hosts (min: 0, max: 250, current: 1)

 ip-address: 192.0.2.1

never-block-networks (min: 0, max: 250, current: 1)

 ip-address: 209.165.200.224/27

block-hosts (min: 0, max: 250, current: 0)

--MORE--

```

**Step 8** Exit network access submode.

```

sensor(config-net-gen)# exit
sensor(config-net)# exit
Apply Changes?[yes]:

```

**Step 9** Press **Enter** to apply the changes or enter **no** to discard them.

## Specifying the Block Time



**Note** If you change the default block time, you are changing a signature parameter, which affects all signatures.



**Note** The time for manual blocks is set when you request the block.

Use the **global-block-timeout** command in the service event action rules submode to change the amount of time an automatic block lasts. The default is 30 minutes.

To change the default block time, follow these steps:

**Step 1** Log in to the CLI using an account with administrator privileges.

**Step 2** Enter event action rules submode.

```

sensor# configure terminal
sensor(config)# service event-action-rules rules0
sensor(config-rul)#

```

**Step 3** Enter general submode.

```

sensor(config-rul)# general

```

**Step 4** Specify the block time. The value is the time duration of the block event in minutes (0 to 10000000).

```

sensor(config-rul-gen)# global-block-timeout 60

```

**Step 5** Verify the setting.

```

sensor(config-rul-gen)# show settings
general

```

```

global-overrides-status: Enabled <defaulted>
global-filters-status: Enabled <defaulted>
global-summarization-status: Enabled <defaulted>
global-metaevent-status: Enabled <defaulted>
global-deny-timeout: 3600 <defaulted>
global-block-timeout: 60 default: 30
max-denied-attackers: 10000 <defaulted>

sensor(config-rul-gen)#

```

**Step 6** Exit event action rules submode.

```

sensor(config-rul-gen)# exit
sensor(config-rul)# exit
Apply Changes:[yes]:

```

**Step 7** Press **Enter** to apply the changes or enter **no** to discard them.




---

**Note** There is a time delay while the signatures are updated.

---

## Enabling ACL Logging

Use the **enable-acl-logging {true | false}** command in the service network access submode to enable ACL logging, which causes ARC to append the log parameter to block entries in the ACL or VACL. This causes the device to generate syslog events when packets are filtered. Enable ACL logging only applies to routers and switches. The default is disabled.

To enable ACL logging, follow these steps:

---

**Step 1** Log in to the CLI using an account with administrator privileges.

**Step 2** Enter network access submode:

```

sensor# configure terminal
sensor(config)# service network-access
sensor(config-net)#

```

**Step 3** Enter general submode.

```

sensor(config-net)# general

```

**Step 4** Enable ACL logging.

```

sensor(config-net-gen)# enable-acl-logging true

```

**Step 5** Verify that ACL logging is enabled.

```

sensor(config-net-gen)# show settings
general

log-all-block-events-and-errors: true <defaulted>
enable-nvram-write: false <defaulted>
enable-acl-logging: true default: false
allow-sensor-block: false <defaulted>
block-enable: true <defaulted>
block-max-entries: 250 <defaulted>

```



```
max-interfaces: 250 <defaulted>
master-blocking-sensors (min: 0, max: 100, current: 0)

```

**Step 6** Disable ACL logging by using the **false** keyword.

```
sensor(config-net-gen)# enable-acl-logging false
```

**Step 7** Verify that ACL logging is disabled.

```
sensor(config-net-gen)# show settings
general

log-all-block-events-and-errors: true <defaulted>
enable-nvram-write: false <defaulted>
enable-acl-logging: false default: false
allow-sensor-block: false <defaulted>
block-enable: true <defaulted>
block-max-entries: 250 <defaulted>
max-interfaces: 250 <defaulted>
master-blocking-sensors (min: 0, max: 100, current: 0)

```

**Step 8** Exit network access mode.

```
sensor(config-net-gen)# exit
sensor(config-net)# exit
Apply Changes:[yes]:
```

**Step 9** Press **Enter** to apply the changes or enter **no** to discard them.

## Enabling Writing to NVRAM

Use the **enable-nvram-write {true | false}** command to configure the sensor to have the router write to NVRAM when ARC first connects. If **enable-nvram-write** is enabled, NVRAM is written each time the ACLs are updated. The default is disabled.

Enabling NVRAM writing ensures that all changes for blocking are written to NVRAM. If the router is rebooted, the correct blocks will still be active. If NVRAM writing is disabled, a short time without blocking occurs after a router reboot. And not enabling NVRAM writing increases the life of the NVRAM and decreases the time for new blocks to be configured.

To enable writing to NVRAM, follow these steps:

**Step 1** Log in to the CLI using an account with administrator privileges.

**Step 2** Enter network access submode.

```
sensor# configure terminal
sensor(config)# service network-access
sensor(config-net)#
```

**Step 3** Enter general submode.

```
sensor(config-net)# general
```

**Step 4** Enable writing to NVRAM.

```
sensor(config-net-gen)# enable-nvram-write true
```

**Step 5** Verify that writing to NVRAM is enabled.

```
sensor(config-net-gen)# show settings
general

log-all-block-events-and-errors: true <defaulted>
enable-nvram-write: true default: false
enable-acl-logging: false default: false
allow-sensor-block: false <defaulted>
block-enable: true <defaulted>
block-max-entries: 250 <defaulted>
max-interfaces: 250 <defaulted>
master-blocking-sensors (min: 0, max: 100, current: 0)

```

**Step 6** Disable writing to NVRAM.

```
sensor(config-net-gen)# enable-nvram-write false
```

**Step 7** Verify that writing to NVRAM is disabled.

```
sensor(config-net-gen)# show settings
general

log-all-block-events-and-errors: true <defaulted>
enable-nvram-write: false default: false
enable-acl-logging: false default: false
allow-sensor-block: false <defaulted>
block-enable: true <defaulted>
block-max-entries: 250 <defaulted>
max-interfaces: 250 <defaulted>
master-blocking-sensors (min: 0, max: 100, current: 0)

```

**Step 8** Exit network access submode.

```
sensor(config-net-gen)# exit
sensor(config-net)# exit
Apply Changes:[yes]:
```

**Step 9** Press **Enter** to apply the changes or enter **no** to discard them.

---

## Logging All Blocking Events and Errors

Use the **log-all-block-events-and-errors {true | false}** command in the service network access submode to configure the sensor to log events that follow blocks from start to finish. For example, when a block is added to or removed from a device, an event is logged. You may not want all these events and errors to be logged. Disabling **log-all-block-events-and-errors** suppresses the new events and errors. The default is enabled.

To disable blocking event and error logging, follow these steps:

---

**Step 1** Log in to the CLI using an account with administrator privileges.

**Step 2** Enter network access mode.

```
sensor# configure terminal
sensor(config)# service network-access
sensor(config-net)#
```

**Step 3** Enter general submenu.

```
sensor(config-net)# general
```

**Step 4** Disable blocking event and error logging.

```
sensor(config-net-gen)# log-all-block-events-and-errors false
```

**Step 5** Verify that logging is disabled.

```
sensor(config-net-gen)# show settings
general

log-all-block-events-and-errors: false default: true
enable-nvram-write: false default: false
enable-acl-logging: false default: false
allow-sensor-block: false <defaulted>
block-enable: true <defaulted>
block-max-entries: 250 <defaulted>
max-interfaces: 250 <defaulted>
master-blocking-sensors (min: 0, max: 100, current: 0)

```

**Step 6** Enable blocking event and error logging.

```
sensor(config-net-gen)# log-all-block-events-and-errors true
```

**Step 7** Verify that logging is enabled.

```
sensor(config-net-gen)# show settings
general

log-all-block-events-and-errors: true default: true
enable-nvram-write: false default: false
enable-acl-logging: false default: false
allow-sensor-block: false <defaulted>
block-enable: true <defaulted>
block-max-entries: 250 <defaulted>
max-interfaces: 250 <defaulted>
master-blocking-sensors (min: 0, max: 100, current: 0)

```

**Step 8** Exit network access mode.

```
sensor(config-net-gen)# exit
sensor(config-net)# exit
Apply Changes:[yes]:
```

**Step 9** Press **Enter** to apply the changes or type **no** to discard them.

## Configuring the Maximum Number of Blocking Interfaces

Use the **max-interfaces** command to configure the maximum number of interfaces for performing blocks. For example, a PIX Firewall counts as one interface. A router with one interface counts as one, but a router with two interfaces counts as two. At most you can configure 250 blocking interfaces on a router, switch, or firewall. You can configure up to 250 Catalyst 6K switches, 250 routers, and 250 firewalls.

The **max-interfaces** command configures the limit of the sum total of all interfaces and devices. In addition to configuring the limit on the sum total of interfaces and devices, there is a fixed limit on the number of blocking interfaces you can configure per device. Use the **show settings** command in network access mode to view the specific maximum limits per device.

To configure the maximum number of blocking interfaces, follow these steps:

**Step 1** Log in to the CLI using an account with administrator privileges.

**Step 2** Enter network access mode.

```
sensor# configure terminal
sensor(config)# service network-access
sensor(config-net)#
```

**Step 3** Enter general submodule.

```
sensor(config-net)# general
```

**Step 4** Specify the maximum number of interfaces.

```
sensor(config-net-gen)# max-interfaces 50
```

**Step 5** Verify the number of maximum interfaces.

```
sensor(config-net-gen)# show settings
general

log-all-block-events-and-errors: true default: true
enable-nvram-write: false default: false
enable-acl-logging: false default: false
allow-sensor-block: false <defaulted>
block-enable: true <defaulted>
block-max-entries: 250 <defaulted>
max-interfaces: 50 default: 250
master-blocking-sensors (min: 0, max: 100, current: 0)

```

**Step 6** Return the setting to the default of 250.

```
sensor(config-net-gen)# default max-interfaces
```

**Step 7** Verify the default setting.

```
sensor(config-net-gen)# show settings
general

log-all-block-events-and-errors: true default: true
enable-nvram-write: false default: false
enable-acl-logging: false default: false
allow-sensor-block: false <defaulted>
block-enable: true <defaulted>
block-max-entries: 250 <defaulted>
max-interfaces: 250 <defaulted>
master-blocking-sensors (min: 0, max: 100, current: 0)

```

**Step 8** Exit network access mode.

```
sensor(config-net-gen)# exit
sensor(config-net)# exit
Apply Changes:[yes]:
```

**Step 9** Press **Enter** to apply the changes or enter **no** to discard them.

## Configuring Addresses Never to Block

Use the **never-block-hosts** and the **never-block-networks** commands in the service network access submode to configure hosts and network that should never be blocked.

The following commands apply:

- *ip\_address*—Specifies the IP address of the device that should never be blocked.
- *ip\_address/netmask*—Specifies the IP address of the network that should never be blocked. The format is A.B.C.D/nn.

You must tune your sensor to identify hosts and networks that should never be blocked, not even manually, because you may have a trusted network device whose normal, expected behavior appears to be an attack. Such a device should never be blocked, and trusted, internal networks should never be blocked. You can specify a single host or an entire network.



### Note

The **never-block-hosts** and the **never-block-networks** commands apply only to the Request Block Host and Request Block Connection event actions. It does not apply to the Deny Attacker Inline, Deny Connection Inline, or Deny Packet Inline event actions. Use event action rules to filter out the hosts that you do not want blocked, denied, or dropped.

### Configuring Addresses Never to Be Blocked

To set up addresses never to be blocked by blocking devices, follow these steps:

**Step 1** Log in to the CLI using an account with administrator privileges.

**Step 2** Enter network access submode.

```
sensor# configure terminal
sensor(config)# service network-access
sensor(config-net)#
```

**Step 3** Enter general submode.

```
sensor(config-net)# general
```

**Step 4** Specify the address that should never be blocked:

- For a single host
 

```
sensor(config-net-gen)# never-block-hosts 192.0.2.1
```
- For a network
 

```
sensor(config-net-gen)# never-block-networks 209.165.200.224/27
```

**Step 5** Verify the settings.

```
sensor(config-net-gen)# show settings
general

log-all-block-events-and-errors: true <defaulted>
enable-nvram-write: false <defaulted>
enable-acl-logging: false <defaulted>
```

```

allow-sensor-block: false default: false
block-enable: true default: true
block-max-entries: 100 default: 250
max-interfaces: 250 <defaulted>
master-blocking-sensors (min: 0, max: 100, current: 0)

never-block-hosts (min: 0, max: 250, current: 2)

 ip-address: 192.0.2.1

never-block-networks (min: 0, max: 250, current: 2)

 ip-address: 209.165.200.224/27
--MORE--

```

**Step 6** Exit network access submode.

```

sensor(config-net-gen)# exit
sensor(config-net)# exit
Apply Changes?[yes]:

```

**Step 7** Press **Enter** to apply the changes or enter **no** to discard them.

#### For More Information

For the procedure for configuring event action filters, see [Configuring Event Action Filters, page 8-21](#).

## Configuring User Profiles



#### Note

If the username or password is not needed to log in to the device, do not set a value for it.



#### Note

You **MUST** create a user profile before configuring the blocking device.

Use the **user-profiles** *profile\_name* command in the service network access submode to set up user profiles for the other devices that the sensor will manage. The user profiles contain userid, password, and enable password information. For example, routers that all share the same passwords and usernames can be under one user profile.

To set up user profiles, follow these steps:

**Step 1** Log in to the CLI using an account with administrator privileges.

**Step 2** Enter network access mode.

```

sensor# configure terminal
sensor(config)# service network-access
sensor(config-net)#

```

**Step 3** Create the user profile name.

```

sensor(config-net)# user-profiles PROFILE1

```

**Step 4** Enter the username for that user profile.

```
sensor(config-net-use)# username username
```

**Step 5** Specify the password for the user.

```
sensor(config-net-use)# password
Enter password[:] *****
Re-enter password *****
```

**Step 6** Specify the enable password for the user.

```
sensor(config-net-use)# enable-password
Enter enable-password[:] *****
Re-enter enable-password *****
```

**Step 7** Verify the settings.

```
sensor(config-net-use)# show settings
profile-name: PROFILE1

enable-password: <hidden>
password: <hidden>
username: jsmith default:

sensor(config-net-use)#
```

**Step 8** Exit network access submode.

```
sensor(config-net-use)# exit
sensor(config-net)# exit
Apply Changes:[yes]:
```

**Step 9** Press **Enter** to apply the changes or enter **no** to discard them.

## Configuring Blocking and Rate Limiting Devices

This section describes how to configure devices that the sensor uses to perform blocking or rate limiting. It contains the following topics:

- [How the Sensor Manages Devices, page 14-21](#)
- [Configuring the Sensor to Manage Cisco Routers, page 14-23](#)
- [Configuring the Sensor to Manage Catalyst 6500 Series Switches and Cisco 7600 Series Routers, page 14-25](#)
- [Configuring the Sensor to Manage Cisco Firewalls, page 14-27](#)

## How the Sensor Manages Devices



**Note**

ACLs do not apply to rate limiting devices.

The ARC uses ACLs on Cisco routers and switches to manage those devices. These ACLs are built as follows:

1. A **permit** line with the sensor IP address or, if specified, the NAT address of the sensor.




---

**Note** If you permit the sensor to be blocked, this line does not appear in the ACL.

---

2. Pre-Block ACL (if specified). This ACL must already exist on the device.




---

**Note** The ARC reads the lines in the ACL and copies these lines to the beginning of the ACL.

---

3. Any active blocks.
4. Either specify a Post-Block ACL, which must already exist on the device, or specify **permit ip any any** (do not use if a Post-Block ACL is specified). The ARC reads the lines in the ACL and copies these lines to the end of the ACL.




---

**Note** Make sure the last line in the ACL is **permit ip any any** if you want all unmatched packets to be permitted.

---

The ARC uses two ACLs to manage devices. Only one is active at any one time. It uses the offline ACL name to build the new ACL, then applies it to the interface. The ARC then reverses the process on the next cycle.



**Caution**

---

The ACLs that the ARC makes should never be modified by you or any other system. These ACLs are temporary and new ACLs are constantly being created by the sensor. The only modifications that you can make are to the Pre- and Post-Block ACLs.

---

If you need to modify the Pre-Block or Post-Block ACL, do the following:

1. Disable blocking on the sensor.
2. Make the changes to the configuration of the device.
3. Reenable blocking on the sensor.

When blocking is reenabled, the sensor reads the new device configuration.



**Caution**

---

A single sensor can manage multiple devices, but you cannot use multiple sensors to control a single device. In this case, use a master blocking sensor.

---

**For More Information**

- For the procedure for enabling blocking, see [Configuring Blocking Properties, page 14-7](#).
- For the procedure for configuring the sensor to be a master blocking sensor, see [Configuring the Sensor to be a Master Blocking Sensor, page 14-28](#).



## Configuring the Sensor to Manage Cisco Routers

This section describes how to configure the sensor to manage Cisco routers. It contains the following topics:

- [Routers and ACLs, page 14-23](#)
- [Configuring the Sensor to Manage Cisco Routers, page 14-23](#)

### Routers and ACLs

**Note**

---

Pre-Block and Post-Block ACLs do not apply to rate limiting.

---

You create and save Pre-Block and Post-Block ACLs in your router configuration. These ACLs must be extended IP ACLs, either named or numbered. See your router documentation for more information on creating ACLs. Enter the names of these ACLs that are already configured on your router in the Pre-Block ACL and Post-Block ACL fields.

The Pre-Block ACL is mainly used for permitting what you do not want the sensor to ever block. When a packet is checked against the ACL, the first line that gets matched determines the action. If the first line matched is a permit line from the Pre-Block ACL, the packet is permitted even though there may be a deny line (from an automatic block) listed later in the ACL. The Pre-Block ACL can override the deny lines resulting from the blocks.

The Post-Block ACL is best used for additional blocking or permitting that you want to occur on the same interface or direction. If you have an existing ACL on the interface or direction that the sensor will manage, that existing ACL can be used as a Post-Block ACL. If you do not have a Post-Block ACL, the sensor inserts **permit ip any any** at the end of the new ACL.

When the sensor starts up, it reads the contents of the two ACLs. It creates a third ACL with the following entries:

- A **permit** line for the sensor IP address
- Copies of all configuration lines of the Pre-Block ACL
- A **deny** line for each address being blocked by the sensor
- Copies of all configuration lines of the Post-Block ACL

The sensor applies the new ACL to the interface and direction that you designate.

**Note**

---

When the new ACL is applied to an interface or direction of the router, it removes the application of any other ACL to that interface or direction.

---

## Configuring the Sensor to Manage Cisco Routers

To configure a sensor to manage a Cisco router to perform blocking and rate limiting, follow these steps:

---

**Step 1** Log in to the CLI using an account with administrator privileges.

**Step 2** Enter network access submode.

```
sensor# configure terminal
sensor(config)# service network-access
```

```
sensor(config-net)#
```

**Step 3** Specify the IP address for the router controlled by the ARC.

```
sensor(config-net)# router-devices ip_address
```

**Step 4** Enter the logical device name that you created when you configured the user profile. The ARC accepts anything you enter. It does not check to see if the user profile exists.

```
sensor(config-net-rou)# profile-name user_profile_name
```

**Step 5** Specify the method used to access the sensor. If unspecified, SSH 3DES is used.

```
sensor(config-net-rou)# communication {telnet | ssh-3des}
```



**Note** If you are using 3DES, you must use the command `ssh host-key ip_address` to accept the key or ARC cannot connect to the device.

**Step 6** Specify the sensor NAT address.

```
sensor(config-net-rou)# nat-address nat_address
```



**Note** This changes the IP address in the first line of the ACL from the address of the sensor to the NAT address. This is not a NAT address configured on the device being managed. It is the address the sensor is translated to by an intermediate device, one that is between the sensor and the device being managed.

**Step 7** Specify whether the router will perform blocking, rate limiting, or both.



**Note** The default is blocking. You do not have to configure response capabilities if you want the router to perform blocking only.

a. Rate limiting only

```
sensor(config-net-rou)# response-capabilities rate-limit
```

b. Both blocking and rate limiting

```
sensor(config-net-rou)# response-capabilities block|rate-limit
```

**Step 8** Specify the interface name and direction.

```
sensor(config-net-rou)# block-interfaces interface_name {in | out}
```



**Caution**

The name of the interface must either be the complete name of the interface or an abbreviation that the router recognizes with the `interface` command.

**Step 9** (Optional) Add the pre-ACL name (blocking only).

```
sensor(config-net-rou-blo)# pre-acl-name pre_acl_name
```

**Step 10** (Optional) Add the post-ACL name (blocking only).

```
sensor(config-net-rou-blo)# post-acl-name post_acl_name
```

**Step 11** Verify the settings.

```

sensor(config-net-rou-blo)# exit
sensor(config-net-rou)# show settings
 ip-address: 192.0.2.1

 communication: ssh-3des default: ssh-3des
 nat-address: 19.89.149.219 default: 0.0.0.0
 profile-name: PROFILE1
 block-interfaces (min: 0, max: 100, current: 1)

 interface-name: GigabitEthernet0/1
 direction: in

 pre-acl-name: <defaulted>
 post-acl-name: <defaulted>

 response-capabilities: block|rate-limit default: block

sensor(config-net-rou)#

```

**Step 12** Exit network access submode.

```

sensor(config-net-rou)# exit
sensor(config-net)# exit
sensor(config)# exit
Apply Changes?[yes]:

```

**Step 13** Press **Enter** to apply the changes or enter **no** to discard them.

#### For More Information

- For the procedure for configuring user profiles, see [Configuring User Profiles, page 14-20](#).
- For the procedure for adding a device to the known hosts list, see [Adding Hosts to the SSH Known Hosts List, page 4-47](#).

## Configuring the Sensor to Manage Catalyst 6500 Series Switches and Cisco 7600 Series Routers

This section describes how to configure the sensor to manage Cisco switches. It contains the following topics:

- [Switches and VACLs, page 14-25](#)
- [Configuring the Sensor to Manage Catalyst 6500 Series Switches and Cisco 7600 Series Routers, page 14-26](#)

### Switches and VACLs

You can configure the ARC to block using VACLs on the switch itself when running Cisco Catalyst software, or to block using router ACLs on the MSFC or on the switch itself when running Cisco IOS software. This section describes blocking using VACLs. You cannot configure switches that use VACLs to perform rate limiting. You must configure the blocking interfaces on the Catalyst 6500 series switch and specify the VLAN of traffic you want blocked.

You create and save Pre-Block and Post-Block VACLs in your switch configuration. These VACLs must be extended IP VACLs, either named or numbered. See your switch documentation for more information on creating VACLs. Enter the names of these VACLs that are already configured on your switch in the Pre-Block VACL and Post-Block VACL fields.

The Pre-Block VACL is used mainly for permitting what you do not want the sensor to ever block. When a packet is checked against the VACL, the first line that gets matched determines the action. If the first line matched is a permit line from the Pre-Block VACL, the packet is permitted even though there may be a deny line (from an automatic block) listed later in the VACL. The Pre-Block VACL can override the deny lines resulting from the blocks.

The Post-Block VACL is best used for additional blocking or permitting that you want to occur on the same VLAN. If you have an existing VACL on the VLAN that the sensor will manage, the existing VACL can be used as a Post-Block VACL. If you do not have a Post-Block VACL, the sensor inserts **permit ip any any** at the end of the new VACL.

When the sensor starts up, it reads the contents of the two VACLs. It creates a third VACL with the following entries:

- A **permit** line for the sensor IP address
- Copies of all configuration lines of the Pre-Block VACL
- A **deny** line for each address being blocked by the sensor
- Copies of all configuration lines of the Post-Block VACL

The sensor applies the new VACL to the VLAN that you designate.

**Note**

When the new VACL is applied to a VLAN of the switch, it removes the application of any other VACL to that VLAN.

**For More Information**

For the procedure for configuring blocking using router ACLs, see [Configuring Blocking and Rate Limiting Devices, page 14-21](#).

## Configuring the Sensor to Manage Catalyst 6500 Series Switches and Cisco 7600 Series Routers

To configure the sensor to manage Catalyst 6500 series switches and Cisco 7600 series routers, follow these steps:

**Step 1** Log in to the CLI using an account with administrator privileges.

**Step 2** Enter network access submode.

```
sensor# configure terminal
sensor(config)# service network-access
sensor(config-net)#
```

**Step 3** Specify the IP address for the router controlled by the ARC.

```
sensor(config-net)# cat6k-devices ip_address
```

**Step 4** Enter the user profile name that you created when you configured the user profile. The ARC accepts anything you type. It does not accept it, check to see if the logical device exists.

```
sensor(config-net-cat)# profile-name user_profile_name
```

**Step 5** Specify the method used to access the sensor. If unspecified, SSH 3DES is used.

```
sensor(config-net-cat)# communication {telnet | ssh-3des}
```



**Note** If you are using 3DES, you must use the command `ssh host-key ip_address` to accept the key or ARC cannot connect to the device.

**Step 6** Specify the sensor NAT address.

```
sensor(config-net-cat)# nat-address nat_address
```



**Note** This changes the IP address in the first line of the ACL from the IP address of the sensor to the NAT address. This is not a NAT address configured on the device being managed. It is the address the sensor is translated to by an intermediate device, one that is between the sensor and the device being managed.

**Step 7** Specify the VLAN number.

```
sensor(config-net-cat)# block-vlans vlan_number
```

**Step 8** (Optional) Add the pre-VACL name.

```
sensor(config-net-cat-blo)# pre-vacl-name pre_vacl_name
```

**Step 9** (Optional) Add the post-VACL name.

```
sensor(config-net-cat-blo)# post-vacl-name post_vacl_name
```

**Step 10** Exit network access submode.

```
sensor(config-net-cat-blo)# exit
sensor(config-net-cat)# exit
sensor(config-net)# exit
sensor(config)# exit
Apply Changes:[yes]:
```

**Step 11** Press **Enter** to apply the changes or enter **no** to discard them.

#### For More Information

- For the procedure for configuring user profiles, see [Configuring User Profiles, page 14-20](#).
- For the procedure for adding a device to the known hosts list, see [Adding Hosts to the SSH Known Hosts List, page 4-47](#).

## Configuring the Sensor to Manage Cisco Firewalls

To configure the sensor to manage Cisco firewalls, follow these steps:

**Step 1** Log in to the CLI using an account with administrator privileges.

**Step 2** Enter network access submode.

```
sensor# configure terminal
sensor(config)# service network-access
```

```
sensor(config-net)#
```

**Step 3** Specify the IP address for the firewall controlled by the ARC.

```
sensor(config-net)# firewall-devices ip_address
```

**Step 4** Enter the user profile name that you created when you configured the user profile. ARC accepts anything you type. It does not check to see if the logical device exists.

```
sensor(config-net-fir)# profile-name user_profile_name
```

**Step 5** Specify the method used to access the sensor. If unspecified, SSH 3DES is used.

```
sensor(config-net-fir)# communication {telnet | ssh-3des}
```




---

**Note** If you are using 3DES, you must use the command `ssh host-key ip_address` to accept the key or the ARC cannot connect to the device.

---

**Step 6** Specify the sensor NAT address.

```
sensor(config-net-fir)# nat-address nat_address
```




---

**Note** This changes the IP address in the first line of the ACL from the IP address of the sensor to the NAT address. This is not a NAT address configured on the device being managed. It is the address the sensor is translated to by an intermediate device, one that is between the sensor and the device being managed.

---

**Step 7** Exit network access submode.

```
sensor(config-net-fir)# exit
sensor(config-net)# exit
sensor(config)# exit
Apply Changes:[yes]:
```

**Step 8** Press **Enter** to apply the changes or enter **no** to discard them.

---

#### For More Information

- For the procedure for configuring user profiles, see [Configuring User Profiles, page 14-20](#).
- For the procedure for adding a device to the known hosts list, see [Adding Hosts to the SSH Known Hosts List, page 4-47](#).

## Configuring the Sensor to be a Master Blocking Sensor

Multiple sensors (blocking forwarding sensors) can forward blocking requests to a specified master blocking sensor, which controls one or more devices. The master blocking sensor is the ARC running on a sensor that controls blocking on one or more devices on behalf of one or more other sensors. The ARC on a master blocking sensor controls blocking on devices at the request of the ARCs running on other sensors. Master blocking sensors can also forward rate limits.

**Caution**

Two sensors cannot control blocking or rate limiting on the same device. If this situation is needed, configure one sensor as the master blocking sensor to manage the devices and the other sensors can forward their requests to the master blocking sensor.

When you add a master blocking sensor, you reduce the number of blocking devices per sensor. For example, if you want to block on 10 firewalls and 10 routers with one blocking interface/direction each, you can assign 10 to the sensor and assign the other 10 to a master blocking sensor.

On the blocking forwarding sensor, identify which remote host serves as the master blocking sensor; on the master blocking sensor you must add the blocking forwarding sensors to its access list.

If the master blocking sensor requires TLS for web connections, you must configure the ARC of the blocking forwarding sensor to accept the X.509 certificate of the master blocking sensor remote host. Sensors by default have TLS enabled, but you can change this option.

**Note**

Typically the master blocking sensor is configured to manage the network devices. Blocking forwarding sensors are not normally configured to manage other network devices, although doing so is permissible.

Even if you have no devices configured for blocking or rate limiting, a sensor that is configured for blocking or rate limiting can forward blocking and rate limiting requests to a master blocking sensor. When a signature fires that has blocking or rate limit requests configured as event actions, the sensor forwards the block or rate limit request to the master blocking sensor, which then performs the block or rate limit.

**Caution**

Only one sensor should control all blocking interfaces on a device.

Use the **master-blocking-sensors** *master\_blocking\_sensor\_ip\_address* command in the service network access submode to configure a master blocking sensor.

The following commands apply:

- **master\_blocking\_sensor\_ip\_address**—Specifies the IP address of sensor for forward block requests.
- **password**—Specifies the account password of sensor for forward block requests.
- **port**—Specifies the port of sensor for forward block requests.
- **tls {true | false}** —Set to true if the remote sensor requires TLS; otherwise, set to false.
- **username**—Specifies the account name of sensor for forward block requests.

**Configuring the Master Blocking Sensor**

To configure ARC on a sensor to forward blocks to a master blocking sensor, follow these steps:

**Step 1** Log in to the CLI using an account with administrator privileges on both the master blocking sensor and the blocking forwarding sensor.

**Step 2** Enter configuration mode on both sensors.

```
sensor# configure terminal
```

**Step 3** Configure TLS if necessary:

- a. On the master blocking sensor, check to see if it requires TLS and what port number is used. If `enable-tls` is true, go to Step b.

```
sensor(config)# service web-server
sensor(config-web)# show settings
 enable-tls: true <defaulted>
 port: 443 <defaulted>
 server-id: HTTP/1.1 compliant <defaulted>
sensor(config-web)#
```

- b. On the blocking forwarding sensor, configure it to accept the X.509 certificate of the master blocking sensor.

```
sensor(config-web)# exit
sensor(config)# tls trusted-host ip-address master_blocking_sensor_ip_address port
port_number
```

**Example**

```
sensor(config)# tls trusted-host ip-address 192.0.2.1 port 8080
Certificate MD5 fingerprint is
F4:4A:14:BA:84:F4:51:D0:A4:E2:15:38:7E:77:96:D8Certificate SHA1 fingerprint is
84:09:B6:85:C5:43:60:5B:37:1E:6D:31:6A:30:5F:7E:4D:4D:E8:B2
Would you like to add this to the trusted certificate table for this host?[yes]:
```

**Note**

You are prompted to accept the certificate based on the certificate fingerprint. Sensors provide only self-signed certificates (instead of certificates signed by a recognized certificate authority). You can verify the master blocking sensor host sensor certificate by logging in to the host sensor and typing the **show tls fingerprint** command to see that the fingerprints of the host certificate match.

- Step 4** Enter **yes** to accept the certificate from the master blocking sensor.

- Step 5** Enter network access mode.

```
sensor(config)# service network-access
```

- Step 6** Enter general submenu.

```
sensor(config-net)# general
```

- Step 7** Add a master blocking sensor entry.

```
sensor(config-net-gen)# master-blocking-sensors master_blocking_sensor_ip_address
```

- Step 8** Specify the username for an administrative account on the master blocking sensor host.

```
sensor(config-net-gen-mas)# username username
```

- Step 9** Specify the password for the user.

```
sensor(config-net-gen-mas)# password
Enter password []: ****
Re-enter mbs-password []: ****
sensor(config-net-gen-mas)#
```

- Step 10** Specify the port number for the host HTTP communications. The default is 80/443 if not specified.

```
sensor(config-net-gen-mas)# port port_number
```



**Step 11** Specify whether or not the host uses TLS/SSL.

```
sensor(config-net-gen-mas)# tls {true | false}
sensor(config-net-gen-mas)
```



**Note** If you set the value to true, you need to use the command **tls trusted-host ip-address master\_blocking\_sensor\_ip\_address**.

**Step 12** Exit network access submode.

```
sensor(config-net-gen-mas)# exit
sensor(config-net-gen)# exit
sensor(config-net)# exit
sensor(config)# exit
Apply Changes:[yes]:
```

**Step 13** Press **Enter** to apply the changes or enter **no** to discard them.

**Step 14** On the master blocking sensor, add the block forwarding sensor IP address to the access list.

#### For More Information

For the procedure for adding the blocking forward sensor IP address to the access list, see [Changing the Access List, page 4-6](#).

## Configuring Host Blocking



#### Note

Connection blocks and network blocks are not supported on adaptive security appliances. Adaptive security appliances only support host blocks with additional connection information.

Use the **block host ip-address [timeout minutes]** command in privileged EXEC mode to block a host. Use the **no** form of the command to remove a block on a host. You must have blocking configured before you can set up host blocks. You can also view a list of hosts that are being blocked. If you do not configure the amount of time for the host block, it is permanent.

The following commands apply:

- *ip-address*—Specifies the IP address of the host to be blocked.
- *minutes*—(Optional) Specifies the duration of the host block in minutes. The valid range is 0 to 70560 minutes.

#### Blocking a Host

To block a host, follow these steps:

**Step 1** Log in to the CLI using an account with administrator or operator privileges.

**Step 2** Configure the host block for 15 minutes, for example. The host block ends in 15 minutes.

```
sensor# block host 192.0.2.1 timeout 15
```

**Step 3** Start a host block. The host block lasts until you remove it.

```
sensor# block host 192.0.2.1
```

**Step 4** End the host block.

```
sensor# no block host 192.0.2.1
sensor#
```

---

## Configuring Network Blocking



### Note

Connection blocks and network blocks are not supported on adaptive security appliances. Adaptive security appliances only support host blocks with additional connection information.

---

Use the **block network** *ip-address/netmask* [**timeout** *minutes*] command in privileged EXEC mode to block a network. Use the **no** form of the command to remove a block on a network. You must have blocking configured before you can set up network blocks. You can also view a list of networks that are being blocked. If you do not configure the amount of time for the network block, it is permanent.

The following commands apply:

- *ip-address/netmask*—Specifies the network subnet to be blocked in *X.X.X.X/nn* format, where *X.X.X.X* specifies the sensor IP address as a 32-bit address written as 4 octets separated by periods where *X* = 0-255, and *nn* specifies the number of bits (1032) in the netmask.
- *minutes*—(Optional) Specifies the duration of the network block in minutes. The valid range is 0 to 70560 minutes.

### Blocking a Network

To block a network, follow these steps:

---

**Step 1** Log in to the CLI using an account with administrator or operator privileges.

**Step 2** Configure the network block for 15 minutes, for example. The network block ends in 15 minutes.

```
sensor# block network 192.0.2.0/24 timeout 15
```

**Step 3** Start a network block. The network block lasts until you remove it.

```
sensor# block network 192.0.2.0/24
```

**Step 4** End the network block.

```
sensor# no block network 192.0.2.0/24
sensor#
```

---

# Configuring Connection Blocking

**Note**

Connection blocks and network blocks are not supported on adaptive security appliances. Adaptive security appliances only support host blocks with additional connection information.

Use the **block connection** *source-ip-address destination-ip-address* [**port** *port-number*] [**protocol** *type*] [**timeout** *minutes*] command in privileged EXEC mode to block a connection between two IP addresses. Use the **no** form of the command to remove the connection block. You must have blocking configured before you can set up connection blocks. You can also view a list of connections that are being blocked. If you do not configure the amount of time for the connection block, it is permanent.

The following commands apply:

- *source-ip-address*—Specifies the source IP address in a connection block.
- *destination-ip-address*—Specifies the destination IP address in a connection block.
- *port-number*—(Optional) Specifies the destination port number. The valid range is 0 to 65535.
- *type*—(Optional) Specifies the protocol type. The valid types are **tcp** or **udp**.
- *minutes*—(Optional) Specifies the duration of the connection block in minutes. The valid range is 0 to 70560 minutes.

## Blocking a Connection

To block a connection, follow these steps:

- 
- Step 1** Log in to the CLI using an account with administrator or operator privileges.
- Step 2** Configure the connection block between a source IP address and a destination IP address specifying the port, protocol, and time, for example. The connection block ends in 30 minutes.
- ```
sensor# block connection 10.0.0.0 172.16.0.0 port 80 protocol tcp timeout 30
```
- Step 3** Start a connection block. The connection block lasts until you remove it.
- ```
sensor# block connection 10.0.0.0 172.16.0.0
```
- Step 4** End the connection block.
- ```
sensor# no block connection 10.0.0.0
sensor#
```
-

Obtaining a List of Blocked Hosts and Connections

Use the **show statistics** command to obtain a list of blocked hosts and blocked connections.

To obtain a list of blocked hosts and connections, follow these steps:

-
- Step 1** Log in to the CLI.
 - Step 2** Check the statistics for the ARC. The `Host` entry indicates which hosts are being blocked and how long the blocks are.

```

sensor# show statistics network-access
Current Configuration
  LogAllBlockEventsAndSensors = true
  EnableNvramWrite = false
  EnableAclLogging = false
  AllowSensorBlock = false
  BlockMaxEntries = 250
  MaxDeviceInterfaces = 250
  NetDevice
    Type = Cisco
    IP = 10.1.1.1
    NATAddr = 0.0.0.0
    Communications = telnet
    BlockInterface
      InterfaceName = fa0/0
      InterfaceDirection = in
  State
    BlockEnable = true
    NetDevice
      IP = 10.1.1.1
      AclSupport = uses Named ACLs
      Version = 12.2
      State = Active
    BlockedAddr
      Host
        IP = 192.168.1.1
        Vlan =
        ActualIp =
        BlockMinutes = 80
        MinutesRemaining = 76

```



Configuring SNMP

This chapter describes how to configure SNMP, and contains the following sections:

- [SNMP Notes and Caveats, page 15-1](#)
- [Understanding SNMP, page 15-1](#)
- [Configuring SNMP, page 15-2](#)
- [Configuring SNMPv3 Users, page 15-4](#)
- [Displaying SNMPv3 Users and Engine IDs, page 15-7](#)
- [Configuring SNMP Traps, page 15-8](#)
- [Supported MIBS, page 15-11](#)

SNMP Notes and Caveats

The following notes and caveats apply to SNMP:

- To have the sensor send SNMP traps, you must also select **request-snmp-trap** as the event action when you configure signatures.
- MIB II is available on the sensor, but we do not support it. We know that some elements are not correct (for example, the packet counts from the IF MIB on the sensing interfaces). While you can use elements from MIB II, we do not guarantee that they all provide correct information. We fully support the other listed MIBs and their output is correct.
- CISCO-PROCESS-MIB is available on the sensor, but we do not support it. We know that some elements are not available. While you can use elements from CISCO-PROCESS-MIB, we do not guarantee that they all provide correct information. We fully support the other listed MIBs and their output is correct.
- We recommend that you configure SNMPv3 users with security levels that require authentication, such as authPriv and authNoPriv, with authPriv being the most highly recommended. Configuring SNMPv3 users with the noAuthNoPriv security level is NOT recommended.

Understanding SNMP

SNMP is an application layer protocol that facilitates the exchange of management information between network devices. SNMP enables network administrators to manage network performance, find and solve network problems, and plan for network growth.

SNMP is a simple request/response protocol. The network-management system issues a request, and managed devices return responses. This behavior is implemented by using one of four protocol operations: Get, GetNext, Set, and Trap.

You can configure the sensor for monitoring by SNMP. SNMP defines a standard way for network management stations to monitor the health and status of many types of devices, including switches, routers, and sensors.

You can configure the sensor to send SNMP traps. SNMP traps enable an agent to notify the management station of significant events by way of an unsolicited SNMP message. Trap-directed notification has the following advantage—if a manager is responsible for a large number of devices, and each device has a large number of objects, it is impractical to poll or request information from every object on every device. The solution is for each agent on the managed device to notify the manager without solicitation. It does this by sending a message known as a trap of the event.

After receiving the event, the manager displays it and can take an action based on the event. For instance, the manager can poll the agent directly, or poll other associated device agents to get a better understanding of the event.

**Note**

Trap-directed notification results in substantial savings of network and agent resources by eliminating frivolous SNMP requests. However, it is not possible to totally eliminate SNMP polling. SNMP requests are required for discovery and topology changes. In addition, a managed device agent cannot send a trap if the device has had a catastrophic outage.

You can use SNMPv2 and SNMPv3 protocol concurrently. If the SNMP request contains version 3 user information, then you get a version 3 reply (provided the same user is configured as a version 3 user in the IPS). If the SNMP request is a version 2 request, the IPS returns the response (provided the correct version 2 community string is configured).

**Note**

Encryption of the SNMPv3 payload uses AES-128 and authentication of the user password uses HMAC-SHA-96.

Configuring SNMP

**Caution**

To have the sensor send SNMP traps, you must also select **request-snmp-trap** as the event action when you configure signatures.

Configure general SNMP parameters in the service notification submenu.

The following commands apply:

- **default**—Sets the value back to the system default setting.
- **enable-set-get {true | false}**—Enables the **gets** and **sets** of object identifiers (OIDs).
- **no**—Removes an entry or selection setting.
- **read-only-community**—Specifies the read-only community name for the SNMP agent. The default is public.
- **read-write-community**—Specifies the read-write community name for the SNMP agent. The default is private.

- **snmp-agent-port**—Specifies the port the SNMP agent will listen on. The default SNMP port number is 161.
- **snmp-agent-protocol**—Specifies the protocol the SNMP agent will communicate with. The default protocol is UDP.
- **system-contact**—Specifies the contact information for this sensor. The system-contact option modifies the SNMPv2-MIB::sysContact.0 value.
- **system-location**—Specifies the location of the sensor. The system-location option modifies the SNMPv2-MIB::sysLocation.0 value.

Configuring SNMP General Parameters

To configure SNMP general parameters, follow these steps:

Step 1 Log in to the CLI using an account with administrator privileges.

Step 2 Enter notification submode.

```
sensor# configure terminal
sensor(config)# service notification
sensor(config-not)#
```

Step 3 Enable SNMP so that the SNMP management workstation can issue requests to the sensor SNMP agent.

```
sensor(config-not)# enable-set-get true
```

Step 4 Specify the SNMP agent parameters. These values configure the community name on the sensor SNMP agent. A community name is a plain-text password mechanism that is used to weakly authenticate SNMP queries.

- a. Assign the read-only community string. The read-only community name specifies the password for queries to the SNMP agent.

```
sensor(config-not)# read-only-community PUBLIC1
```

- b. Assign the read-write community string. The read-write community name specifies the password for sets to the SNMP agent.

```
sensor(config-not)# read-write-community PRIVATE1
```



Note The management workstation sends SNMP requests to the sensor SNMP agent, which resides on the sensor. If the management workstation issues a request and the community string does not match what is on the sensor, the sensor rejects it.

- c. Assign the sensor contact user ID.

```
sensor(config-not)# system-contact BUSINESS
```

- d. Enter the location of the sensor.

```
sensor(config-not)# system-location AUSTIN
```

- e. Enter the port of the sensor SNMP agent.

```
sensor(config-not)# snmp-agent-port 161
```



Note You must reboot the sensor if you change the port or protocol.

- f. Specify the protocol the sensor SNMP agent will use.

```
sensor(config-not)# snmp-agent-protocol udp
```



Note You must reboot the sensor if you change the port or protocol.

- Step 5** Verify the settings.

```
sensor(config-not)# show settings
trap-destinations (min: 0, max: 10, current: 0)
-----
-----
error-filter: error|fatal <defaulted>
enable-detail-traps: false <defaulted>
enable-notifications: false <defaulted>
enable-set-get: true default: false
snmp-agent-port: 161 default: 161
snmp-agent-protocol: udp default: udp
read-only-community: PUBLIC1 default: public
read-write-community: PRIVATE1 default: private
trap-community-name: public <defaulted>
system-location: AUSTIN default: Unknown
system-contact: BUSINESS default: Unknown
sensor(config-not)#
```

- Step 6** Exit notification submode.

```
sensor(config-not)# exit
Apply Changes?[yes]:
```

- Step 7** Press **Enter** to apply the changes or enter **no** to discard them.

For More Information

For the procedure for assigning actions to signatures, see [Assigning Actions to Signatures, page 7-20](#).

Configuring SNMPv3 Users

You configure SNMP v3users in the service notification submode.

The following parameters apply:

- **v3users [user-name]**—Lets you add SNMPv3 users. The maximum number of characters for the user name is 32.
- **access-control**—Sets the access privilege for the SNMPv3 user:
 - **rouser**—User with read-only access.
 - **rwuser**—User with read-write access.



Note Both rouser and rwuser can do 'get' operations, but rwuser access control is mandatory to do 'set' operations.

- **security-level** [**noAuthNoPriv** | **authNoPriv** | **authPriv**]—Sets the security level for the SNMPv3 user:
 - **noAuthNoPriv**—No Authentication and No Privacy, which means that no security is applied to messages.
 - **authNoPriv**—Authentication but No Privacy, which means that messages are authenticated.
 - **authPriv**—Authentication and Privacy, which means that messages are authenticated and encrypted.

**Note**

We recommend that you configure SNMPv3 users with security levels that require authentication, such as **authPriv** and **authNoPriv**, with **authPriv** being the most highly recommended. Configuring SNMPv3 users with the **noAuthNoPriv** security level is NOT recommended.

**Note**

For SNMPv3 users with **authPriv** and **authNoPriv** security levels, you must specify all security parameters or the changes are not applied and you receive a warning message.

- **auth-protocol** [**SHA**]—Sets the authentication protocol for the SNMPv3 user to SHA. If you do not set the authentication protocol to SHA, the default is none.
- **auth-passphrase** *passphrase*—Lets you enter an authentication passphrase for the SNMPv3 user. The valid range is 8 to 257 characters; no spaces or double quotes allowed.
- **priv-protocol** [**AES**]—Sets the encryption protocol to AES. If you do not set the privacy protocol, the default is none.
- **priv-passphrase** *passphrase*—Lets you enter an encryption passphrase for the SNMPv3 user. The valid range is 8 to 257 characters; no spaces or double quotes allowed.

**Caution**

The same passphrase value for both authentication and privacy is allowed, although it is not a recommended security practice.

- **default**—Sets the value back to the system default setting.
- **no**—Removes an entry or selection setting.

Configuring SNMPv3 Users

To configure SNMPv3 users, follow these steps:

Step 1 Log in to the CLI using an account with administrator privileges.

Step 2 Enter notification submode.

```
sensor# configure terminal
sensor(config)# service notification
sensor(config-not)#
```

Step 3 Add an SNMPv3 user.

```
sensor(config-not)# v3users user1
sensor(config-not-v3u)
```

Step 4 Set the access privilege for this user:

```
sensor(config-not-v3u)# access-control rwuser
sensor(config-not-v3u)
```

Step 5 Set the security level for this user.

```
sensor(config-not-v3u)# security-level authPriv
sensor(config-not-v3u)
```

Step 6 Set the authentication protocol to SHA.

```
sensor(config-not-v3u)# auth-protocol SHA
sensor(config-not-v3u)
```

Step 7 Enter an authentication passphrase for the SNMPv3 user and confirm it.

```
sensor(config-not-v3u)# auth-passphrase
Enter auth-passphrase[]: *****
Re-enter auth-passphrase: *****
sensor(config-not-v3u)#
```

Step 8 Set the privacy protocol to AES.

```
sensor(config-not-v3u)# priv-protocol AES
sensor(config-not-v3u)#
```

Step 9 Enter a privacy passphrase for the SNMPv3 user and confirm it.

```
sensor(config-not-v3u)# priv-passphrase
Enter priv-passphrase[]: *****
Re-enter priv-passphrase: *****
sensor(config-not-v3u)#
```

Step 10 Verify the settings.

```
sensor(config-not-v3u)# show settings
user-name: user1
-----
access-control: rwuser default: rouser
security-level: authPriv default: noAuthNoPriv
auth-protocol: SHA default:
priv-protocol: AES default:
auth-passphrase: <hidden>
priv-passphrase: <hidden>
-----
sensor(config-not-v3u)#
```

Step 11 Exit notification submenu.

```
sensor(config-not-v3u)# exit
sensor(config-not)# exit
Apply Changes:[yes]:
```

Step 12 Press **Enter** to apply the changes or enter **no** to discard them.

Displaying SNMPv3 Users and Engine IDs

You need to know which SNMPv3 users are configured on the sensor so you can add them to SNMP traps. Every SNMPv3 agent has an engine ID that serves as a unique identifier for the agent. The engine ID is used with a hashing function to generate keys for authentication and encryption of SNMP v3 messages. Use the **show snmpv3 users** and **show snmpv3 engineID** commands in EXEC mode to display users and engine IDs.

To display SNMPv3 users and engine IDs, follow these steps:

Step 1 Log in to the CLI.

Step 2 Display the SNMPv3 users configured on the sensor.

```
sensor# show snmpv3 users
Username: cisco1233333
Engine ID: 80001f88034c4e35ea727f
Access Control: rouser
Security Level: authNoPriv
Auth Protocol: SHA
-----
Username: cisco1234
Engine ID: 80001f88034c4e35ea727f
Access Control: rouser
Security Level: authNoPriv
Auth Protocol: SHA
-----
Username: cisco12344
Engine ID: 80001f88034c4e35ea727f
Access Control: rouser
Security Level: noAuthNoPriv
-----
Username: cisco1234444
Engine ID: 80001f88034c4e35ea727f
Access Control: rouser
Security Level: authPriv
Auth Protocol: SHA
Priv Protocol: AES
-----
Username: cisco123456
Engine ID: 80001f88034c4e35ea727f
Access Control: rwuser
Security Level: authPriv
Auth Protocol: SHA
Priv Protocol: AES
sensor#
```

Step 3 Display the SNMPv3 Engine ID.

```
sensor# show snmpv3 engineID
Local SNMP engineID: 80001f88034c4e35ea727f
sensor#
```

Step 4 Display the SNMPv3 users associated with a trap destination.

```
sensor# show snmpv3 users
Username: sat
Engine ID: 80001f8803503de59e5831
Access Control: rouser
Security Level: noAuthNoPriv
Associated with Trap Destination(s): 10.10.10.10:162
sensor#
```

Configuring SNMP Traps

You can configure SNMP traps in the service notification submode. An SNMP trap is a notification. You configure the sensor to send traps based on whether the event is fatal, an error, or a warning.

You can also associate SNMPv3 users with SNMP trap destinations. If no SNMPv3 user is associated with a trap, then an SNMPv2 trap is sent. For example, if a version 3 user is associated with a trap destination, all traps for that destination will be version 3 traps using the configured user. No version 2 trap is sent to that trap destination. If a version 3 user is not configured, then a version 2 trap is sent. Traps can be sent to one destination using version 3 and to another destination using version 2. Support for SNMPv3 is valid for IPS 7.2(2)E4 and later.

The following parameters apply:

- **enable-detail-traps {true | false}**—Enables the sending of detailed traps with no size limit. Otherwise traps are sent in sparse mode (less than 484 bytes).
- **enable-health-traps {true | false}** —Enables the sending of both heartbeat and health metric change traps.



Note To receive sensor health information through SNMP traps, you must have the sensor health metrics enabled. Use the **heartbeat-events enable** command in service health monitor submode to enable sensor health metrics.

- **enable-notifications {true | false}**—Enables event notifications.
- **error-filter {warning | error | fatal}**—Determines which errors generate an SNMP trap. An SNMP trap is generated for every evError event that matches the filter. The default is error and fatal.
- **trap-community-name**—Specifies the community name used when sending traps. If no community name is specified the general trap community name is used.
- **trap-destinations**—Defines the destinations to send error events and alert events generated from signature actions:
 - **trap-community-name**—Specifies the community name used when sending the trap. If no community name is specified the general trap community name is used.
 - **trap-port**—Specifies the port number to send the SNMP trap to.
 - **trap-v3user**—Specifies the SNMPv3 user of the trap destination. If no SNMPv3 user is specified, SNMPv2 is used.



Note You must save all configuration changes for an SNMPv3 user to be associated with the trap destination.



Note You cannot remove an SNMPv3 user associated with a trap destination unless you disassociate the user from the trap destination.

Configuring SNMP Traps

To configure SNMP traps and add SNMPv3 users to the traps, follow these steps:

Step 1 Log in to the CLI using an account with administrator privileges.

Step 2 Enter notification submode.

```
sensor# configure terminal
sensor(config)# service notification
sensor(config-not)#
```

Step 3 Enable SNMP traps.

```
sensor(config-not)# enable-notifications true
```

Step 4 Specify the parameters for the SNMP trap:

- a. Specify the error events you want to be notified about through SNMP traps.

```
sensor(config-not)# error-filter error|warning|fatal
```



Note The **error-filter [error | warning | fatal]** command includes error, warning, and fatal traps. It filters in (not filters out) the traps based on severity.

- b. Specify whether you want detailed SNMP traps.

```
sensor(config-not)# enable-detail-traps true
```

- c. Specify whether you want health traps (heartbeat and health metric change traps).

```
sensor(config-not)# enable-health-traps true
```



Note Make sure heartbeat and sensor health metrics are enabled.

- d. Enter the community string to be included in the detailed traps.

```
sensor(config-not)# trap-community-name TRAP1
```

Step 5 Specify the parameters for the SNMP trap destinations so the sensor knows which management workstations to send them to:

- a. Enter the IP address of the SNMP management station.

```
sensor(config-not)# trap-destinations 10.0.0.0
```

- b. Enter the UDP port of the SNMP management station. The default is 162

```
sensor(config-not-tra)# trap-port 162
```

- c. Enter the trap community string.

```
sensor(config-not-tra)# trap-community-name AUSTIN_PUBLIC1
```



Note The community string appears in the trap and is useful if you are receiving multiple types of traps from multiple agents. For example, a router or sensor could be sending the traps, and if you put something that identifies the router or sensor specifically in your community string, you can filter the traps based on the community string.

- d. Add an SNMPv3 user. Check first to see what SNMPv3 users have been configured on the sensor. If no SNMPv3 users have been configured on the sensor, SNMPv2 is used.

```
sensor(config-not-tra)# trap-v3user ?
cisco1233333    Configured SNMPv3 User
cisco1234      Configured SNMPv3 User
cisco12344     Configured SNMPv3 User
cisco1234444   Configured SNMPv3 User
cisco123456    Configured SNMPv3 User
sensor(config-not-tra)# trap-v3user cisco1234
sensor(config-not-tra)#
```

Step 6 Verify the settings.

```
sensor(config-not-tra)# exit
sensor(config-not)# show settings
trap-destinations (min: 0, max: 10, current: 1)
-----
ip-address: 10.0.0.0
-----
trap-community-name: AUSTIN_PUBLIC1 default:
trap-port: 162 default: 162
trap-v3user: cisco1234 default:
-----
error-filter: warning|error|fatal default: error|fatal
enable-detail-traps: true default: false
enable-health-traps: true default: false
enable-notifications: true default: false
enable-set-get: false <defaulted>
snmp-agent-port: 161 <defaulted>
snmp-agent-protocol: udp <defaulted>
read-only-community: default:
read-write-community: default:
trap-community-name: TRAP1 default: public
system-location: Unknown <defaulted>
system-contact: Unknown <defaulted>
v3users (min: 0, max: 25, current: 5)
-----
user-name: cisco1233333
-----
access-control: rouser <defaulted>
security-level: authNoPriv default: noAuthNoPriv
auth-protocol: SHA default:
priv-protocol: default:
auth-passphrase: <hidden>
priv-passphrase: <hidden>
-----
user-name: cisco1234
-----
access-control: rouser <defaulted>
security-level: authNoPriv default: noAuthNoPriv
```

```

auth-protocol: SHA default:
priv-protocol: default:
auth-passphrase: <hidden>
priv-passphrase: <hidden>
-----
user-name: cisco12344
-----
access-control: rouser <defaulted>
security-level: noAuthNoPriv <defaulted>
auth-protocol: default:
priv-protocol: default:
auth-passphrase: <hidden>
priv-passphrase: <hidden>
-----
user-name: cisco1234444
-----
access-control: rouser <defaulted>
security-level: authPriv default: noAuthNoPriv
auth-protocol: SHA default:
priv-protocol: AES default:
auth-passphrase: <hidden>
priv-passphrase: <hidden>
-----
user-name: cisco123456
-----
access-control: rwuser default: rouser
security-level: authPriv default: noAuthNoPriv
auth-protocol: SHA default:
priv-protocol: AES default:
auth-passphrase: <hidden>
priv-passphrase: <hidden>
-----
-----
sensor(config-not)#

```

Step 7 Exit notification submode.

```

sensor(config-not)# exit
Apply Changes:[yes]:

```

Step 8 Press **Enter** to apply the changes or enter **no** to discard them.

For More Information

For the procedure for assigning actions to signatures, see [Assigning Actions to Signatures, page 7-20](#).

Supported MIBS

The following private MIBs are supported on the sensor:

- CISCO-CIDS-MIB
 - The CISCO-CIDS-MIB has been updated to include SNMP health data.
- CISCO-ENHANCED-MEMPOOL-MIB
- CISCO-ENTITY-ALARM-MIB

You can obtain these private Cisco MIBs under the heading SNMP v2 MIBs at this URL:

<http://www.cisco.com/public/sw-center/netmgmt/cmtk/mibs.shtml>

**Note**

MIB II is available on the sensor, but we do not support it. We know that some elements are not correct (for example, the packet counts from the IF MIB on the sensing interfaces). While you can use elements from MIB II, we do not guarantee that they all provide correct information. We fully support the other listed MIBs and their output is correct.

**Note**

CISCO-PROCESS-MIB is available on the sensor, but we do not support it. We know that some elements are not available. While you can use elements from CISCO-PROCESS-MIB, we do not guarantee that they all provide correct information. We fully support the other listed MIBs and their output is correct.



Working With Configuration Files

This chapter describes how to use commands that show, copy, and erase the configuration file. It contains the following sections:

- [Displaying the Current Configuration, page 16-1](#)
- [Displaying the Current Submode Configuration, page 16-3](#)
- [Filtering the Current Configuration Output, page 16-16](#)
- [Filtering the Current Submode Configuration Output, page 16-18](#)
- [Displaying the Contents of a Logical File, page 16-19](#)
- [Backing Up and Restoring the Configuration File Using a Remote Server, page 16-21](#)
- [Creating and Using a Backup Configuration File, page 16-24](#)
- [Erasing the Configuration File, page 16-24](#)

Displaying the Current Configuration



Note

The CLI output is an example of what your configuration may look like. It will not match exactly due to the optional setup choices, sensor model, and IPS version you have installed.

Use the **show configuration** or the **more current-config** command to display the contents of the current configuration.

To display the contents of the current configuration, follow these steps:

Step 1 Log in to the CLI.

Step 2 Display the current configuration.

```
sensor# show configuration
! -----
! Current configuration last modified Fri Oct 10 09:41:17 2014
! -----
! Version 7.3(1)
! Host:
!   Realm Keys                key1.0
! Signature Definition:
!   Signature Update          S741.0   2013-09-10
!   Threat Profile Version    2
! -----
```

```
service interface
exit
! -----
service authentication
exit
! -----
service event-action-rules rules0
exit
! -----
service host
network-settings
host-ip 10.106.1.4/24,10.106.1.1
host-name sensor
telnet-option enabled
access-list 0.0.0.0/0
dns-primary-server disabled
exit
time-zone-settings
offset 0
standard-time-zone-name UTC
exit
exit
! -----
service logger
exit
! -----
service network-access
exit
! -----
service notification
exit
! -----
service signature-definition sig0
exit
! -----
service ssh-known-hosts
exit
! -----
service trusted-certificates
exit
! -----
service web-server
exit
! -----
service anomaly-detection ad0
exit
! -----
service external-product-interface
exit
! -----
service health-monitor
exit
! -----
service global-correlation
exit
! -----
```

```

service aaa
exit
! -----
service analysis-engine
exit

sensor#

```

Displaying the Current Submode Configuration

Use the **show settings** command in a submode to display the current configuration of that submode. To display the current configuration of a submode, follow these steps:

-
- Step 1** Log in to the CLI.
 - Step 2** Display the current configuration of the service analysis engine submode.

```

sensor# configure terminal
sensor(config)# service analysis-engine
sensor(config-ana)# show settings
  global-parameters
  -----
  ip-logging
  -----
  max-open-iplog-files: 20 <defaulted>
  -----
  virtual-sensor (min: 1, max: 255, current: 1)
  -----
  <protected entry>
  name: vs0 <defaulted>
  -----
  description: default virtual sensor <defaulted>
  signature-definition: sig0 <protected>
  event-action-rules: rules0 <protected>
  physical-interface (min: 0, max: 999999999, current: 0)
  -----
  logical-interface (min: 0, max: 999999999, current: 0)
  -----
  -----
sensor(config-ana)# exit
sensor(config)# exit
sensor#

```

- Step 3** Display the current configuration of the service anomaly detection submode.

```

sensor(config)# service anomaly-detection ad0
sensor(config-ano)# show settings
  worm-timeout: 600 seconds <defaulted>
  learning-accept-mode
  -----
  auto
  -----
  action: rotate <defaulted>
  schedule
  -----

```

```

periodic-schedule
-----
start-time: 10:00:00 <defaulted>
interval: 24 hours <defaulted>
-----
-----
internal-zone
-----
enabled: true <defaulted>
ip-address-range: 0.0.0.0 <defaulted>
tcp
-----
dst-port (min: 0, max: 65535, current: 0)
-----
-----
default-thresholds
-----
scanner-threshold: 100 <defaulted>
threshold-histogram (min: 0, max: 3, current: 3)
-----
<protected entry>
dest-ip-bin: low <defaulted>
num-source-ips: 10 <defaulted>
<protected entry>
dest-ip-bin: medium <defaulted>
num-source-ips: 1 <defaulted>
<protected entry>
dest-ip-bin: high <defaulted>
num-source-ips: 1 <defaulted>
-----
-----
enabled: true <defaulted>
-----
-----
udp
-----
dst-port (min: 0, max: 65535, current: 0)
-----
-----
default-thresholds
-----
scanner-threshold: 100 <defaulted>
threshold-histogram (min: 0, max: 3, current: 3)
-----
<protected entry>
dest-ip-bin: low <defaulted>
num-source-ips: 10 <defaulted>
<protected entry>
dest-ip-bin: medium <defaulted>
num-source-ips: 1 <defaulted>
<protected entry>
dest-ip-bin: high <defaulted>
num-source-ips: 1 <defaulted>
-----
-----
enabled: true <defaulted>
-----
-----
other
-----
protocol-number (min: 0, max: 255, current: 0)
-----
-----
default-thresholds

```

```

-----
scanner-threshold: 100 <defaulted>
threshold-histogram (min: 0, max: 3, current: 3)
-----
  <protected entry>
  dest-ip-bin: low <defaulted>
  num-source-ips: 10 <defaulted>
  <protected entry>
  dest-ip-bin: medium <defaulted>
  num-source-ips: 1 <defaulted>
  <protected entry>
  dest-ip-bin: high <defaulted>
  num-source-ips: 1 <defaulted>
-----

enabled: true <defaulted>
-----

illegal-zone
-----
enabled: true <defaulted>
ip-address-range: 0.0.0.0 <defaulted>
tcp
-----
dst-port (min: 0, max: 65535, current: 0)
-----

default-thresholds
-----
scanner-threshold: 100 <defaulted>
threshold-histogram (min: 0, max: 3, current: 3)
-----
  <protected entry>
  dest-ip-bin: low <defaulted>
  num-source-ips: 10 <defaulted>
  <protected entry>
  dest-ip-bin: medium <defaulted>
  num-source-ips: 1 <defaulted>
  <protected entry>
  dest-ip-bin: high <defaulted>
  num-source-ips: 1 <defaulted>
-----

enabled: true <defaulted>
-----

udp
-----
dst-port (min: 0, max: 65535, current: 0)
-----

default-thresholds
-----
scanner-threshold: 100 <defaulted>
threshold-histogram (min: 0, max: 3, current: 3)
-----
  <protected entry>
  dest-ip-bin: low <defaulted>
  num-source-ips: 10 <defaulted>
  <protected entry>
  dest-ip-bin: medium <defaulted>
  num-source-ips: 1 <defaulted>
  <protected entry>
  dest-ip-bin: high <defaulted>
  num-source-ips: 1 <defaulted>

```

```

-----
-----
enabled: true <defaulted>
-----
other
-----
protocol-number (min: 0, max: 255, current: 0)
-----
default-thresholds
-----
scanner-threshold: 100 <defaulted>
threshold-histogram (min: 0, max: 3, current: 3)
-----
<protected entry>
dest-ip-bin: low <defaulted>
num-source-ips: 10 <defaulted>
<protected entry>
dest-ip-bin: medium <defaulted>
num-source-ips: 1 <defaulted>
<protected entry>
dest-ip-bin: high <defaulted>
num-source-ips: 1 <defaulted>
-----
-----
enabled: true <defaulted>
-----
external-zone
-----
enabled: true <defaulted>
tcp
-----
dst-port (min: 0, max: 65535, current: 0)
-----
default-thresholds
-----
scanner-threshold: 100 <defaulted>
threshold-histogram (min: 0, max: 3, current: 3)
-----
<protected entry>
dest-ip-bin: low <defaulted>
num-source-ips: 10 <defaulted>
<protected entry>
dest-ip-bin: medium <defaulted>
num-source-ips: 1 <defaulted>
<protected entry>
dest-ip-bin: high <defaulted>
num-source-ips: 1 <defaulted>
-----
-----
enabled: true <defaulted>
-----
udp
-----
dst-port (min: 0, max: 65535, current: 0)
-----
default-thresholds
-----
scanner-threshold: 100 <defaulted>
threshold-histogram (min: 0, max: 3, current: 3)
-----

```

```

        <protected entry>
        dest-ip-bin: low <defaulted>
        num-source-ips: 10 <defaulted>
        <protected entry>
        dest-ip-bin: medium <defaulted>
        num-source-ips: 1 <defaulted>
        <protected entry>
        dest-ip-bin: high <defaulted>
        num-source-ips: 1 <defaulted>
    -----
    enabled: true <defaulted>
    -----
other
-----
protocol-number (min: 0, max: 255, current: 0)
-----
default-thresholds
-----
scanner-threshold: 100 <defaulted>
threshold-histogram (min: 0, max: 3, current: 3)
-----
    <protected entry>
    dest-ip-bin: low <defaulted>
    num-source-ips: 10 <defaulted>
    <protected entry>
    dest-ip-bin: medium <defaulted>
    num-source-ips: 1 <defaulted>
    <protected entry>
    dest-ip-bin: high <defaulted>
    num-source-ips: 1 <defaulted>
    -----
    enabled: true <defaulted>
    -----
ignore
-----
    enabled: true <defaulted>
    source-ip-address-range: 0.0.0.0 <defaulted>
    dest-ip-address-range: 0.0.0.0 <defaulted>
    -----
sensor(config-ano)# exit
sensor(config)# exit
sensor# exit

```

Step 4 Display the current configuration of the service authentication submode.

```

sensor# configure terminal
sensor(config)# service authentication
sensor(config-aut)# show settings
    attemptLimit: 0 <defaulted>
sensor(config-aut)# exit
sensor(config)# exit
sensor#

```

Step 5 Display the current configuration of the service event action rules submode.

```

sensor# configure terminal
sensor(config)# service event-action-rules rules0
sensor(config-rul)# show settings
    variables (min: 0, max: 256, current: 0)
    -----

```

```

-----
overrides (min: 0, max: 12, current: 0)
-----
-----
filters (min: 0, max: 4096, current: 0 - 0 active, 0 inactive)
-----
general
-----
  global-overrides-status: Enabled <defaulted>
  global-filters-status: Enabled <defaulted>
  global-summarization-status: Enabled <defaulted>
  global-metaevent-status: Enabled <defaulted>
  global-deny-timeout: 3600 <defaulted>
  global-block-timeout: 30 <defaulted>
  max-denied-attackers: 10000 <defaulted>
-----
target-value (min: 0, max: 5, current: 0)
-----
-----
sensor(config-rul)# exit
sensor(config)# exit
sensor# exit

```

Step 6 Display the current configuration of the external product interface submode.

```

sensor(config)# service external-product-interface
sensor(config-ext)# show settings
  cisco-security-agents-mc-settings (min: 0, max: 2, current: 0)
-----
-----
sensor(config-ext)# exit
sensor(config)# exit
sensor#

```

Step 7 Display the current configuration of the service global-correlation submode.

```

sensor# configure terminal
sensor(config)# service global-correlation
sensor(config-glo)# show settings
  network-participation: off <defaulted>
  global-correlation-inspection: on <defaulted>
  global-correlation-inspection-influence: standard <defaulted>
  reputation-filtering: on <defaulted>
  test-global-correlation: off <defaulted>
sensor(config-glo)# exit
sensor(config)# exit
sensor# exit

```

Step 8 Display the current configuration of the service health-monitor submode.

```

sensor# configure terminal
sensor(config)# service health-monitor
sensor(config-hea)# show settings
  enable-monitoring: true <defaulted>
  persist-security-status: 5 minutes <defaulted>
  heartbeat-events
-----
  enable: 300 seconds <defaulted>
-----
application-failure-policy
-----
  enable: true <defaulted>
  status: red <defaulted>
-----
-----
bypass-policy

```



```

-----
enable: true <defaulted>
status: red <defaulted>
-----
interface-down-policy
-----
enable: true <defaulted>
status: red <defaulted>
-----
inspection-load-policy
-----
enable: true <defaulted>
yellow-threshold: 80 percent <defaulted>
red-threshold: 91 percent <defaulted>
-----
missed-packet-policy
-----
enable: true <defaulted>
yellow-threshold: 1 percent <defaulted>
red-threshold: 6 percent <defaulted>
-----
memory-usage-policy
-----
enable: false <defaulted>
yellow-threshold: 80 percent <defaulted>
red-threshold: 91 percent <defaulted>
-----
signature-update-policy
-----
enable: true <defaulted>
yellow-threshold: 30 days <defaulted>
red-threshold: 60 days <defaulted>
-----
license-expiration-policy
-----
enable: true <defaulted>
yellow-threshold: 30 days <defaulted>
red-threshold: 0 days <defaulted>
-----
event-retrieval-policy
-----
enable: true <defaulted>
yellow-threshold: 300 seconds <defaulted>
red-threshold: 600 seconds <defaulted>
-----
global-correlation-policy
-----
enable: true <defaulted>
yellow-threshold: 86400 seconds <protected>
red-threshold: 259200 seconds <protected>
-----
network-participation-policy
-----
enable: false <defaulted>
yellow-threshold: 1 connection failures <protected>
red-threshold: 6 connection failures <protected>
-----
sensor(config-hea)# exit
sensor(config)# exit
sensor# exit

```

Step 9 Display the current configuration of the service host submode.

```
sensor# configure terminal
```

```

sensor(config)# service host
sensor(config-hos)# show settings
network-settings
-----
host-ip: 192.0.2.0/24,192.0.2.17 default: 192.168.1.2/24,192.168.1.1
host-name: sensor default: sensor
telnet-option: enabled default: disabled
access-list (min: 0, max: 512, current: 2)
-----
network-address: 10.0.0.0/8
-----
network-address: 64.0.0.0/8
-----

ftp-timeout: 300 seconds <defaulted>
login-banner-text: <defaulted>
-----
time-zone-settings
-----
offset: 0 minutes default: 0
standard-time-zone-name: UTC default: UTC
-----
ntp-option
-----
disabled
-----

summertime-option
-----
disabled
-----

auto-upgrade-option
-----
disabled
-----

crypto
-----
key (min: 0, max: 10, current: 2)
-----
<protected entry>
name: realm-cisco.pub <defaulted>
type
-----
rsa-pubkey
-----
length: 2048 <defaulted>
exponent: 65537 <defaulted>
modulus: 24442189989357747083874855335232628843599968934198559648
63019947387841151932503911172668940194754549155390407658020393330611891292508300
85940304031186014499632568812428068058089581614196337399623060624990057049103055
90153955935086060008679776808073640186063435723252375575293126304558068704301863
80562114437439289069456670922074995827390284761610591515752008405140243673083189
77822469964934598367010389389888297490802884118543730076293589703535912161993319
47093130298688830012547215572646349623539468838641064915313947806852904082351955
13217273138099965383039716130153270715220046567107828128924197692417332033911704
3 <defaulted>
-----
<protected entry>

```

```

name: realm-trend.pub <defaulted>
type
-----
  rsa-pubkey
-----
    length: 2048 <defaulted>
    exponent: 65537 <defaulted>
    modulus: 21765561422573021314159855351418723031625093380777053696
63817289527060570932551065489818190713745672148260527030060667208366606603802679
30439066724143390626495479300550101618179584637287052936465692146572612651375969
20354521585644221602944203520804404212975401970895119903756769601133853673296766
45289795777973491984056587045214514820063366950731346400044308491594626434706999
47608668822814014830063399534204647069509052443439525363706527255224510771122235
80181150460544783251498481432705991010069844368525754878413669427639752950801767
99905309235232456295580086724203297914095984224328444391582223138423799100838191
9 <defaulted>
-----
-----
-----
sensor(config-hos)# exit
sensor(config)# exit
sensor#

```

Step 10 Display the current configuration of the service interface submode.

```

sensor# configure terminal
sensor(config)# service interface
sensor(config-int)# show settings
  physical-interfaces (min: 0, max: 999999999, current: 4)
-----
  <protected entry>
  name: GigabitEthernet0/0 <defaulted>
-----
    media-type: tx <protected>
    description: <defaulted>
    admin-state: disabled <defaulted>
    duplex: auto <defaulted>
    speed: auto <defaulted>
    alt-tcp-reset-interface
-----
    none
-----
-----
  subinterface-type
-----
    none
-----
-----
-----
  <protected entry>
  name: GigabitEthernet0/1 <defaulted>
-----
    media-type: tx <protected>
    description: <defaulted>
    admin-state: disabled <protected>
    duplex: auto <defaulted>
    speed: auto <defaulted>
    alt-tcp-reset-interface
-----
    none
-----

```

```

-----
subinterface-type
-----
none
-----
-----
-----
<protected entry>
name: GigabitEthernet2/0 <defaulted>
-----
media-type: xl <protected>
description: <defaulted>
admin-state: disabled <defaulted>
duplex: auto <defaulted>
speed: auto <defaulted>
alt-tcp-reset-interface
-----
none
-----
-----
subinterface-type
-----
none
-----
-----
-----
<protected entry>
name: GigabitEthernet2/1 <defaulted>
-----
media-type: xl <protected>
description: <defaulted>
admin-state: disabled <defaulted>
duplex: auto <defaulted>
speed: auto <defaulted>
alt-tcp-reset-interface
-----
none
-----
-----
subinterface-type
-----
none
-----
-----
-----
command-control: GigabitEthernet0/1 <protected>
inline-interfaces (min: 0, max: 999999999, current: 0)
-----
-----
bypass-mode: auto <defaulted>
interface-notifications
-----
missed-percentage-threshold: 0 percent <defaulted>
notification-interval: 30 seconds <defaulted>
idle-interface-delay: 30 seconds <defaulted>
-----
sensor(config-int)# exit

```

```
sensor(config)# exit
sensor#
```

Step 11 Display the current configuration for the service logger submode.

```
sensor# configure terminal
sensor(config)# service logger
sensor(config-log)# show settings
  master-control
  -----
  enable-debug: false <defaulted>
  individual-zone-control: false <defaulted>
  -----
  zone-control (min: 0, max: 999999999, current: 14)
  -----
  <protected entry>
  zone-name: Cid
  severity: debug <defaulted>
  <protected entry>
  zone-name: AuthenticationApp
  severity: warning <defaulted>
  <protected entry>
  zone-name: Cli
  severity: warning <defaulted>
  <protected entry>
  zone-name: csi
  severity: warning <defaulted>
  <protected entry>
  zone-name: ctlTransSource
  severity: warning <defaulted>
  <protected entry>
  zone-name: IdapiCtlTrans
  severity: warning <defaulted>
  <protected entry>
  zone-name: IdsEventStore
  severity: warning <defaulted>
  <protected entry>
  zone-name: MpInstaller
  severity: warning <defaulted>
  <protected entry>
  zone-name: nac
  severity: warning <defaulted>
  <protected entry>
  zone-name: sensorApp
  severity: warning <defaulted>
  <protected entry>
  zone-name: tls
  severity: warning <defaulted>
  <protected entry>
  zone-name: intfc
  severity: warning <defaulted>
  <protected entry>
  zone-name: cmgr
  severity: warning <defaulted>
  <protected entry>
  zone-name: cplane
  severity: warning <defaulted>
  -----
sensor(config-log)# exit
sensor(config)# exit
sensor#
```

Step 12 Display the current configuration for the service network access submode.

```

sensor# configure terminal
sensor(config)# service network-access
sensor(config-net)# show settings
  general
  -----
  log-all-block-events-and-errors: true <defaulted>
  enable-nvram-write: false <defaulted>
  enable-acl-logging: false <defaulted>
  allow-sensor-block: false <defaulted>
  block-enable: true <defaulted>
  block-max-entries: 250 <defaulted>
  max-interfaces: 250 <defaulted>
  rate-limit-max-entries: 250 <defaulted>
  master-blocking-sensors (min: 0, max: 100, current: 0)
  -----
  never-block-hosts (min: 0, max: 250, current: 0)
  -----
  never-block-networks (min: 0, max: 250, current: 0)
  -----
  block-hosts (min: 0, max: 250, current: 0)
  -----
  block-networks (min: 0, max: 250, current: 0)
  -----
  -----
  user-profiles (min: 0, max: 250, current: 1)
  -----
  profile-name: test
  -----
  enable-password: <hidden>
  password: <hidden>
  username: <defaulted>
  -----
  -----
  cat6k-devices (min: 0, max: 250, current: 0)
  -----
  -----
  router-devices (min: 0, max: 250, current: 0)
  -----
  -----
  firewall-devices (min: 0, max: 250, current: 0)
  -----
  -----
sensor(config-net)# exit
sensor(config)# exit
sensor#

```

Step 13 Display the current configuration for the notification submode.

```

sensor# configure terminal
sensor(config)# service notification
sensor(config-not)# show settings
  trap-destinations (min: 0, max: 10, current: 0)
  -----
  -----
  error-filter: error|fatal <defaulted>
  enable-detail-traps: false <defaulted>
  enable-notifications: false <defaulted>

```

```

enable-set-get: false <defaulted>
snmp-agent-port: 161 <defaulted>
snmp-agent-protocol: udp <defaulted>
read-only-community: public <defaulted>
read-write-community: private <defaulted>
trap-community-name: public <defaulted>
system-location: Unknown <defaulted>
system-contact: Unknown <defaulted>
sensor(config-not)# exit
sensor(config)# exit
sensor#

```

Step 14 Display the current configuration for the signature definition submode.

```

sensor# configure terminal
sensor(config)# service signature-definition sig0
sensor(config-sig)# show settings
  variables (min: 0, max: 256, current: 1)
  -----
  <protected entry>
  variable-name: WEBPORTS
  -----
    web-ports: 80-80,3128-3128,8000-8000,8010-8010,8080-8080,8888-8888,2432
6-24326 <defaulted>
  -----
  application-policy
  -----
    http-policy
  -----
    http-enable: false <defaulted>
    max-outstanding-http-requests-per-connection: 10 <defaulted>
    aic-web-ports: 80-80,3128-3128,8000-8000,8010-8010,8080-8080,8888-8888,
24326-24326 <defaulted>
  -----
    ftp-enable: false <defaulted>
  -----
  fragment-reassembly
  -----
    ip-reassemble-mode: nt <defaulted>
  -----
  stream-reassembly
  -----
--MORE--

```

Step 15 Display the current configuration for the SSH known hosts submode.

```

sensor# configure terminal
sensor(config)# service ssh-known-hosts
sensor(config-ssh)# show settings
  rsal-keys (min: 0, max: 500, current: 0)
  -----
sensor(config-ssh)# exit
sensor(config)# exit
sensor#

```

Step 16 Display the current configuration for the trusted certificates submode.

```

sensor# configure terminal
sensor(config)# service trusted-certificate
sensor(config-tru)# show settings
  trusted-certificates (min: 0, max: 500, current: 1)
  -----
  common-name: 10.89.130.108

```

```

certificate: MIICJDCCAY0CCPbSkGXUchJIMA0GCSqGSIB3DQEBBQUAMFcxzAJBgNVBAYTA
1VTMRwwGgYDVQQKEzNDaXNjbyBTeXN0ZW1zLCBjbmMuMRlweAYDVQQLEw1TU00tSVBtMjAx
FjAUBGNVBAMTDTFwLjg5LjEzMC4xMDgwHhcNMDMwMTAzMDE1MjEwWhcNMDUwMTAzMDE1MjEwWjB
XMQswCQYDVQQUQEw0xMC44OS4xMzAuMTA4MIGfMA0GCSqGSIB3DQEBAQUAA4GNADCBiQKBgQ
CzldqLFG4MT4bfgh3mJfP/DCilnnaLfzHK9FdnhmWI4FY+9MVvAI7MOhAcuV6HYfyp6n6cYvH+E
swz19uv7H5nouID9St9GI3YrSUt1IQAJ4QVL2DwWP230x6KdHrYqcj+Nmhc7AnnPypjidwGSfF+V
etIJLEeRPh/mI2JcmwF2QIDAQABMA0GCSqGSIB3DQEBBQUAA4GBAAUI2PLANTOehxvcfwd6
UAFXvy8uifbjqKMC1jrrF+f9KGkxmR+XZvUaGOS83FYDXlXJvB5Xyxms+Y01wGjzKKpxegBoan8
OB8o193Ueszdpvz2xYmiEgywCDyVJRsW3hAFMXWMS5XsBUIHtw0btHH0j7E1FZxUjZv12
fGz8hlnY
-----
sensor(config-tru)# exit
sensor(config)# exit
sensor#

```

Step 17 Display the current configuration for the web server submode.

```

sensor# configure terminal
sensor(config)# service web-server
sensor(config-web)# show settings
  enable-tls: true <defaulted>
  port: 443 <defaulted>
  server-id: HTTP/1.1 compliant <defaulted>
sensor(config-web)# exit
sensor(config)# exit
sensor#

```

Filtering the Current Configuration Output

Use the **more** *keyword* | [**begin** | **exclude** | **include**] *regular-expression* command to search the output of the more command.

The following commands apply:

- *keyword*—Specifies either the current-config or the backup-config:
 - **current-config**—Specifies the current running configuration. This configuration becomes persistent as the commands are entered.
 - **backup-config**—Specifies the storage location for the configuration backup file.
- |—The pipe symbol indicates that an output processing specification follows.
- **begin**—Begins unfiltered output of the **more** command with the first line that contains the regular expression specified.
- **exclude**—Excludes lines in the output of the **more** command that contain a particular regular expression.
- **include**—Includes only the lines in the output of the **more** command that contain the regular expression you specify.
- *regular-expression*—Specifies any regular expression found in the **more** command output.



Note The *regular-expression* option is case sensitive and allows for complex matching requirements.

Filtering Using the More Command

To filter the **more** command, follow these steps:

- Step 1** Log in to the CLI using an account with administrator privileges.
- Step 2** Filter the current-config output beginning with the regular expression “ip,” for example.

```
sensor# more current-config | begin ip
generating current config:
host-ip 192.0.2.0/24,192.0.2.17
host-name sensor
telnet-option enabled
access-list 10.0.0.0/8
access-list 64.0.0.0/8
exit
time-zone-settings
offset 0
standard-time-zone-name UTC
exit
exit
! -----
service interface
exit
! -----
service logger
master-control
enable-debug true
exit
exit
! -----
service network-access
general
log-all-block-events-and-errors true
--MORE--
```



Note Press **Ctrl-C** to stop the output and return to the CLI prompt.

- Step 3** Exclude the regular expression “ip” from the current-config output.

```
sensor# more current-config | exclude ip
generating current config:
! -----
! Version 7.0(1)
! Current configuration last modified Fri Feb 11 15:10:57 2009
! -----
service analysis-engine
virtual-sensor vs0
physical-interface FastEthernet0/1
exit
exit
! -----
service authentication
exit
! -----
service event-action-rules rules0
exit
! -----
service host
network-settings
host-name sensor
telnet-option enabled
```

```
access-list 10.0.0.0/8
access-list 64.0.0.0/8
exit
time-zone-settings
--MORE--
```



Note Press **Ctrl-C** to stop the output and return to the CLI prompt.

Step 4 Include the regular expression “ip” in the current-config output.

```
sensor# more current-config | include ip
generating current config:
host-ip 192.0.2.0/24,192.0.2.17
engine atomic-ip
```

Filtering the Current Submode Configuration Output

Use the `show settings | [begin | exclude | include] regular_expression` command in the submode you are interested in to search or filter the output of the contents of the submode configuration.

The following commands apply:

- |—The pipe symbol indicates that an output processing specification follows.
- **begin**—Begins unfiltered output of the `show settings` command with the first line that contains the regular expression specified.
- **exclude**—Excludes lines in the output of the `show settings` command that contain a particular regular expression.
- **include**—Includes only the lines in the output of the `show settings` command that contain the regular expression you specify.
- *regular_expression*—Specifies any regular expression found in the `show settings` command output.



Note The *regular_expression* option is case sensitive and allows for complex matching requirements.

Filtering the Submode Output

To search or filter the output of the contents of the submode configuration, follow these steps:

Step 1 Log in to the CLI using an account with administrator privileges.

Step 2 Search the output of the event action rules settings for the regular expression, “filters,” for example.

```
sensor# configure terminal
sensor(config)# service event-action-rules
sensor(config-rul)# show settings | begin filters
filters (min: 0, max: 4096, current: 0 - 0 active, 0 inactive)
-----
general
-----
global-overrides-status: Enabled <defaulted>
global-filters-status: Enabled <defaulted>
```

```

global-summarization-status: Enabled <defaulted>
global-metaevent-status: Enabled <defaulted>
global-deny-timeout: 3600 <defaulted>
global-block-timeout: 15 default: 30
max-denied-attackers: 10000 <defaulted>
-----
target-value (min: 0, max: 5, current: 0)
-----
-----
sensor(config-rul)#

```

Step 3 Filter the output of the network access settings to exclude the regular expression.

```

sensor# configure terminal
sensor(config)# service network-access
sensor(config-net)# show settings | exclude false
  general
  -----
  log-all-block-events-and-errors: true default: true
  block-enable: true default: true
  block-max-entries: 11 default: 250
  max-interfaces: 13 default: 250
  master-blocking-sensors (min: 0, max: 100, current: 1)
  -----
  ipaddress: 192.0.2.0
  -----
  password: <hidden>
  port: 443 default: 443
  tls: true default: true
  username: cisco default:
  -----
  never-block-hosts (min: 0, max: 250, current: 1)
  -----
  ip-address: 10.89.146.112
  -----
  never-block-networks (min: 0, max: 250, current: 1)
  -----
  ip-address: 88.88.88.0/24
--MORE--

```

Step 4 Filter the output of the host settings to include the regular expression “ip.”

```

sensor# configure terminal
sensor(config)# service host
sensor(config-hos)# show settings | include ip
  host-ip: 192.0.2.0/24,192.0.2.17 default: 192.168.1.2/24,192.168.1.1
sensor(config-hos)#

```

Displaying the Contents of a Logical File



Note

Operators and viewers can only display the current configuration. Only administrators can view hidden fields such as passwords.

Use the **more** *keyword* command to display the contents of a logical file, such as the current system configuration or the saved backup system configuration.

The following commands apply:

- *keyword*—Specifies either the current-config or the backup-config:
 - **current-config**—Specifies the current running configuration. This configuration becomes persistent as the commands are entered.
 - **backup-config**—Specifies the storage location for the configuration backup file.

You can disable the more prompt in **more current-config** or **more backup-config** by setting the terminal length to zero using the **terminal length 0** command. The **more** command then displays the entire file content without pausing.

Displaying the Logical File Contents

To display the contents of a logical file, follow these steps:

Step 1 Log in to the CLI using an account with administrator privileges.

Step 2 Display the contents of the current configuration file.

```
sensor# more current-config
Generating current config:
```

The current configuration is displayed.

```
! -----
! Current configuration last modified Fri Oct 10 09:41:17 2014
! -----
! Version 7.3(1)
! Host:
!   Realm Keys                key1.0
! Signature Definition:
!   Signature Update          S741.0   2013-09-10
!   Threat Profile Version    2
! -----
service interface
exit
! -----
service authentication
exit
! -----
service event-action-rules rules0
exit
! -----
service host
network-settings
host-ip 10.106.1.4/24,10.106.1.1
host-name sensor
telnet-option enabled
access-list 0.0.0.0/0
dns-primary-server disabled
exit
time-zone-settings
offset 0
standard-time-zone-name UTC
exit
exit
! -----
service logger
exit
```

```
! -----
service network-access
exit
! -----
service notification
exit
! -----
service signature-definition sig0
exit
! -----
service ssh-known-hosts
exit
! -----
service trusted-certificates
exit
! -----
service web-server
exit
! -----
service anomaly-detection ad0
exit
! -----
service external-product-interface
exit
! -----
service health-monitor
exit
! -----
service global-correlation
exit
! -----
service aaa
exit
! -----
service analysis-engine
exit

sensor#
```

For More Information

For the procedure for using the **terminal** command, see [Modifying Terminal Properties](#), page 17-21.

Backing Up and Restoring the Configuration File Using a Remote Server



Note

We recommend copying the current configuration file to a remote server before upgrading.

Use the **copy [/erase] source_url destination_url keyword** command to copy the configuration file to a remote server. You can then restore the current configuration from the remote server. You are prompted to back up the current configuration first.

The following commands apply:

- **/erase**—Erases the destination file before copying.
This keyword only applies to the current-config; the backup-config is always overwritten. If this keyword is specified for destination current-config, the source configuration is applied to the system default configuration. If it is not specified for the destination current-config, the source configuration is merged with the current-config.
- *source_url*—The location of the source file to be copied. It can be a URL or keyword.
- *destination_url*—The location of the destination file to be copied. It can be a URL or a keyword.
- **current-config**—The current running configuration. The configuration becomes persistent as the commands are entered.
- **backup-config**—The storage location for the configuration backup.

The exact format of the source and destination URLs varies according to the file. Here are the valid types:

- **ftp:**—Source or destination URL for an FTP network server. The syntax for this prefix is:

```
ftp://[[username@]location][relativeDirectory]/filename
```

```
ftp://[[username@]location][absoluteDirectory]/filename
```



Note You are prompted for a password.

- **scp:**—Source or destination URL for the SCP network server. The syntax for this prefix is:

```
scp://[[username@]location][relativeDirectory]/filename
```

```
scp://[[username@]location][absoluteDirectory]/filename
```



Note You are prompted for a password. You must add the remote host to the SSH known hosts list.

- **http:**—Source URL for the web server. The syntax for this prefix is:

```
http://[[username@]location][directory]/filename
```



Note The directory specification should be an absolute path to the desired file.

- **https:**—Source URL for the web server. The syntax for this prefix is:

```
https://[[username@]location][directory]/filename
```



Note The directory specification should be an absolute path to the desired file. The remote host must be a TLS trusted host.



Caution

Copying a configuration file from another sensor may result in errors if the sensing interfaces and virtual sensors are not configured the same.

Backing Up the Current Configuration to a Remote Server

To back up your current configuration to a remote server, follow these steps:

Step 1 Log in to the CLI using an account with administrator privileges.

Step 2 Back up the current configuration to the remote server.

```
sensor# copy current-config scp://user@192.0.2.0//configuration/cfg current-config
Password: *****
Warning: Copying over the current configuration may leave the box in an unstable state.
Would you like to copy current-config to backup-config before proceeding? [yes]:
```

Step 3 Enter **yes** to copy the current configuration to a backup configuration.

```
cfg          100% |*****| 36124          00:00
```

Restoring the Current Configuration From a Backup File

To restore your current configuration from a backup file, follow these steps:

Step 1 Log in to the CLI using an account with administrator privileges.

Step 2 Back up the current configuration to the remote server.

```
sensor# copy scp://user@192.0.2.0//configuration/cfg current-config
Password: *****
Warning: Copying over the current configuration may leave the box in an unstable state.
Would you like to copy current-config to backup-config before proceeding? [yes]:
```

Step 3 Enter **yes** to copy the current configuration to a backup configuration.

```
cfg          100% |*****| 36124          00:00
```

```
Warning: Replacing existing network-settings may leave the box in an unstable state.
Would you like to replace existing network settings
(host-ipaddress/netmask/gateway/access-list) on sensor before proceeding? [no]:
sensor#
```

Step 4 Enter **no** to retain the currently configured hostname, IP address, subnet mask, management interface, and access list. We recommend you retain this information to preserve access to your sensor after the rest of the configuration has been restored.

For More Information

- For the procedure for adding the remote host to the SSH known host list, see [Adding Hosts to the SSH Known Hosts List, page 4-47](#).
- For the procedure for adding the remote host to the TLS trusted hosts list, see [Adding TLS Trusted Hosts, page 4-53](#).

Creating and Using a Backup Configuration File

To protect your configuration, you can back up the current configuration and then display it to confirm that is the configuration you want to save. If you need to restore this configuration, you can merge the backup configuration file with the current configuration or overwrite the current configuration file with the backup configuration file.

To back up your current configuration, follow these steps:

-
- Step 1** Log in to the CLI using an account with administrator privileges.
- Step 2** Save the current configuration. The current configuration is saved in a backup file.
- ```
sensor# copy current-config backup-config
```
- Step 3** Display the backup configuration file. The backup configuration file is displayed.
- ```
sensor# more backup-config
```
- Step 4** You can either merge the backup configuration with the current configuration, or you can overwrite the current configuration:
- Merge the backup configuration into the current configuration.


```
sensor# copy backup-config current-config
```
 - Overwrite the current configuration with the backup configuration.


```
sensor# copy /erase backup-config current-config
```
-

Erasing the Configuration File

Use the `erase {backup-config | current-config}` command to delete a logical file. The following commands apply:

- **current-config**—The current running configuration. The configuration becomes persistent as the commands are entered.
- **backup-config**—The storage location for the configuration backup.

To erase the current configuration and return all settings back to the default, follow these steps:

-
- Step 1** Log in to the CLI using an account with administrator privileges.
- ```
sensor# erase current-config
Warning: Removing the current-config file will result in all configuration being reset to
default, including system information such as IP address.
User accounts will not be erased. They must be removed manually using the "no username"
command.
Continue? []:
```
- Step 2** Press **Enter** to continue or enter **no** to stop.
-





## Administrative Tasks for the Sensor

---

This chapter contains procedures that will help you with the administrative aspects of your sensor. It contains the following sections:

- [Administrative Notes and Caveats, page 17-2](#)
- [Recovering the Password, page 17-2](#)
- [Clearing the Sensor Databases, page 17-10](#)
- [Displaying and Deleting SDEE Server Subscriptions, page 17-11](#)
- [Displaying the Inspection Load of the Sensor, page 17-12](#)
- [Configuring Health Status Information, page 17-14](#)
- [Showing Sensor Overall Health Status, page 17-19](#)
- [Creating a Banner Login, page 17-20](#)
- [Terminating CLI Sessions, page 17-20](#)
- [Modifying Terminal Properties, page 17-21](#)
- [Configuring Events, page 17-22](#)
- [Configuring the System Clock, page 17-25](#)
- [Displaying and Clearing the Denied Attackers List, page 17-27](#)
- [Displaying Policy Lists, page 17-29](#)
- [Displaying Statistics, page 17-30](#)
- [Displaying Tech Support Information, page 17-41](#)
- [Displaying Version Information, page 17-43](#)
- [Diagnosing Network Connectivity, page 17-45](#)
- [Resetting the Appliance, page 17-46](#)
- [Displaying Command History, page 17-47](#)
- [Displaying Hardware Inventory, page 17-47](#)
- [Tracing the Route of an IP Packet, page 17-48](#)
- [Displaying Submode Settings, page 17-49](#)

## Administrative Notes and Caveats

The following notes and caveats apply to administrative tasks for the sensor:

- Administrators may need to disable the password recovery feature for security reasons.
- If you try to recover the password on a sensor on which password recovery is disabled, the process proceeds with no errors or warnings; however, the password is not reset. If you cannot log in to the sensor because you have forgotten the password, and password recovery is set to disabled, you must reimage your sensor.
- We do not recommend that you use **clear database** command unless under the direction of TAC or in some testing conditions when you need to clear accumulated state information and start with a clean database.
- The ASA 5500-X IPS SSP and ASA 5585-X IPS SSP do not support bypass mode. The adaptive security appliance will either fail open, fail close, or fail over depending on the configuration of the adaptive security appliance and the type of activity being done on the IPS.
- When the sensor is first starting, it is normal for certain health metric statuses to be red until the sensor is fully up and running.
- You do not need to set the system clock if your sensor is synchronized by a valid outside timing mechanism such as an NTP clock source.
- The **show inventory** command does not apply to the ASA IPS modules (ASA 5500-X IPS SSP or ASA 5585-X IPS SSP).

## Recovering the Password

For most IPS platforms, you can now recover the password on the sensor rather than using the service account or reimaging the sensor. This section describes how to recover the password for the various IPS platforms. It contains the following topics:

- [Understanding Password Recovery, page 17-2](#)
- [Recovering the Password for the Appliance, page 17-3](#)
- [Recovering the Password for the ASA 5500-X IPS SSP, page 17-4](#)
- [Recovering the Password for the ASA 5585-X IPS SSP, page 17-6](#)
- [Disabling Password Recovery, page 17-8](#)
- [Verifying the State of Password Recovery, page 17-9](#)
- [Troubleshooting Password Recovery, page 17-9](#)

## Understanding Password Recovery



### Note

---

Administrators may need to disable the password recovery feature for security reasons.

---

Password recovery implementations vary according to IPS platform requirements. Password recovery is implemented only for the cisco administrative account and is enabled by default. The IPS administrator can then recover user passwords for other accounts using the CLI. The cisco user password reverts to **cisco** and must be changed after the next login.

Table 17-1 lists the password recovery methods according to platform.

**Table 17-1 Password Recovery Methods According to Platform**

| Platform                                   | Description                                             | Recovery Method                         |
|--------------------------------------------|---------------------------------------------------------|-----------------------------------------|
| 4300 series sensors<br>4500 series sensors | Standalone IPS appliances                               | GRUB prompt or ROMMON                   |
| ASA 5500-X IPS SSP<br>ASA 5585-X IPS SSP   | ASA 5500 series adaptive security appliance IPS modules | Adaptive security appliance CLI command |

## Recovering the Password for the Appliance

This section describes the two ways to recover the password for appliances. It contains the following topics:

- [Using the GRUB Menu, page 17-3](#)
- [Using ROMMON, page 17-4](#)

### Using the GRUB Menu



#### Note

You must have a terminal server or direct serial connection to the appliance to use the GRUB menu to recover the password.

For the IPS 4355, IPS 4360, IPS 4510, IPS 4520, and IPS 4520-XL appliances, the password recovery is found in the GRUB menu, which appears during bootup. When the GRUB menu appears, press any key to pause the boot process.

To recover the password on appliances, follow these steps:

**Step 1** Reboot the appliance to see the GRUB menu.

```
GNU GRUB version 0.94 (632K lower / 523264K upper memory)

0: Cisco IPS
1: Cisco IPS Recovery
2: Cisco IPS Clear Password (cisco)

```

```
Use the ^ and v keys to select which entry is highlighted.
Press enter to boot the selected OS, 'e' to edit the
Commands before booting, or 'c' for a command-line.
```

```
Highlighted entry is 0:
```

**Step 2** Press any key to pause the boot process.

**Step 3** Choose **2: Cisco IPS Clear Password (cisco)**. The password is reset to **cisco**. Log in to the CLI with username **cisco** and password **cisco**. You can then change the password.

## Using ROMMON

For the IPS 4345, IPS 4360, IPS 4510, IPS 4520, and IPS 4520-XL, you can use the ROMMON to recover the password. To access the ROMMON CLI, reboot the sensor from a terminal server or direct connection and interrupt the boot process.



### Note

After recovering the password, you must reset the confreg to **0**, otherwise, when you try to upgrade the sensor, the upgrade fails because when the sensor reboots, it goes to password recovery (**confreg 0x7**) rather than to the upgrade option.

To recover the password using the ROMMON CLI, follow these steps:

- 
- Step 1** Reboot the appliance.
- Step 2** To interrupt the boot process, press **ESC** or **Control-R** (terminal server) or send a **BREAK** command (direct connection). The boot code either pauses for 10 seconds or displays something similar to one of the following:
- Evaluating boot options
  - Use BREAK or ESC to interrupt boot
- Step 3** Enter the following commands to reset the password:

```
confreg 0x7
boot
```

Sample ROMMON session:

```
Booting system, please wait...
CISCO SYSTEMS
Embedded BIOS Version 1.0(11)2 01/25/06 13:21:26.17
...
Evaluating BIOS Options...
Launch BIOS Extension to setup ROMMON
Cisco Systems ROMMON Version (1.0(11)2) #0: Thu Jan 26 10:43:08 PST 2006
Platform IPS-4360-K9
Use BREAK or ESC to interrupt boot.
Use SPACE to begin boot immediately.
Boot interrupted.
Management0/0
Link is UP
MAC Address:000b.fcfa.d155
Use ? for help.
rommon #0> confreg 0x7
Update Config Register (0x7) in NVRAM...
rommon #1> boot
```

- Step 4** Enter the following command to reset the confreg value to 0:

```
confreg 0
```

---

## Recovering the Password for the ASA 5500-X IPS SSP

You can reset the password to the default (**cisco**) for the ASA 5500-X IPS SSP using the CLI or the ASDM. Resetting the password causes it to reboot. IPS services are not available during a reboot.

**Note**

To reset the password, you must have ASA 8.6.1 or later.

Use the **sw-module module ips password-reset** command to reset the password to the default **cisco**. If the module in the specified slot has an IPS version that does not support password recovery, the following error message is displayed:

```
ERROR: the module in slot <n> does not support password recovery.
```

To reset the password on the ASA 5500-X IPS SSP, follow these steps:

**Step 1** Log into the adaptive security appliance and enter the following command:

```
asa# sw-module module ips password-reset
Reset the password on module ips? [confirm]
```

**Step 2** Press **Enter** to confirm.

```
Password-Reset issued for module ips.
```

**Step 3** Verify the status of the module. Once the status reads **Up**, you can session to the ASA 5500-X IPS SSP.

```
asa# show module ips
Mod Card Type Model Serial No.

ips ASA 5555-X IPS Security Services Processor ASA5555-IPS FCH151070GR

Mod MAC Address Range Hw Version Fw Version Sw Version

ips 503d.e59c.7c4c to 503d.e59c.7c4c N/A N/A 7.1(4)E4

Mod SSM Application Name Status SSM Application Version

ips IPS Up 7.1(4)E4

Mod Status Data Plane Status Compatibility

ips Up Up

Mod License Name License Status Time Remaining

ips IPS Module Enabled 210 days
```

**Step 4** Session to the ASA 5500-X IPS SSP.

```
asa# session ips
Opening command session with module ips.
Connected to module ips. Escape character sequence is 'CTRL-^X'.
```

**Step 5** Enter the default username (**cisco**) and password (**cisco**) at the login prompt.

```
login: cisco
Password: cisco
```

```
You are required to change your password immediately (password aged)
Changing password for cisco.
(current) password: cisco
```

**Step 6** Enter your new password twice.

```
New password: new password
Retype new password: new password
```

\*\*\*NOTICE\*\*\*

This product contains cryptographic features and is subject to United States and local country laws governing import, export, transfer and use. Delivery of Cisco cryptographic products does not imply third-party authority to import, export, distribute or use encryption. Importers, exporters, distributors and users are responsible for compliance with U.S. and local country laws. By using this product you agree to comply with applicable laws and regulations. If you are unable to comply with U.S. and local laws, return this product immediately.

A summary of U.S. laws governing Cisco cryptographic products may be found at:  
<http://www.cisco.com/wwl/export/crypto/tool/stqrg.html>

If you require further assistance please contact us by sending email to [export@cisco.com](mailto:export@cisco.com).

\*\*\*LICENSE NOTICE\*\*\*

There is no license key installed on this IPS platform. The system will continue to operate with the currently installed signature set. A valid license must be obtained in order to apply signature updates. Please go to <http://www.cisco.com/go/license> to obtain a new license or install a license.

asa-ssp#

---

### Using the ASDM

To reset the password in the ASDM, follow these steps:

- 
- Step 1** From the ASDM menu bar, choose **Tools > IPS Password Reset**.




---

**Note** This option does not appear in the menu if there is no IPS present.

---

- Step 2** In the IPS Password Reset confirmation dialog box, click **OK** to reset the password to the default (**cisco**). A dialog box displays the success or failure of the password reset. If the reset fails, make sure you have the correct ASA and IPS software versions.

- Step 3** Click **Close** to close the dialog box. The sensor reboots.
- 

## Recovering the Password for the ASA 5585-X IPS SSP




---

**Note** To reset the password, you must have ASA 8.2.(4.4) or later or ASA 8.4.2 or later. The ASA 5585-X IPS SSP is not supported in ASA 8.3(x).

---

You can reset the password to the default (**cisco**) for the ASA 5585-X IPS SSP using the CLI or the ASDM. Resetting the password causes it to reboot. IPS services are not available during a reboot.

Use the **hw-module module slot\_number password-reset** command to reset the password to the default **cisco**. If the module in the specified slot has an IPS version that does not support password recovery, the following error message is displayed:

```
ERROR: the module in slot <n> does not support password recovery.
```

To reset the password on the ASA 5585-X IPS SSP, follow these steps:

**Step 1** Log into the adaptive security appliance and enter the following command:

```
asa# hw-module module 1 password-reset
Reset the password on module in slot 1? [confirm]
```

**Step 2** Press **Enter** to confirm.

```
Password-Reset issued for slot 1.
```

**Step 3** Verify the status of the module. Once the status reads `Up`, you can session to the ASA 5585-X IPS SSP.

```
asa# show module 1
Mod Card Type Model Serial No.

 1 ASA 5585-X IPS Security Services Processor-4 ASA5585-SSP-IPS40 JAF1436ABSG

Mod MAC Address Range Hw Version Fw Version Sw Version

 1 5475.d029.8c74 to 5475.d029.8c7f 0.1 2.0(12)3 7.1(4)E4

Mod SSM Application Name Status SSM Application Version

 1 IPS Up 7.1(4)E4

Mod Status Data Plane Status Compatibility

 1 Up Up
```

**Step 4** Session to the ASA 5585-X IPS SSP.

```
asa# session 1
Opening command session with slot 1.
Connected to slot 1. Escape character sequence is 'CTRL-^X'.
```

**Step 5** Enter the default username (**cisco**) and password (**cisco**) at the login prompt.

```
login: cisco
Password: cisco
```

```
You are required to change your password immediately (password aged)
Changing password for cisco.
(current) password: cisco
```

**Step 6** Enter your new password twice.

```
New password: new password
Retype new password: new password
```

\*\*\*NOTICE\*\*\*

This product contains cryptographic features and is subject to United States and local country laws governing import, export, transfer and use. Delivery of Cisco cryptographic products does not imply third-party authority to import, export, distribute or use encryption. Importers, exporters, distributors and users are responsible for compliance with U.S. and local country laws. By using this product you agree to comply with applicable laws and regulations. If you are unable to comply with U.S. and local laws, return this product immediately.

A summary of U.S. laws governing Cisco cryptographic products may be found at:  
<http://www.cisco.com/wwl/export/crypto/tool/stqrg.html>

If you require further assistance please contact us by sending email to [export@cisco.com](mailto:export@cisco.com).

\*\*\*LICENSE NOTICE\*\*\*

There is no license key installed on this IPS platform. The system will continue to operate with the currently installed signature set. A valid license must be obtained in order to apply signature updates. Please go to <http://www.cisco.com/go/license> to obtain a new license or install a license.

ips\_ssp#

### Using the ASDM

To reset the password in the ASDM, follow these steps:

- Step 1** From the ASDM menu bar, choose **Tools > IPS Password Reset**.



**Note** This option does not appear in the menu if there is no IPS present.

- Step 2** In the IPS Password Reset confirmation dialog box, click **OK** to reset the password to the default (**cisco**). A dialog box displays the success or failure of the password reset. If the reset fails, make sure you have the correct ASA and IPS software versions.
- Step 3** Click **Close** to close the dialog box. The sensor reboots.

## Disabling Password Recovery



### Caution

If you try to recover the password on a sensor on which password recovery is disabled, the process proceeds with no errors or warnings; however, the password is not reset. If you cannot log in to the sensor because you have forgotten the password, and password recovery is set to disabled, you must reimage your sensor.

Password recovery is enabled by default. You can disable password recovery through the CLI, or .

### Disabling Password Recovery Using the CLI

To disable password recovery in the CLI, follow these steps:

- Step 1** Log in to the CLI using an account with administrator privileges.
- Step 2** Enter global configuration mode.
- ```
sensor# configure terminal
```
- Step 3** Enter host mode.
- ```
sensor(config)# service host
```
- Step 4** Disable password recovery.
- ```
sensor(config-hos)# password-recovery disallowed
```


Disabling Password Recovery Using the

To disable password recovery in the , follow these steps:

-
- Step 1** Log in to the using an account with administrator privileges.
 - Step 2** Choose **Configuration > Sensor Setup > Network**.
 - Step 3** To disable password recovery, uncheck the **Allow Password Recovery** check box.
-

Verifying the State of Password Recovery

Use the **show settings | include password** command to verify whether password recovery is enabled.

To verify whether password recovery is enabled, follow these steps:

-
- Step 1** Log in to the CLI.
 - Step 2** Enter service host submode.

```
sensor# configure terminal
sensor (config)# service host
sensor (config-hos)#
```
 - Step 3** Verify the state of password recovery by using the **include** keyword to show settings in a filtered output.

```
sensor(config-hos)# show settings | include password
  password-recovery: allowed <defaulted>
sensor(config-hos)#
```
-

Troubleshooting Password Recovery

When you troubleshoot password recovery, pay attention to the following:

- You cannot determine whether password recovery has been disabled in the sensor configuration from the ROMMON prompt, GRUB menu, switch CLI, or router CLI. If you attempt password recovery, it always appears to succeed. If it has been disabled, the password is not reset to **cisco**. The only option is to reimage the sensor.
- You can disable password recovery in the host configuration. For the platforms that use external mechanisms and ROMMON, although you can run commands to clear the password, if password recovery is disabled in the IPS, the IPS detects that password recovery is not allowed and rejects the external request.
- To check the state of password recovery, use the **show settings | include password** command.

Clearing the Sensor Databases



Caution

We do not recommend that you use **clear database** command unless under the direction of the TAC or in some testing conditions when you need to clear accumulated state information and start with a clean database.

Use the **clear database** [*virtual-sensor*] **all** | **nodes** | **alerts** | **inspectors** command in privileged EXEC mode to clear specific parts of the sensor database. The **clear database** command is useful for troubleshooting and testing.

The following commands apply:

- *virtual-sensor*—Specifies the name of a virtual sensor configured on the sensor.
- **all**— Clears all nodes, inspectors, and alerts databases.



Caution

This command causes summary alerts to be discarded.

- **nodes**—Clears the overall packet database elements, including the packet nodes, TCP session information, and inspector lists.
- **alerts**—Clears the alert database including the alerts nodes, Meta inspector information, summary state, and event count structures.
- **inspectors**—Clears the inspector lists in the nodes. Inspector lists represent the packet work and observations collected during the time the sensor is running.

Clearing the Sensor Database

To clear the sensor database, follow these steps:

Step 1 Log in to the CLI using an account with administrator privileges.

Step 2 Clear the entire sensor database.

```
sensor# clear database all
Warning: Executing this command will delete database on all virtual sensors
Continue? [yes]:
```

Step 3 Enter **yes** to clear all the databases on the sensor.

Step 4 Clear the packet nodes.

```
sensor# clear database nodes
Warning: Executing this command will delete database on all virtual sensors
Continue? [yes]:
```

Step 5 Enter **yes** to clear the packet nodes database.

Step 6 Clear the alerts database on a specific virtual sensor.

```
sensor# clear database vs0 alerts
Warning: Executing this command will delete database on all virtual sensors
Continue? [yes]:
```

Step 7 Enter **yes** to clear the alerts database.

Step 8 Clear inspector lists on the sensor.

```
sensor# clear database inspectors
```

```
Warning: Executing this command will delete database on all virtual sensors
Continue? [yes]:
```

Step 9 Enter **yes** to clear the inspectors database.

Displaying and Deleting SDEE Server Subscriptions



Note

Support for SDEE subscription is valid for IPS 7.2(2)E4 and later, and IPS 73(2)E4 and later.

You can display the details of the SDEE client subscriptions on the sensor. You can view the SDEE subscription ID, the status (expired or valid) of the subscription, the IP address of each SDEE client for each listed subscription, and see the last time the subscription was read. The SDEE server automatically deletes SDEE subscriptions that appear to be idle or left open for 24 hours, although the timer checks for expired subscriptions every 12 hours.

To display and delete SDEE server subscriptions, follow these steps:

Step 1 Log in to the CLI.

Step 2 Display the SDEE server subscriptions.

```
sensor# show statistics sdee-server
General
  Open Subscriptions = 3
  Blocked Subscriptions = 1
  Maximum Available Subscriptions = 5
  Maximum Events Per Retrieval = 500
Subscriptions
  sub-13-13979b85
    State = Read Pending
    IP Address = 10.142.52.151
    Last Read Time = 17:12:54 UTC Tue Jan 28 2014
    Last Read Time (nanoseconds) = 1390929174555130000
    Status = Valid
  sub-14-798ebed7
    State = Open
    IP Address = 10.142.52.127
    Last Read Time = 17:12:53 UTC Tue Jan 28 2014
    Last Read Time (nanoseconds) = 1390929173318336000
    Status = Valid
  sub-15-fb8e1c5e
    State = Open
    IP Address = 10.89.11.67
    Last Read Time = 17:12:51 UTC Tue Jan 28 2014
    Last Read Time (nanoseconds) = 1390929171920713000
    Status = Valid
sensor#
```

Step 3 Use the subscription ID to clear a specific SDEE subscription.

```
sensor# clear sdee-subscription sub-13-13979b85
Warning: Going to delete sdee subscription id sub-13-13979b85

Continue? []:
```

Step 4 Enter **yes** to delete the subscription.

Step 5 Verify that the subscription was deleted. Subscription ID sub-13-13979b85 no longer appears in the list.

```

sensor# show statistics sdee-server
General
  Open Subscriptions = 3
  Blocked Subscriptions = 2
  Maximum Available Subscriptions = 5
  Maximum Events Per Retrieval = 500
Subscriptions
sub-14-798ebed7
  State = Read Pending
  IP Address = 10.142.52.127
  Last Read Time = 12:01:14 UTC Wed Jan 29 2014
  Last Read Time (nanoseconds) = 1390996874294610000
  Status = Valid
sub-15-fb8e1c5e
  State = Open
  IP Address = 10.89.11.67
  Last Read Time = 12:01:11 UTC Wed Jan 29 2014
  Last Read Time (nanoseconds) = 1390996871880999000
  Status = Valid
sub-16-d4220496
  State = Read Pending
  IP Address = 10.142.52.151
  Last Read Time = 12:01:14 UTC Wed Jan 29 2014
  Last Read Time (nanoseconds) = 1390996874311401000
  Status = Valid
sensor#

```

For More Information

For detailed information about the SDEE Server, see [SDEE, page A-33](#).

Displaying the Inspection Load of the Sensor

Use the **show inspection-load** command in privileged EXEC mode to display a timestamp and the current inspection load of the sensor. Use the **history** option to display a histogram of the inspection load over the past 60 minutes and over the past 72 hours.

Use this command to determine the load on the sensor instead of the CPU Usage information from the **show statistics host** command. The inspection load is a more accurate representation of the processing level of the sensor. The calculation of the inspection load has also been enhanced to provide a more accurate calculation of the sensor load at lower traffic levels.



Note

The Processing Load category in the **show statistics virtual-sensor** output has been renamed to Inspection Load and shows the same value seen in the **show inspection load** command.



Note

The **show inspection-load** command is not currently supported for the IPS 4500 series sensors.

To display the inspection load of the sensor, follow these steps:

Step 1 Log in to the CLI.

Step 2 Show the current inspection load with a timestamp of the sensor.

```
sensor# show inspection-load

sensor 08:18:13 PM Friday Jan 15 2011 UTC

Inspection Load Percentage = 1
```

Step 3 Show the histogram of the inspection load.

```
sensor# show inspection-load history
sensor 15:36:42 UTC Mon Jan 30 2012

sensor 08:18:13 PM Friday Jan 15 2011 UTC

Inspection Load Percentage = 65

100
90
80
70 * *
60 * * * * * * * * * * * * * * * * *
50 * * * * * * * * * * * * * * * * *
40 * * * * * * * * * * * * * * * * *
30 * * * * * * * * * * * * * * * * *
20 * * * * * * * * * * * * * * * * *
10 * * * * * * * * * * * * * * * * *
0.....1.....2.....3.....4.....5.....6

Inspection Load Percentage (last 6 minutes at 10 second intervals)
```

```
100
90
80
70
60 * * * * * * * * * * * * * * * * *
50 * * * * * * * * * * * * * * * * *
40 * * * * * * * * * * * * * * * * *
```

```

30 #####*#####*#####*#####*#####*#####*#####*#####*#####*#####*#####*#####*#####*#####
20 #####*#####*#####*#####*#####*#####*#####*#####*#####*#####*#####*#####*#####*#####
10 #####*#####*#####*#####*#####*#####*#####*#####*#####*#####*#####*#####*#####*#####
0...5...1...1...2...2...3...3...4...4...5...5...6
       0    5    0    5    0    5    0    5    0    5    0    5    0

```

Inspection Load Percentage (last 60 minutes) *=maximum #=average

```

100
 90
 80
 70
 60 * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
 50 * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
 40 * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
 30 #####*#####*#####*#####*#####*#####*#####*#####*#####*#####*#####*#####*#####*#####
 20 #####*#####*#####*#####*#####*#####*#####*#####*#####*#####*#####*#####*#####*#####
 10 #####*#####*#####*#####*#####*#####*#####*#####*#####*#####*#####*#####*#####*#####
0...5...1...1...2...2...3...3...4...4...5...5...6...6...7.
       0    5    0    5    0    5    0    5    0    5    0    5    0

```

Inspection Load Percentage (last 72 hours) *=maximum #=average sensor#

Configuring Health Status Information

Configure the health statistics for the sensor in service health monitor submode. Use the **show health** command to see the results. The health status categories are rated by red and green with red being critical.

The following commands apply:

- **application-failure-policy {enable | disable} {true | false} status {green | yellow | red}**—Lets you choose to have an application failure applied to the overall sensor health rating.
- **bypass-policy {enable | disable} {true | false} status {green | yellow | red}**—Lets you choose to know if bypass mode is active and have that apply to the overall sensor health rating.



Note The ASA 5500-X IPS SSP and ASA 5585-X IPS SSP do not support bypass mode. The adaptive security appliance will either fail open, fail close, or fail over depending on the configuration of the adaptive security appliance and the type of activity being done on the IPS.

- **enable-monitoring {true | false}**—Lets you choose to monitor sensor health and security.
- **event-retrieval-policy {enable | disable} {true | false} red-threshold yellow-threshold seconds**—Lets you set a threshold for when the last event was retrieved and have that apply to the overall sensor health rating. The health status is degraded to red or yellow when that threshold is met. The range for the threshold is 0 to 4294967295 seconds.



Note The event retrieval metric keeps track of when the last event was retrieved by an external monitoring application such as the IME. Disable **event retrieval policy** if you are not doing external event monitoring.

- **global-correlation-policy {enable | disable} {true | false}**—Lets you apply this metric to the overall sensor health rating.
- **heartbeat-events {enable | disable} seconds**—Lets you enable heartbeat events to be emitted at the specified interval in seconds and have that apply to the overall sensor health rating. The range for the interval is 15 to 86400 seconds.
- **inspection-load-policy {enable | disable} {true | false} red-threshold yellow-threshold seconds**—Lets you set the threshold for inspection load. The health status is degraded to red or yellow when that threshold is met. The range is 0 to 100.
- **interface-down-policy {enable | disable} {true | false} status {green | yellow | red}**—Lets you choose to know if one or more enabled interfaces are down and have that apply to the overall sensor health rating.
- **license-expiration-policy {enable | disable} {true | false} red-threshold yellow-threshold**—Lets you set a threshold for when the license expires and whether this metric is applied to the overall sensor health rating. The range for the threshold is 0 to 4294967295 seconds.
- **memory-usage-policy {enable | disable} {true | false} red-threshold yellow-threshold**—Lets you set a threshold percentage for memory usage and whether this metric is applied to the overall sensor health rating. The range is 0 to 100. The default for red is 91% and the default for yellow is 80%.
- **missed-packet-policy {enable | disable} {true | false} red-threshold yellow-threshold**—Lets you set a threshold percentage for missed packets and whether this metric is applied to the overall sensor health rating.
- **network-participation-policy {enable | disable} {true | false}**—Lets you apply this metric to the overall sensor health rating.
- **persist-security-status**—Lets you set the number of minutes that a lower security persists following the occurrence of the latest event to lower the security status.
- **signature-update-policy {enable | disable} {true | false} red-threshold yellow-threshold**—Lets you set a threshold for the number of days elapsed since the last signature update and whether this metric is applied to the overall sensor health rating. The range for the threshold is 0 to 4294967295 seconds

ASA 5500-X IPS SSP and Memory Usage

For the ASA 5500-X IPS SSP, the memory usage is 93%. The default health thresholds for the sensor are 80% for yellow and 91% for red, so the sensor health will be shown as red on these platforms even for normal operating conditions. You can tune the threshold percentage for memory usage so that it reads more accurately for these platforms by configuring the **memory-usage-policy** option in the sensor health metrics.

**Note**

Make sure you have the **memory-usage-policy** option in the sensor health metrics enabled.

Table 17-2 lists the yellow-threshold and the red-threshold health values.

Table 17-2 ASA 5500-X IPS SSP Memory Usage Values

Platform	Yellow	Red	Memory Used
ASA 5512-X IPS SSP	85%	91%	28%
ASA 5515-X IPS SSP	88%	92%	14%
ASA 5525-X IPS SSP	88%	92%	14%
ASA 5545-X IPS SSP	93%	96%	13%
ASA 5555-X IPS SSP	95%	98%	17%

Configuring Health Statistics

To configure the health statistics for the sensor, follow these steps:

Step 1 Log in to the CLI using an account with administrator privileges.

Step 2 Enter service health monitor submode.

```
sensor# configure terminal
sensor(config)# service health-monitor
sensor(config-hea)#
```

Step 3 Enable the metrics for application failure status.

```
sensor(config-hea)# application-failure-policy
sensor(config-hea-app)# enable true
sensor(config-hea-app)# status red
sensor(config-hea-app)# exit
sensor(config-hea)#
```

Step 4 Enable the metrics for bypass policy.

```
sensor(config-hea)# bypass-policy
sensor(config-hea-byp)# enable true
sensor(config-hea-byp)# status yellow
sensor(config-hea-byp)# exit
sensor(config-hea)#
```

Step 5 Enable the metrics for sensor health and security monitoring.

```
sensor(config-hea)# enable-monitoring true
sensor(config-hea)#
```

Step 6 Set the event retrieval thresholds for event retrieval metrics.


```
sensor(config-hea)# event-retrieval-policy
sensor(config-hea-eve)# enable true
sensor(config-hea-eve)# red-threshold 100000
sensor(config-hea-eve)# yellow-threshold 100
sensor(config-hea-eve)# exit
sensor(config-hea)#
```

Step 7 Enable health metrics for global correlation.

```
sensor(config-hea)# global-correlation-policy
sensor(config-hea-glo)# enable true
sensor(config-hea-glo)# exit
sensor(config-hea)#
```

Step 8 Enable the metrics for heartbeat events to be emitted at the specified interval of seconds.

```
sensor(config-hea)# heartbeat-events enable 20000
sensor(config-hea)#
```

Step 9 Set the inspection load threshold.

```
sensor(config-hea)# inspection-load-policy
sensor(config-hea-ins)# enable true
sensor(config-hea-ins)# red-threshold 100
sensor(config-hea-ins)# yellow-threshold 50
sensor(config-hea-ins)# exit
sensor(config-hea)#
```

Step 10 Enable the interface down policy.

```
sensor(config-hea)# interface-down-policy
sensor(config-hea-int)# enable true
sensor(config-hea-int)# status yellow
sensor(config-hea-int)# exit
sensor(config-hea)#
```

Step 11 Set the number of days until the license expires.

```
sensor(config-hea)# license-expiration-policy
sensor(config-hea-lic)# enable true
sensor(config-hea-lic)# red-threshold 400000
sensor(config-hea-lic)# yellow-threshold 200000
sensor(config-hea-lic)# exit
sensor(config-hea)#
```

Step 12 Set the threshold for memory usage.

```
sensor(config-hea)# memory-usage-policy
sensor(config-hea-mem)# enable true
sensor(config-hea-mem)# red-threshold 100
sensor(config-hea-mem)# yellow-threshold 50
sensor(config-hea-mem)# exit
sensor(config-hea)#
```

Step 13 Set the missed packet threshold.

```
sensor(config-hea)# missed-packet-policy
sensor(config-hea-mis)# enable true
sensor(config-hea-mis)# red-threshold 50
sensor(config-hea-mis)# yellow-threshold 20
sensor(config-hea-mis)# exit
sensor(config-hea)#
```

Step 14 Set the number of minutes that a lower security persists following the occurrence of the latest event to lower the security status.

```
sensor(config-hea)# persist-security-status 10
sensor(config-hea)#
```

Step 15 Set the number of days since the last signature update.

```
sensor(config-hea)# signature-update-policy
sensor(config-hea-sig)# enable true
sensor(config-hea-sig)# red-threshold 30000
sensor(config-hea-sig)# yellow-threshold 10000
sensor(config-hea-sig)# exit
sensor(config-hea)#
```

Step 16 Verify your settings.

```
sensor(config-hea)# show settings
enable-monitoring: true default: true
persist-security-status: 10 minutes default: 5
heartbeat-events
-----
enable: 20000 seconds default: 300
-----
application-failure-policy
-----
enable: true default: true
status: red default: red
-----
bypass-policy
-----
enable: true default: true
status: yellow default: red
-----
interface-down-policy
-----
enable: true default: true
status: yellow default: red
-----
inspection-load-policy
-----
enable: true default: true
yellow-threshold: 50 percent default: 80
red-threshold: 100 percent default: 91
-----
missed-packet-policy
-----
enable: true default: true
yellow-threshold: 20 percent default: 1
red-threshold: 50 percent default: 6
-----
memory-usage-policy
-----
enable: true default: false
yellow-threshold: 50 percent default: 80
red-threshold: 100 percent default: 91
-----
signature-update-policy
-----
enable: true default: true
yellow-threshold: 10000 days default: 30
red-threshold: 30000 days default: 60
-----
license-expiration-policy
-----
enable: true default: true
yellow-threshold: 200000 days default: 30
```

```

red-threshold: 400000 days default: 0
-----
event-retrieval-policy
-----
enable: true <defaulted>
yellow-threshold: 100000 seconds default: 300
red-threshold: 100 seconds default: 600
-----
sensor(config-hea)#

```

Step 17 Exit health monitoring submode.

```

sensor(config-hea)# exit
Apply Changes:[yes]:

```

Step 18 Press **Enter** to apply the changes or enter **no** to discard them.

Showing Sensor Overall Health Status



Caution

When the sensor is first starting, it is normal for certain health metric statuses to be red until the sensor is fully up and running.



Note

The ASA 5500-X IPS SSP and ASA 5585-X IPS SSP do not support bypass mode. The adaptive security appliance will either fail open, fail close, or fail over depending on the configuration of the adaptive security appliance and the type of activity being done on the IPS.

Use the **show health** command in privileged EXEC mode to display the overall health status information of the sensor. The health status categories are rated by red and green with red being critical.

To display the overall health status of the sensor, follow these steps:

Step 1 Log in to the CLI.

Step 2 Show the health and security status of the sensor.

```

sensor# show health
Overall Health Status                               Red
Health Status for Failed Applications                Green
Health Status for Signature Updates                  Green
Health Status for License Key Expiration             Red
Health Status for Running in Bypass Mode             Green
Health Status for Interfaces Being Down              Red
Health Status for the Inspection Load                Green
Health Status for the Time Since Last Event Retrieval Green
Health Status for the Number of Missed Packets       Green
Health Status for the Memory Usage                   Not Enabled
Health Status for Global Correlation                 Red
Health Status for Network Participation               Not Enabled

Security Status for Virtual Sensor vs0               Green
sensor#

```

Creating a Banner Login

Use the **banner login** command to create a banner login that will be displayed before the user and password login prompts. The maximum message length is 2500 characters. Use the **no banner login** command to remove the banner.

To create a banner login, follow these steps:

Step 1 Log in to the CLI using an account with administrator privileges.

Step 2 Enter global configuration mode.

```
sensor# configure terminal
```

Step 3 Create the banner login.

```
sensor(config)# banner login
Banner[ ]:
```

Step 4 Enter your message.

```
Banner[ ]: This message will be displayed on banner login. ^M Thank you
sensor(config)#
```



Note To use a ? or a carriage return in the message, press **Ctrl-V-?** or **Ctrl-V-Enter**. They are represented by ^M.

Example

```
This message will be displayed on login.
Thank you
login: cisco
Password:****
```

Step 5 Remove the banner login. The banner no longer appears at login.

```
sensor(config)# no banner login
```

Terminating CLI Sessions



Caution

You can only clear CLI login sessions with the **clear line** command. You cannot clear service logins with this command.

Use the **clear line cli_id [message]** command to terminate another CLI session. If you use the **message** keyword, you can send a message along with the termination request to the receiving user. The maximum message length is 2500 characters.

The following commands apply:

- **cli_id**—Specifies the CLI ID number associated with the login session. Use the **show users** command to find the CLI ID number.
- **message**—Specifies the message to send to the receiving user.

If an administrator tries to log in when the maximum sessions have been reached, the following message appears:

```
Error: The maximum allowed CLI sessions are currently open, would you like to terminate
one of the open sessions? [no]
```

If an operator or viewer tries to log in when the maximum sessions are open, the following message appears:

```
Error: The maximum allowed CLI sessions are currently open, please try again later.
```

To terminate a CLI session, follow these steps:

Step 1 Log in to the CLI using an account with administrator privileges.



Note Operator and viewer can only clear lines with the same username as the current login.

Step 2 Find the CLI ID number associated with the login session.

```
sensor# show users
      CLI ID  User      Privilege
*   13533    jtaylor  administrator
      15689    jsmith   operator
      20098    viewer   viewer
```

Step 3 Terminate the CLI session of jsmith.

```
sensor# clear line cli_id message
Message[]:
```

Example

```
sensor# clear line 15689 message
Message{}: Sorry! I need to terminate your session.
sensor#
```

The user jsmith receives the following message from the administrator jtaylor.

```
sensor#
***
***
*** Termination request from jtaylor
***
Sorry! I need to terminate your session.
```

Modifying Terminal Properties



Note You are not required to specify the screen length for some types of terminal sessions because the specified screen length can be learned by some remote hosts.

Use the **terminal [length] screen _length** command to modify terminal properties for a login session. The *screen_length* option lets you set the number of lines that appear on the screen before the `--more--` prompt is displayed. A value of zero results in no pause in the output. The default value is 24 lines.

To modify the terminal properties, follow these steps:

Step 1 Log in to the CLI.

Step 2 To have no pause between multi-screen outputs, use 0 for the screen length value.

```
sensor# terminal length 0
```



Note The screen length values are not saved between login sessions.

Step 3 To have the CLI pause and display the `--more--` prompt every 10 lines, use 10 for the *screen length* value.

```
sensor# terminal length 10
```

Configuring Events

This section describes how to display and clear events from the Event Store, and contains the following topics:

- [Displaying Events, page 17-22](#)
- [Clearing Events from the Event Store, page 17-25](#)

Displaying Events



Note

The Event Store has a fixed size of 30 MB for all platforms.



Note

Events are displayed as a live feed. To cancel the request, press **Ctrl-C**.

Use the **show events** [**alert** [informational] [low] [medium] [high] [**include-traits** *traits*] [**exclude-traits** *traits*] [**min-threat-rating** *min-rr*] [**max-threat-rating** *max-rr*] | **error** [warning] [error] [fatal] | **NAC** | **status**] [*hh:mm:ss* [*month day* [*year*]]] | **past** *hh:mm:ss*] command to display events from Event Store. Events are displayed beginning at the start time. If you do not specify a start time, events are displayed beginning at the current time. If you do not specify an event type, all events are displayed.

The following commands apply:

- **alert**—Displays alerts. Provides notification of some suspicious activity that may indicate an attack is in process or has been attempted. Alert events are generated by the Analysis Engine whenever a signature is triggered by network activity. If no level is selected (informational, low, medium, or high), all alert events are displayed.
- **include-traits**—Displays alerts that have the specified traits.
- **exclude-traits**—Does not display alerts that have the specified traits.
- **traits**—Specifies the trait bit position in decimal (0 to 15).

- **min-threat-rating**—Displays events with a threat rating above or equal to this value. The default is 0. The valid range is 0 to 100.
- **max-threat-rating**—Displays events with a threat rating below or equal to this value. The default is 100. The valid range is 0 to 100.
- **error**—Displays error events. Error events are generated by services when error conditions are encountered. If no level is selected (warning, error, or fatal), all error events are displayed.
- **NAC**—Displays the ARC (block) requests.



Note The ARC is formerly known as NAC. This name change has not been completely implemented throughout the IDM, the IME, and the CLI.

- **status**—Displays status events.
- **past**—Displays events starting in the past for the specified hours, minutes, and seconds.
- **hh:mm:ss**—Specifies the hours, minutes, and seconds in the past to begin the display.



Note The **show events** command continues to display events until a specified event is available. To exit, press **Ctrl-C**.

Displaying Events

To display events from the Event Store, follow these steps:

- Step 1** Log in to the CLI.
- Step 2** Display all events starting now. The feed continues showing all events until you press **Ctrl-C**.

```
sensor# show events
evError: eventId=1041472274774840147 severity=warning vendor=Cisco
  originator:
    hostId: sensor2
    appName: cidwebserver
    appInstanceId: 12075
  time: 2011/01/07 04:41:45 2011/01/07 04:41:45 UTC
  errorMessage: name=errWarning received fatal alert: certificate_unknown

evError: eventId=1041472274774840148 severity=error vendor=Cisco
  originator:
    hostId: sensor2
    appName: cidwebserver
    appInstanceId: 351
  time: 2011/01/07 04:41:45 2011/01/07 04:41:45 UTC
  errorMessage: name=errTransport WebSession::sessionTask(6) TLS connection exception: handshake incomplete.
```

- Step 3** Display the block requests beginning at 10:00 a.m. on February 9, 2011.

```
sensor# show events NAC 10:00:00 Feb 9 2011
evShunRqst: eventId=1106837332219222281 vendor=Cisco
  originator:
    deviceName: Sensor1
    appName: NetworkAccessControllerApp
    appInstance: 654
  time: 2011/02/09 10:33:31 2011/08/09 13:13:31
  shunInfo:
    host: connectionShun=false
```

```

srcAddr: 11.0.0.1
destAddr:
srcPort:
destPort:
protocol: numericType=0 other
timeoutMinutes: 40
evAlertRef: hostId=esendHost 123456789012345678
sensor#

```

Step 4 Display errors with the warning level starting at 10:00 a.m. on February 9, 2011.

```

sensor# show events error warning 10:00:00 Feb 9 2011
evError: eventId=1041472274774840197 severity=warning vendor=Cisco
originator:
  hostId: sensor
  appName: cidwebserver
  appInstanceId: 12160
time: 2011/01/07 04:49:25 2011/01/07 04:49:25 UTC
errorMessage: name=errWarning received fatal alert: certificate_unknown

```

Step 5 Display alerts from the past 45 seconds.

```

sensor# show events alert past 00:00:45

evIdsAlert: eventId=1109695939102805307 severity=medium vendor=Cisco
originator:
  hostId: sensor
  appName: sensorApp
  appInstanceId: 367
time: 2011/03/02 14:15:59 2011/03/02 14:15:59 UTC
signature: description=Nachi Worm ICMP Echo Request id=2156 version=S54
  subsigId: 0
  sigDetails: Nachi ICMP
interfaceGroup:
vlan: 0
participants:
  attacker:
    addr: locality=OUT 10.89.228.202
  target:
    addr: locality=OUT 10.89.150.185
riskRatingValue: 70
interface: fe0_1
protocol: icmp

evIdsAlert: eventId=1109695939102805308 severity=medium vendor=Cisco
originator:
--MORE--

```

Step 6 Display events that began 30 seconds in the past.

```

sensor# show events past 00:00:30
evStatus: eventId=1041526834774829055 vendor=Cisco
originator:
  hostId: sensor
  appName: mainApp
  appInstanceId: 2215
time: 2011/01/08 02:41:00 2011/01/08 02:41:00 UTC
controlTransaction: command=getVersion successful=true
description: Control transaction response.
requestor:
  user: cids
  application:
    hostId: 64.101.182.101
    appName: -cidcli

```



```
appInstanceId: 2316

evStatus: eventId=1041526834774829056 vendor=Cisco
originator:
  hostId: sensor
  appName: login(pam_unix)
  appInstanceId: 2315
time: 2011/01/08 02:41:00 2011/01/08 02:41:00 UTC
syslogMessage:
  description: session opened for user cisco by cisco(uid=0)
```

Clearing Events from the Event Store

Use the **clear events** command to clear the Event Store.

To clear events from the Event Store, follow these steps:

Step 1 Log in to the CLI using an account with administrator privileges.

Step 2 Clear the Event Store.

```
sensor# clear events
Warning: Executing this command will remove all events currently stored in the event
store.
Continue with clear? []:
```

Step 3 Enter **yes** to clear the events.

Configuring the System Clock

This section explains how to display and manually set the system clock. It contains the following topics:

- [Displaying the System Clock, page 17-25](#)
- [Manually Setting the System Clock, page 17-26](#)

Displaying the System Clock

Use the **show clock [detail]** command to display the system clock. You can use the **detail** option to indicate the clock source (NTP or system) and the current summertime setting (if any). The system clock keeps an authoritative flag that indicates whether the time is authoritative (believed to be accurate). If the system clock has been set by a timing source, such as NTP, the flag is set.

Table 17-3 lists the system clock flags.

Table 17-3 System Clock Flags

Symbol	Description
*	Time is not authoritative.
(blank)	Time is authoritative.
.	Time is authoritative, but NTP is not synchronized.

To display the system clock, follow these steps:

Step 1 Log in to the CLI.

Step 2 Display the system clock.

```
sensor# show clock
*19:04:52 UTC Thu Apr 03 2008
```

Step 3 Display the system clock with details. The following example indicates that the sensor is getting its time from NTP and that is configured and synchronized.

```
sensor# show clock detail
20:09:43 UTC Thu Apr 03 2011
Time source is NTP
Summer time starts 03:00:00 UTC Sun Mar 09 2011
Summer time stops 01:00:00 UTC Sun Nov 02 2011
```

Step 4 Display the system clock with details. The following example indicates that no time source is configured.

```
sensor# show clock detail
*20:09:43 UTC Thu Apr 03 2011
No time source
Summer time starts 03:00:00 UTC Sun Mar 09 2011
Summer time stops 01:00:00 UTC Sun Nov 02 2011
```

Manually Setting the System Clock



Note

You do not need to set the system clock if your sensor is synchronized by a valid outside timing mechanism such as an NTP clock source.

Use the **clock set** *hh:mm [:ss] month day year* command to manually set the clock on the appliance. Use this command if no other time sources are available. The **clock set** command does not apply to the following platforms, because they get their time from the adaptive security appliance in which they are installed:

- ASA 5500-X IPS SSP
- ASA 5585-X IPS SSP

To manually set the clock on the appliance, follow these steps:

Step 1 Log in to the CLI using an account with administrator privileges.

Step 2 Set the clock manually.

```
sensor# clock set 13:21 Mar 29 2011
```



Note The time format is 24-hour time.

Displaying and Clearing the Denied Attackers List

Use the **show statistics denied-attackers** command to display the list of denied attackers. Use the **clear denied-attackers** [*virtual_sensor*] [*ip-address ip_address*] command to delete the denied attackers list and clear the virtual sensor statistics.

If your sensor is configured to operate in inline mode, the traffic is passing through the sensor. You can configure signatures to deny packets, connections, and attackers while in inline mode, which means that single packets, connections, and specific attackers are denied, that is, not transmitted, when the sensor encounters them. When the signature fires, the attacker is denied and placed in a list. As part of sensor administration, you may want to delete the list or clear the statistics in the list.

The following commands apply:

- *virtual_sensor*—(Optional) Specifies the virtual sensor whose denied attackers list should be cleared.
- *ip_address*—(Optional) Specifies the IP address to clear.

Displaying and Deleting Denied Attackers

To display the list of denied attackers and delete the list and clear the statistics, follow these steps:

Step 1 Log in to the CLI using an account with administrator privileges.

Step 2 Display the list of denied IP addresses. The statistics show that there are two IP addresses being denied at this time.

```
sensor# show statistics denied-attackers
Denied Attackers and hit count for each.
 10.20.4.2 = 9
 10.20.5.2 = 5
```

Step 3 Delete the denied attackers list.

```
sensor# clear denied-attackers
Warning: Executing this command will delete all addresses from the list of attackers
currently being denied by the sensor.
Continue with clear? [yes]:
```

Step 4 Enter **yes** to clear the list.

Step 5 Delete the denied attackers list for a specific virtual sensor.

```
sensor# clear denied-attackers vs0
```

```
Warning: Executing this command will delete all addresses from the list of attackers being
denied by virtual sensor vs0.
Continue with clear? [yes]:
```

Step 6 Enter **yes** to clear the list.

Step 7 Remove a specific IP address from the denied attackers list for a specific virtual sensor.

```
sensor# clear denied-attackers vs0 ip-address 192.0.2.0
Warning: Executing this command will delete ip address 192.0.2.0 from the list of
attackers being denied by virtual sensor vs0.
Continue with clear? [yes]:
```

Step 8 Enter **yes** to clear the list.

Step 9 Verify that you have cleared the list. You can use the **show statistics denied-attackers** or **show statistics virtual-sensor** command.

```
sensor# show statistics denied-attackers
Denied Attackers and hit count for each.
Denied Attackers and hit count for each.
Statistics for Virtual Sensor vs0
  Denied Attackers with percent denied and hit count for each.

  Denied Attackers with percent denied and hit count for each.
```

```
Statistics for Virtual Sensor vs1
  Denied Attackers with percent denied and hit count for each.
```

```
  Denied Attackers with percent denied and hit count for each.
sensor#
```

```
sensor# show statistics virtual-sensor
Virtual Sensor Statistics
Statistics for Virtual Sensor vs0
  Name of current Signature-Definition instance = sig0
  Name of current Event-Action-Rules instance = rules0
  List of interfaces monitored by this virtual sensor = mypair
  Denied Address Information
    Number of Active Denied Attackers = 0
    Number of Denied Attackers Inserted = 2
    Number of Denied Attackers Total Hits = 287
    Number of times max-denied-attackers limited creation of new entry = 0
    Number of exec Clear commands during uptime = 1
  Denied Attackers and hit count for each.
```

Step 10 Clear only the statistics.

```
sensor# show statistics virtual-sensor clear
```

Step 11 Verify that you have cleared the statistics. The statistics have all been cleared except for the Number of Active Denied Attackers and Number of exec Clear commands during uptime categories. It is important to know if the list has been cleared.

```
sensor# show statistics virtual-sensor
Virtual Sensor Statistics
Statistics for Virtual Sensor vs0
  Name of current Signature-Definition instance = sig0
  Name of current Event-Action-Rules instance = rules0
  List of interfaces monitored by this virtual sensor = mypair
  Denied Address Information
    Number of Active Denied Attackers = 2
```

```

Number of Denied Attackers Inserted = 0
Number of Denied Attackers Total Hits = 0
Number of times max-denied-attackers limited creation of new entry = 0
Number of exec Clear commands during uptime = 1
Denied Attackers and hit count for each.
10.20.2.5 = 0
10.20.5.2 = 0

```

Displaying Policy Lists

Use the **list {anomaly-detection-configurations | event-action-rules-configurations | signature-definition-configurations}** in EXEC mode to display the list of policies for these components. The file size is in bytes. A virtual sensor with N/A indicates that the policy is not assigned to a virtual sensor.

To display a list of policies on the sensor, follow these steps:

Step 1 Log in to the CLI.

Step 2 Display the list of policies for anomaly detection.

```

sensor# list anomaly-detection-configurations
Anomaly Detection
  Instance  Size  Virtual Sensor
  -----  ---  -
  ad0       255  vs0
  temp      707  N/A
  MyAD      255  N/A
  ad1       141  vs1
sensor#

```

Step 3 Display the list of policies for event action rules.

```

sensor# list event-action-rules-configurations
Event Action Rules
  Instance  Size  Virtual Sensor
  -----  ---  -
  rules0    112  vs0
  rules1    141  vs1
sensor#

```

Step 4 Display the list of policies for signature definition.

```

sensor# list signature-definition-configurations
Signature Definition
  Instance  Size  Virtual Sensor
  -----  ---  -
  sig0      336  vs0
  sig1      141  vs1
  sig2      141  N/A
sensor#

```

Displaying Statistics

Use the **show statistics** [**analysis-engine** | **anomaly-detection** | **authentication** | **denied-attackers** | **event-server** | **event-store** | **external-product-interface** | **global-correlation** | **host** | **logger** | **network-access** | **notification** | **os-identification** | **sdee-server** | **transaction-server** | **virtual-sensor** | **web-server**] [**clear**] command to display statistics for each sensor application.

Use the **show statistics** {**anomaly-detection** | **denied-attackers** | **os-identification** | **virtual-sensor**} [**name** | **clear**] command to display statistics for these components for all virtual sensors. If you provide the virtual sensor name, the statistics for that virtual sensor only are displayed.



Note

The **clear** option is not available for the analysis engine, anomaly detection, host, network access, or OS identification applications.

For the IPS 4500 series, at the end of the command output, there are extra details for the Ethernet controller statistics, such as the total number of packets received at the Ethernet controller, the total number of packets dropped at the Ethernet controller under high load conditions, and the total packets transmitted including the customer traffic packets and the internal keepalive packet count.



Note

The Ethernet controller statistics are polled at an interval of 5 seconds from the hardware side. The keepalives are sent or updated at an interval of 10 ms. Because of this, there may be a disparity in the actual count reflected in the total packets transmitted. At times, it is even possible that the total packets transmitted may be less than the keepalive packets transmitted.

To display statistics for the sensor, follow these steps:

- Step 1** Log in to the CLI.
- Step 2** Display the statistics for the Analysis Engine.

```

sensor# show statistics analysis-engine
Analysis Engine Statistics
  Number of seconds since service started = 318375
  Processing Load Percentage
    Thread   5 sec  1 min   5 min
    0         1      1      1
    1         1      1      1
    2         1      1      1
    3         1      1      1
    Average  1      1      1

The rate of TCP connections tracked per second = 0
The rate of packets per second = 0
The rate of bytes per second = 0
Receiver Statistics
  Total number of packets processed since reset = 0
  Total number of IP packets processed since reset = 0
Transmitter Statistics
  Total number of packets transmitted = 129766
  Total number of packets denied = 0
  Total number of packets reset = 0
Fragment Reassembly Unit Statistics
  Number of fragments currently in FRU = 0
  Number of datagrams currently in FRU = 0
TCP Stream Reassembly Unit Statistics
  TCP streams currently in the embryonic state = 0

```

```

TCP streams currently in the established state = 0
TCP streams currently in the closing state = 0
TCP streams currently in the system = 0
TCP Packets currently queued for reassembly = 0
The Signature Database Statistics.
Total nodes active = 0
TCP nodes keyed on both IP addresses and both ports = 0
UDP nodes keyed on both IP addresses and both ports = 0
IP nodes keyed on both IP addresses = 0
Statistics for Signature Events
Number of SigEvents since reset = 0
Statistics for Actions executed on a SigEvent
Number of Alerts written to the IdsEventStore = 0
Inspection Stats

```

```

GlobalCorrelationStats
SwVersion = 7.3(0.119)E4
SigVersion = 741.0
DatabaseRecordCount = 0
DatabaseVersion = 0
RuleVersion = 0
ReputationFilterVersion = 0
AlertsWithHit = 0
AlertsWithMiss = 0
AlertsWithModifiedRiskRating = 0
AlertsWithGlobalCorrelationDenyAttacker = 0
AlertsWithGlobalCorrelationDenyPacket = 0
AlertsWithGlobalCorrelationOtherAction = 0
AlertsWithAuditRepDenies = 0
ReputationForcedAlerts = 0
EventStoreInsertTotal = 0
EventStoreInsertWithHit = 0
EventStoreInsertWithMiss = 0
EventStoreDenyFromGlobalCorrelation = 0
EventStoreDenyFromOverride = 0
EventStoreDenyFromOverlap = 0
EventStoreDenyFromOther = 0
ReputationFilterDataSize = 0
ReputationFilterPacketsInput = 0
ReputationFilterRuleMatch = 0
DenyFilterHitsNormal = 0
DenyFilterHitsGlobalCorrelation = 0
SimulatedReputationFilterPacketsInput = 0
SimulatedReputationFilterRuleMatch = 0
SimulatedDenyFilterInsert = 0
SimulatedDenyFilterPacketsInput = 0
SimulatedDenyFilterRuleMatch = 0
TcpDeniesDueToGlobalCorrelation = 0
TcpDeniesDueToOverride = 0
TcpDeniesDueToOverlap = 0
TcpDeniesDueToOther = 0
SimulatedTcpDeniesDueToGlobalCorrelation = 0
SimulatedTcpDeniesDueToOverride = 0
SimulatedTcpDeniesDueToOverlap = 0
SimulatedTcpDeniesDueToOther = 0
LateStageDenyDueToGlobalCorrelation = 0
LateStageDenyDueToOverride = 0
LateStageDenyDueToOverlap = 0
LateStageDenyDueToOther = 0
SimulatedLateStageDenyDueToGlobalCorrelation = 0
SimulatedLateStageDenyDueToOverride = 0
SimulatedLateStageDenyDueToOverlap = 0
SimulatedLateStageDenyDueToOther = 0

```

```

AlertHistogram
RiskHistogramEarlyStage
RiskHistogramLateStage
ConfigAggressiveMode = 0
ConfigAuditMode = 0
RegexAccelerationStats
Status = Enabled
DriverVersion = 6.3.1
Devices = 1
Agents = 12
Flows = 7
Channels = 0
SubmittedJobs = 193
CompletedJobs = 193
SubmittedBytes = 2836764
CompletedBytes = 0
TCPFlowsWithoutLCB = 0
UDPFlowsWithoutLCB = 0
TCPMissedPacketsDueToUpdate = 0
UDPMissedPacketsDueToUpdate = 0
MemorySize = 1073741824
HostDirectMemSize = 0
MaliciousSiteDenyHitCounts
MaliciousSiteDenyHitCountsAUDIT
Ethernet Controller Statistics
Total Packets Received = 0
Total Received Packets Dropped = 0
Total Packets Transmitted = 13643"
sensor#

```

Step 3 Display the statistics for anomaly detection.

```

sensor# show statistics anomaly-detection
Statistics for Virtual Sensor vs0
No attack
Detection - ON
Learning - ON
Next KB rotation at 10:00:01 UTC Sat Jan 18 2008
Internal Zone
TCP Protocol
UDP Protocol
Other Protocol
External Zone
TCP Protocol
UDP Protocol
Other Protocol
Illegal Zone
TCP Protocol
UDP Protocol
Other Protocol
Statistics for Virtual Sensor vs1
No attack
Detection - ON
Learning - ON
Next KB rotation at 10:00:00 UTC Sat Jan 18 2008
Internal Zone
TCP Protocol
UDP Protocol
Other Protocol
External Zone
TCP Protocol
UDP Protocol
Other Protocol
Illegal Zone

```



```

    TCP Protocol
    UDP Protocol
    Other Protocol
sensor#

```

Step 4 Display the statistics for authentication.

```

sensor# show statistics authentication
General
  totalAuthenticationAttempts = 128
  failedAuthenticationAttempts = 0
sensor#

```

Step 5 Display the statistics for the denied attackers in the system.

```

sensor# show statistics denied-attackers
Denied Attackers and hit count for each.
Denied Attackers and hit count for each.
Statistics for Virtual Sensor vs0
  Denied Attackers with percent denied and hit count for each.

  Denied Attackers with percent denied and hit count for each.

Statistics for Virtual Sensor vs1
  Denied Attackers with percent denied and hit count for each.

  Denied Attackers with percent denied and hit count for each.

sensor#

```

Step 6 Display the statistics for the Event Server.

```

sensor# show statistics event-server
General
  openSubscriptions = 0
  blockedSubscriptions = 0
Subscriptions
sensor#

```

Step 7 Display the statistics for the Event Store.

```

sensor# show statistics event-store
EEvent store statistics
  General information about the event store
    The current number of open subscriptions = 2
    The number of events lost by subscriptions and queries = 0
    The number of filtered events not written to the event store = 850763
    The number of queries issued = 0
    The number of times the event store circular buffer has wrapped = 0
  Number of events of each type currently stored
    Status events = 4257
    Shun request events = 0
    Error events, warning = 669
    Error events, error = 8
    Error events, fatal = 0
    Alert events, informational = 0
    Alert events, low = 0
    Alert events, medium = 0
    Alert events, high = 0
    Alert events, threat rating 0-20 = 0
    Alert events, threat rating 21-40 = 0

```

```

Alert events, threat rating 41-60 = 0
Alert events, threat rating 61-80 = 0
Alert events, threat rating 81-100 = 0
Cumulative number of each type of event
Status events = 4257
Shun request events = 0
Error events, warning = 669
Error events, error = 8
Error events, fatal = 0
Alert events, informational = 0
Alert events, low = 0
Alert events, medium = 0
Alert events, high = 0
Alert events, threat rating 0-20 = 0
Alert events, threat rating 21-40 = 0
Alert events, threat rating 41-60 = 0
Alert events, threat rating 61-80 = 0
Alert events, threat rating 81-100 = 0
sensor#

```

Step 8 Display the statistics for global correlation.

```

sensor# show statistics global-correlation
Network Participation:
  Counters:
    Total Connection Attempts = 0
    Total Connection Failures = 0
    Connection Failures Since Last Success = 0
  Connection History:
Updates:
  Status Of Last Update Attempt = Disabled
  Time Since Last Successful Update = never
  Counters:
    Update Failures Since Last Success = 0
    Total Update Attempts = 0
    Total Update Failures = 0
  Update Interval In Seconds = 300
  Update Server = update-manifests.ironport.com
  Update Server Address = Unknown
  Current Versions:
Warnings:
  Unlicensed = Global correlation inspection and reputation filtering have been
  disabled because the sensor is unlicensed.
  Action Required = Obtain a new license from http://www.cisco.com/go/license.
sensor#

```

Step 9 Display the statistics for the host.

```

sensor# show statistics host
General Statistics
  Last Change To Host Config (UTC) = 15-Oct-2013 09:11:13
  Command Control Port Device = Management0/0
Network Statistics
  = ma0_0      Link encap:Ethernet HWaddr 00:06:F6:2B:48:4C
  =            inet addr:10.106.132.150 Bcast:10.106.133.255 Mask:255.255.254.0
  =            UP BROADCAST RUNNING MULTICAST MTU:1500 Metric:1
  =            RX packets:429575 errors:0 dropped:0 overruns:0 frame:0
  =            TX packets:5492 errors:0 dropped:0 overruns:0 carrier:0
  =            collisions:0 txqueuelen:100
  =            RX bytes:30017118 (28.6 MiB) TX bytes:1064828 (1.0 MiB)
  =            Base address:0xe000 Memory:fb800000-fb820000
NTP Statistics
  status = Not applicable
Memory Usage

```

```

usedBytes = 6630588416
freeBytes = 1615556608
totalBytes = 8246145024
CPU Statistics
Note: CPU Usage statistics are not a good indication of the sensor processing
load. The Inspection Load Percentage in the output of 'show inspection-load' sh
ould be used instead.
Usage over last 5 seconds = 100
Usage over last minute = 100
Usage over last 5 minutes = 99
Usage over last 5 seconds = 2
Usage over last minute = 1
Usage over last 5 minutes = 1
Usage over last 5 seconds = 100
Usage over last minute = 100
Usage over last 5 minutes = 100
Usage over last 5 seconds = 100
Usage over last minute = 100
Usage over last 5 minutes = 100
Memory Statistics
Memory usage (bytes) = 6630588416
Memory free (bytes) = 1615556608
Auto Update Statistics
lastDirectoryReadAttempt = N/A
lastDownloadAttempt = N/A
lastInstallAttempt = N/A
nextAttempt = N/A
Auxiliary Processors Installed
sensor#

```

Step 10 Display the statistics for the logging application.

```

sensor# show statistics logger
The number of Log interprocessor FIFO overruns = 0
The number of syslog messages received = 11
The number of <evError> events written to the event store by severity
Fatal Severity = 0
Error Severity = 64
Warning Severity = 35
TOTAL = 99
The number of log messages written to the message log by severity
Fatal Severity = 0
Error Severity = 64
Warning Severity = 24
Timing Severity = 311
Debug Severity = 31522
Unknown Severity = 7
TOTAL = 31928
sensor#

```

Step 11 Display the statistics for the ARC.

```

sensor# show statistics network-access
Current Configuration
LogAllBlockEventsAndSensors = true
EnableNvramWrite = false
EnableAclLogging = false
AllowSensorBlock = false
BlockMaxEntries = 11
MaxDeviceInterfaces = 250
NetDevice
Type = PIX
IP = 10.89.150.171
NATAddr = 0.0.0.0

```

```

    Communications = ssh-3des
NetDevice
  Type = PIX
  IP = 192.0.2.4
  NATAddr = 0.0.0.0
  Communications = ssh-3des
NetDevice
  Type = PIX
  IP = 192.0.2.5
  NATAddr = 0.0.0.0
  Communications = telnet
NetDevice
  Type = Cisco
  IP = 192.0.2.6
  NATAddr = 0.0.0.0
  Communications = telnet
  BlockInterface
    InterfaceName = ethernet0/1
    InterfaceDirection = out
    InterfacePostBlock = Post_Acl_Test
  BlockInterface
    InterfaceName = ethernet0/1
    InterfaceDirection = in
    InterfacePreBlock = Pre_Acl_Test
    InterfacePostBlock = Post_Acl_Test
NetDevice
  Type = CAT6000_VACL
  IP = 192.0.2.1
  NATAddr = 0.0.0.0
  Communications = telnet
  BlockInterface
    InterfaceName = 502
    InterfacePreBlock = Pre_Acl_Test
  BlockInterface
    InterfaceName = 507
    InterfacePostBlock = Post_Acl_Test
State
  BlockEnable = true
NetDevice
  IP = 192.0.2.3
  AclSupport = Does not use ACLs
  Version = 6.3
  State = Active
  Firewall-type = PIX
NetDevice
  IP = 192.0.2.7
  AclSupport = Does not use ACLs
  Version = 7.0
  State = Active
  Firewall-type = ASA
NetDevice
  IP = 102.0.2.8
  AclSupport = Does not use ACLs
  Version = 2.2
  State = Active
  Firewall-type = FWSM
NetDevice
  IP = 192.0.2.9
  AclSupport = uses Named ACLs
  Version = 12.2
  State = Active
NetDevice
  IP = 192.0.2.10
  AclSupport = Uses VACLs

```

```

Version = 8.4
State = Active
BlockedAddr
Host
  IP = 203.0.113.1
  Vlan =
  ActualIp =
  BlockMinutes =
Host
  IP = 203.0.113.2
  Vlan =
  ActualIp =
  BlockMinutes =
Host
  IP = 203.0.113.4
  Vlan =
  ActualIp =
  BlockMinutes = 60
  MinutesRemaining = 24
Network
  IP = 203.0.113.9
  Mask = 255.255.0.0
  BlockMinutes =
sensor#

```

Step 12 Display the statistics for the notification application.

```

sensor# show statistics notification
General
  Number of SNMP set requests = 0
  Number of SNMP get requests = 0
  Number of error traps sent = 0
  Number of alert traps sent = 0
sensor#

```

Step 13 Display the statistics for OS identification.

```

sensor# show statistics os-identification
Statistics for Virtual Sensor vs0
  OS Identification
    Configured
    Imported
    Learned
sensor#

```

Step 14 Display the statistics for the SDEE server.

```

sensor# show statistics sdee-server
General
  Open Subscriptions = 4
  Blocked Subscriptions = 0
  Maximum Available Subscriptions = 5
  Maximum Events Per Retrieval = 500
Subscriptions
  sub-1-9ce8d1dc
    State = Open
    IP Address = 10.89.11.67
    Last Read Time = 18:32:19 UTC Fri Jan 31 2014
    Last Read Time (nanoseconds) = 1391193139172303000
    Status = Valid
  sub-2-8701fb19
    State = Open
    IP Address = 10.142.52.127
    Last Read Time = 18:32:17 UTC Fri Jan 31 2014
    Last Read Time (nanoseconds) = 1391193137284703000

```

```

    Status = Valid
sub-3-5aae57d4
    State = Open
    IP Address = 10.142.52.151
    Last Read Time = 18:32:17 UTC Fri Jan 31 2014
    Last Read Time (nanoseconds) = 1391193137250568000
    Status = Valid
sub-4-c4685a12
    State = Open
    IP Address = 10.142.52.110
    Last Read Time = 18:32:19 UTC Fri Jan 31 2014
    Last Read Time (nanoseconds) = 1391193139400954000
    Status = Valid
sensor#

```

Step 15 Display the statistics for the transaction server.

```

sensor# show statistics transaction-server
General
    totalControlTransactions = 35
    failedControlTransactions = 0
sensor#

```

Step 16 Display the statistics for a virtual sensor.

```

sensor# show statistics virtual-sensor vs0

Name of current Signature-Definition instance = sig0
Name of current Event-Action-Rules instance = rules0
List of interfaces monitored by this virtual sensor =
General Statistics for this Virtual Sensor
    Number of seconds since a reset of the statistics = 318712
    MemoryAlloPercent = 13
    MemoryUsedPercent = 13
    MemoryMaxCapacity = 5500000
    MemoryMaxHighUsed = 750721
    MemoryCurrentAllo = 769921
    MemoryCurrentUsed = 750721
    Inspection Load Percentage = 0
    Total packets processed since reset = 0
    Total IP packets processed since reset = 0
    Total IPv4 packets processed since reset = 0
    Total IPv6 packets processed since reset = 0
    Total IPv6 AH packets processed since reset = 0
    Total IPv6 ESP packets processed since reset = 0
    Total IPv6 Fragment packets processed since reset = 0
    Total IPv6 Routing Header packets processed since reset = 0
    Total IPv6 ICMP packets processed since reset = 0
    Total packets that were not IP processed since reset = 0
    Total TCP packets processed since reset = 0
    Total UDP packets processed since reset = 0
    Total ICMP packets processed since reset = 0
    Total packets that were not TCP, UDP, or ICMP processed since reset = 0
    Total ARP packets processed since reset = 0
    Total ISL encapsulated packets processed since reset = 0
    Total 802.1q encapsulated packets processed since reset = 0
    Total GRE Packets processed since reset = 0
    Total GRE Fragment Packets processed since reset = 0
    Total GRE Packets skipped since reset = 0
    Total GRE Packets with Bad Header skipped since reset = 0
    Total IpIp Packets with Bad Header skipped since reset = 0
    Total Encapsulated Tunnel Packets with Bad Header skipped since reset = 0
    Total packets with bad IP checksums processed since reset = 0
    Total packets with bad layer 4 checksums processed since reset = 0

```

```

Total cross queue TCP packets processed since reset = 0
Total cross queue UDP packets processed since reset = 0
HTTP transfer encoding errors = 0
HTTP content encoding errors = 0
HTTP character encoding errors = 0
HTTP connection out of sync = 0
HTTP pipelining or persistence out of sync = 0
Total number of bytes processed since reset = 0
The rate of packets per second since reset = 0
The rate of bytes per second since reset = 0
The average bytes per packet since reset = 0
Denied Address Information
Number of Active Denied Attackers = 0
Number of Denied Attackers Inserted = 0
Number of Denied Attacker Victim Pairs Inserted = 0
Number of Denied Attacker Service Pairs Inserted = 0
Number of Denied Attackers Total Hits = 0
Number of times max-denied-attackers limited creation of new entry = 0
Number of exec Clear commands during uptime = 0
Denied Attackers and hit count for each.
Denied Attackers with percent denied and hit count for each.

AlarmDB Statistics for this Virtual Sensor
Total IP nodes keyed on both IP addresses since reset = 0
Active IP nodes keyed on both IP addresses = 0
IP nodes prevented since reset = 0
The Signature Database Statistics.
The Number of each type of node active in the system
Total nodes active = 0
TCP nodes keyed on both IP addresses and both ports = 0
UDP nodes keyed on both IP addresses and both ports = 0
IP nodes keyed on both IP addresses = 0
The number of each type of node inserted since reset
Total nodes inserted = 0
TCP nodes keyed on both IP addresses and both ports = 0
UDP nodes keyed on both IP addresses and both ports = 0
IP nodes keyed on both IP addresses = 0
The rate of nodes per second for each time since reset
Nodes per second = 0
TCP nodes keyed on both IP addresses and both ports per second = 0
UDP nodes keyed on both IP addresses and both ports per second = 0
IP nodes keyed on both IP addresses per second = 0
The number of root nodes forced to expire because of memory constraints
TCP nodes keyed on both IP addresses and both ports = 0
Packets dropped because they would exceed Database insertion rate limits = 0
Fragment Reassembly Unit Statistics for this Virtual Sensor
Number of fragments currently in FRU = 0
Number of datagrams currently in FRU = 0
Number of fragments received since reset = 0
Number of fragments forwarded since reset = 0
Number of fragments dropped since last reset = 0
Number of fragments modified since last reset = 0
Number of complete datagrams reassembled since last reset = 0
Fragments hitting too many fragments condition since last reset = 0
Number of overlapping fragments since last reset = 0
Number of Datagrams too big since last reset = 0
Number of overwriting fragments since last reset = 0
Number of Initial fragment missing since last reset = 0
Fragments hitting the max partial datagrams limit since last reset = 0
Fragments too small since last reset = 0
Too many fragments per datagram limit since last reset = 0
Number of datagram reassembly timeout since last reset = 0
Too many fragments claiming to be the last since last reset = 0

```

```

    Fragments with bad fragment flags since last reset = 0
TCP Normalizer stage statistics
  Packets Input = 0
  Packets Modified = 0
  Dropped packets from queue = 0
  Dropped packets due to deny-connection = 0
  Duplicate Packets = 0
  Current Streams = 0
  Current Streams Closed = 0
  Current Streams Closing = 0
  Current Streams Embryonic = 0
  Current Streams Established = 0
  Current Streams Denied = 0
  Total SendAck Limited Packets = 0
  Total SendAck Limited Streams = 0
  Total SendAck Packets Sent = 0
Statistics for the TCP Stream Reassembly Unit
  Current Statistics for the TCP Stream Reassembly Unit
    TCP streams currently in the embryonic state = 0
    TCP streams currently in the established state = 0
    TCP streams currently in the closing state = 0
    TCP streams currently in the system = 0
    TCP Packets currently queued for reassembly = 0
  Cumulative Statistics for the TCP Stream Reassembly Unit since reset
    TCP streams that have been tracked since last reset = 0
    TCP streams that had a gap in the sequence jumped = 0
    TCP streams that was abandoned due to a gap in the sequence = 0
    TCP packets that arrived out of sequence order for their stream = 0
    TCP packets that arrived out of state order for their stream = 0
    The rate of TCP connections tracked per second since reset = 0
SigEvent Preliminary Stage Statistics
  Number of Alerts received = 0
  Number of Alerts Consumed by AlertInterval = 0
  Number of Alerts Consumed by Event Count = 0
  Number of FireOnce First Alerts = 0
  Number of FireOnce Intermediate Alerts = 0
  Number of Summary First Alerts = 0
  Number of Summary Intermediate Alerts = 0
  Number of Regular Summary Final Alerts = 0
  Number of Global Summary Final Alerts = 0
  Number of Active SigEventDataNodes = 0
  Number of Alerts Output for further processing = 0
--MORE--

```

Step 17 Display the statistics for the web server.

```

sensor# show statistics web-server
listener-443
  session-11
    remote host = 64.101.182.167
    session is persistent = no
    number of requests serviced on current connection = 1
    last status code = 200
    last request method = GET
    last request URI = cgi-bin/sdee-server
    last protocol version = HTTP/1.1
    session state = processingGetServlet
    number of server session requests handled = 957134
    number of server session requests rejected = 0
    total HTTP requests handled = 365871
    maximum number of session objects allowed = 40
    number of idle allocated session objects = 12
    number of busy allocated session objects = 1
  summarized log messages

```



```

number of TCP socket failure messages logged = 0
number of TLS socket failure messages logged = 0
number of TLS protocol failure messages logged = 0
number of TLS connection failure messages logged = 595015
number of TLS crypto warning messages logged = 0
number of TLS expired certificate warning messages logged = 0
number of receipt of TLS fatal alert message messages logged = 594969
crypto library version = 6.2.1.0
sensor#

```

- Step 18** Clear the statistics for an application, for example, the logging application. The statistics are retrieved and cleared.

```

sensor# show statistics logger clear
The number of Log interprocessor FIFO overruns = 0
The number of syslog messages received = 141
The number of <evError> events written to the event store by severity
  Fatal Severity = 0
  Error Severity = 14
  Warning Severity = 142
  TOTAL = 156
The number of log messages written to the message log by severity
  Fatal Severity = 0
  Error Severity = 14
  Warning Severity = 1
  Timing Severity = 0
  Debug Severity = 0
  Unknown Severity = 28
  TOTAL = 43

```

- Step 19** Verify that the statistics have been cleared. The statistics now all begin from 0.

```

sensor# show statistics logger
The number of Log interprocessor FIFO overruns = 0
The number of syslog messages received = 0
The number of <evError> events written to the event store by severity
  Fatal Severity = 0
  Error Severity = 0
  Warning Severity = 0
  TOTAL = 0
The number of log messages written to the message log by severity
  Fatal Severity = 0
  Error Severity = 0
  Warning Severity = 0
  Timing Severity = 0
  Debug Severity = 0
  Unknown Severity = 0
  TOTAL = 0
sensor#

```

Displaying Tech Support Information



Note

The `/var/log/messages` file is persistent across reboots and the information is displayed in the output of the `show tech-support` command.

**Note**

The **show tech-support** command displays historical interface data for each interface for the past 72 hours.

Use the **show tech-support** [**page**] [**destination-url** *destination_url*] command to display system information on the screen or have it sent to a specific URL. You can use the information as a troubleshooting tool with TAC.

The following parameters are optional:

- **page**—Displays the output, one page of information at a time. Press **Enter** to display the next line of output or use the spacebar to display the next page of information.
- **destination-url**—Indicates the information should be formatted as HTML and sent to the destination that follows this command. If you use this keyword, the output is not displayed on the screen.
- *destination_url*—Indicates the information should be formatted as HTML. The URL specifies where the information should be sent. If you do not use this keyword, the information is displayed on the screen.
- You can specify the following destination types:
 - **ftp**—Destination URL for FTP network server. The syntax for this prefix is:
`ftp://[[username@location]/relativeDirectory]/filename` or
`ftp://[[username@location]/absoluteDirectory]/filename`
 - **scp**—Destination URL for the SCP network server. The syntax for this prefix is:
`scp://[[username@]location]/relativeDirectory]/filename` or
`scp://[[username@]location]/absoluteDirectory]/filename`

Varlog Files

The `/var/log/messages` file has the latest logs. A new softlink called `varlog` has been created under the `/usr/cids/idsRoot/log` folder that points to the `/var/log/messages` file. Old logs are stored in `varlog.1` and `varlog.2` files. The maximum size of these `varlog` files is 200 KB. Once they cross the size limit the content is rotated. The content of `varlog`, `varlog.1`, and `varlog.2` is displayed in the output of the **show tech-support** command.

Displaying Tech Support Information

To display tech support information, follow these steps:

-
- Step 1** Log in to the CLI using an account with administrator privileges.
- Step 2** View the output on the screen. The system information appears on the screen, one page at a time. Press the spacebar to view the next page or press **Ctrl-C** to return to the prompt

```
sensor# show tech-support page
```

- Step 3** To send the output (in HTML format) to a file:

- a. Enter the following command, followed by a valid destination. The `password:` prompt appears.

```
sensor# show tech-support destination-url destination_url
```

Example

To send the tech support output to the file `/absolute/reports/sensor1Report.html`:

```
sensor# show tech support dest
ftp://csidsuser@10.2.1.2//absolute/reports/sensor1Report.html
```

- b. Enter the password for this user account. The `Generating report:` message is displayed.

Displaying Version Information

Use the **show version** command to display version information for all installed operating system packages, signature packages, and IPS processes running on the system. To view the configuration for the entire system, use the **more current-config** command.



Note

The CLI output is an example of what your configuration may look like. It will not match exactly due to the optional setup choices, sensor model, and IPS version you have installed.



Note

For the IPS 4500 series sensors, the **show version** command output contains an extra application called the SwitchApp.

To display the version and configuration, follow these steps:

Step 1 Log in to the CLI.

Step 2 View version information.

```

sensor# show version
Application Partition:

Cisco Intrusion Prevention System, Version 7.3(1)E4

Host:
  Realm Keys          key1.0
Signature Definition:
  Signature Update    S741.0      2013-09-10
  Threat Profile Version 2
OS Version:          2.6.29.1
Platform:             IPS-4360
Serial Number:        FGL1702401M
Licensed, expires:    21-Nov-2014 UTC
Sensor up-time is 22:09.
Using 14372M out of 15943M bytes of available memory (90% usage)
system is using 32.4M out of 160.0M bytes of available disk space (20% usage)
application-data is using 85.6M out of 376.4M bytes of available disk space (24% usage)
boot is using 63.1M out of 70.2M bytes of available disk space (95% usage)
application-log is using 494.0M out of 513.0M bytes of available disk space (96% usage)

MainApp              C-2013_12_16_14_00_7_3_0_143  (Release)  2013-12-16T14:06:20-0600
Running
AnalysisEngine       C-2013_12_16_14_00_7_3_0_143  (Release)  2013-12-16T14:06:20-0600
Running
CollaborationApp     C-2013_12_16_14_00_7_3_0_143  (Release)  2013-12-16T14:06:20-0600
Running
CLI                  C-2013_12_16_14_00_7_3_0_143  (Release)  2013-12-16T14:06:20-0600

Upgrade History:

```

```

IPS-K9-7.3-1-E4 11:22:07 UTC Sat Jan 19 2013

Recovery Partition Version 1.1 - 7.3(1)E4

Host Certificate Valid from: 09-Oct-2014 to 09-Oct-2016
sensor#

```



Note If the `--MORE--` prompt is displayed, press the spacebar to see more information or **Ctrl-C** to cancel the output and get back to the CLI prompt.

Step 3 View configuration information.



Note You can use the **more current-config** or **show configuration** commands.

```

sensor# more current-config
! -----
! Current configuration last modified Fri Oct 10 09:41:17 2014
! -----
! Version 7.3(1)
! Host:
!   Realm Keys          key1.0
! Signature Definition:
!   Signature Update    S741.0   2013-09-10
!   Threat Profile Version 2
! -----
service interface
exit
! -----
service authentication
exit
! -----
service event-action-rules rules0
exit
! -----
service host
network-settings
host-ip 10.106.1.4/24,10.106.1.1
host-name sensor
telnet-option enabled
access-list 0.0.0.0/0
dns-primary-server disabled
exit
time-zone-settings
offset 0
standard-time-zone-name UTC
exit
exit
! -----
service logger
exit
! -----
service network-access
exit
! -----
service notification
exit
! -----
service signature-definition sig0
exit

```

```

! -----
service ssh-known-hosts
exit
! -----
service trusted-certificates
exit
! -----
service web-server
exit
! -----
service anomaly-detection ad0
exit
! -----
service external-product-interface
exit
! -----
service health-monitor
exit
! -----
service global-correlation
exit
! -----
service aaa
exit
! -----
service analysis-engine
exit

sensor#

```

Diagnosing Network Connectivity



Caution

No command interrupt is available for this command. It must run to completion.

Use the **ping** *ip_address* [**count**] command to diagnose basic network connectivity.

To diagnose basic network connectivity, follow these steps:

- Step 1** Log in to the CLI.
- Step 2** Ping the address you are interested in. The count is the number of echo requests to send. If you do not specify a number, 4 requests are sent. The range is 1 to 10,000.

```
sensor# ping ip_address count
```

The following example shows a successful ping:

```

sensor# ping 192.0.2.1 6
PING 192.0.2.1 (192.0.2.1): 56 data bytes
64 bytes from 192.0.2.1: icmp_seq=0 ttl=61 time=0.3 ms
64 bytes from 192.0.2.1: icmp_seq=1 ttl=61 time=0.1 ms
64 bytes from 192.0.2.1: icmp_seq=2 ttl=61 time=0.1 ms
64 bytes from 192.0.2.1: icmp_seq=3 ttl=61 time=0.2 ms
64 bytes from 192.0.2.1: icmp_seq=4 ttl=61 time=0.2 ms
64 bytes from 192.0.2.1: icmp_seq=5 ttl=61 time=0.2 ms

```

```
--- 192.0.2.1 ping statistics ---
6 packets transmitted, 6 packets received, 0% packet loss
round-trip min/avg/max = 0.1/0.1/0.3 ms
```

The following example shows an unsuccessful ping:

```
sensor# ping 172.16.0.0 3
PING 172.16.0.0 (172.16.0.0): 56 data bytes

--- 172.16.0.0 ping statistics ---
3 packets transmitted, 0 packets received, 100% packet loss
sensor#
```

Resetting the Appliance

Use the **reset [powerdown]** command to shut down the applications running on the appliance and to reboot the appliance. You can include the **powerdown** option to power off the appliance, if possible, or to have the appliance left in a state where the power can be turned off.

Shutdown (stopping the applications) begins immediately after you execute the command. Shutdown can take a while, and you can still access CLI commands while it is taking place, but the session is terminated without warning.

To reset the appliance, follow these steps:

-
- Step 1** Log in to the CLI using an account with administrator privileges.
 - Step 2** To stop all applications and reboot the appliance, follow these Steps 2 and 3. Otherwise, to power down the appliance, go to Steps 4 and 5.

```
sensor# reset
Warning: Executing this command will stop all applications and reboot the node.
Continue with reset? []:
```

- Step 3** Enter **yes** to continue the reset.

```
sensor# yes
Request Succeeded.
sensor#
```

- Step 4** Stop all applications and power down the appliance.

```
sensor# reset powerdown
Warning: Executing this command will stop all applications and power off the node if
possible. If the node can not be powered off it will be left in a state that is safe to
manually power down.
Continue with reset? []:
```

- Step 5** Enter **yes** to continue with the reset and power down.

```
sensor# yes
Request Succeeded.
sensor#
```

For More Information

To reset the ASA IPS modules, see the following individual procedures:

- [Reloading, Shutting Down, Resetting, and Recovering the ASA 5500-X IPS SSP, page 18-12](#)
- [Reloading, Shutting Down, Resetting, and Recovering the ASA 5585-X IPS SSP, page 19-12](#)

Displaying Command History

Use the **show history** command to obtain a list of the commands you have entered in the current menu. The maximum number of commands in the list is 50.

To obtain a list of the commands you have used recently, follow these steps:

Step 1 Log in to the CLI.

Step 2 Show the history of the commands you have used in EXEC mode, for example.

```
sensor# show history
clear line
configure terminal
show history
```

Step 3 Show the history of the commands you have used in network access mode, for example.

```
sensor# configure terminal
sensor (config)# service network-access
sensor (config-net)# show history
show settings
show settings terse
show settings | include profile-name|ip-address
exit
show history
sensor (config-net)#
```

Displaying Hardware Inventory

Use the **show inventory** command to display PEP information. This command displays the UDI information that consists of the PID, the VID, and the SN of your sensor. If your sensor supports SFP/SFP+ modules and Regex accelerator cards, they are also displayed. PEP information provides an easy way to obtain the hardware version and serial number through the CLI.

To display PEP information, follow these steps:

Step 1 Log in to the CLI.

Step 2 Display the PEP information for the 4500 series sensor. You can use this information when dealing with TAC.

```
ips-4520# show inventory
Slot No: 1
Name: "Module", DESCR: "IPS 4510- 6 Gig E,4 10 Gig E SFP+"
PID: IPS4510 , VID: V01 , SN: JAF1525BDHN

Name: "RegexAccelerator/0", DESCR: "LCPX5110 (LCPX5110)"
```

```
PID: LCPX5110 , VID: 33554537, SN: L000584510

Name: "TenGigabitEthernet0/0", DESCR: "10G Based-SR"
PID: SFP-10G-SR , VID: V03 , SN: FNS160707RR

Name: "TenGigabitEthernet0/1", DESCR: "10G Based-SR"
PID: SFP-10G-SR , VID: V03 , SN: SPC16260916
ips-4520#
```

Step 3 Display the PEP information for the 4300 series sensors.

```
ips-4360# show inventory
Name: "Chassis", DESCR: "IPS 4360 with SW, 8 GE Data + 1 GE Mgmt"
PID: IPS-4360 , VID: V01 , SN: FGL162740J6

Name: "RegexAccelerator/0", DESCR: "LCPX8640 (humphrey)"
PID: FCH162077NK , VID: 33554537, SN: LXXXXXXYYY

Name: "power supply 1", DESCR: "IPS4360 AC Power Supply "
PID: IPS-4360-PWR-AC , VID: 0700A, SN: 25Y1Y8

Name: "power supply 2", DESCR: "IPS4360 AC Power Supply "
PID: IPS-4360-PWR-AC , VID: 0700A, SN: 25Y1Y9
ips-4360#

ips-4345# show inventory
Name: "power supply 1", DESCR: "IPS-4345-K9 AC Power Supply "
PID: IPS-4345-PWR-AC , VID: A1, SN: 000783
ips-4345#
```

Tracing the Route of an IP Packet



Caution

There is no command interrupt available for this command. It must run to completion.

Use the `trace ip_address count` command to display the route an IP packet takes to a destination. The `ip_address` option is the address of the system to trace the route to. The `count` option lets you define how many hops you want to take. The default is 4. The valid values are 1 to 256.

To trace the route of an IP packet, follow these steps:

Step 1 Log in to the CLI.

Step 2 Display the route of IP packet you are interested in.

```
sensor# trace 10.0.0.1
traceroute to 10.0.0.1 (10.0.0.1), 4 hops max, 40 byte packets
 1 192.0.2.2 (192.0.2.2) 0.267 ms 0.262 ms 0.236 ms
 2 192.0.2.12 (192.0.2.12) 0.24 ms * 0.399 ms
 3 * 192.0.2.12 (192.0.2.12) 0.424 ms *
 4 192.0.2.12 (192.0.2.12) 0.408 ms * 0.406 ms
sensor#
```

Step 3 To configure the route to take more hops than the default of 4, use the `count` option.

```
sensor# trace 10.0.0.1 8
traceroute to 10.0.0.1 (10.0.0.1), 8 hops max, 40 byte packets
```



```

1 192.0.2.2 (192.0.2.2) 0.35 ms 0.261 ms 0.238 ms
2 192.0.2.12 (192.0.2.12) 0.36 ms * 0.344 ms
3 * 192.0.2.12 (192.0.2.12) 0.465 ms *
4 192.0.2.12 (192.0.2.12) 0.319 ms * 0.442 ms
5 * 192.0.2.12 (192.0.2.12) 0.304 ms *
6 192.0.2.12 (192.0.2.12) 0.527 ms * 0.402 ms
7 * 192.0.2.12 (192.0.2.12) 0.39 ms *
8 192.0.2.12 (192.0.2.12) 0.37 ms * 0.486 ms
sensor#

```

Displaying Submode Settings

Use the **show settings [terse]** command in any submode to view the contents of the current configuration.

To display the current configuration settings for a submode, follow these steps:

- Step 1** Log in to the CLI.
- Step 2** Show the current configuration for ARC submode.

```

sensor# configure terminal
sensor (config)# service network-access
sensor (config-net)# show settings
  general
  -----
  log-all-block-events-and-errors: true <defaulted>
  enable-nvram-write: false <defaulted>
  enable-acl-logging: false <defaulted>
  allow-sensor-block: false <defaulted>
  block-enable: true <defaulted>
  block-max-entries: 250 <defaulted>
  max-interfaces: 250 default: 250
  master-blocking-sensors (min: 0, max: 100, current: 0)
  -----
  never-block-hosts (min: 0, max: 250, current: 0)
  -----
  never-block-networks (min: 0, max: 250, current: 0)
  -----
  block-hosts (min: 0, max: 250, current: 0)
  -----
  block-networks (min: 0, max: 250, current: 0)
  -----
  user-profiles (min: 0, max: 250, current: 11)
  -----
  profile-name: 2admin
  -----
  enable-password: <hidden>
  password: <hidden>
  username: pix default:
  -----
  profile-name: r7200

```

```

-----
enable-password: <hidden>
password: <hidden>
username: netrangr default:
-----
profile-name: insidePix
-----
enable-password: <hidden>
password: <hidden>
username: <defaulted>
-----
profile-name: qatest
-----
enable-password: <hidden>
password: <hidden>
username: <defaulted>
-----
profile-name: fwsm
-----
enable-password: <hidden>
password: <hidden>
username: pix default:
-----
profile-name: outsidePix
-----
enable-password: <hidden>
password: <hidden>
username: pix default:
-----
profile-name: cat
-----
enable-password: <hidden>
password: <hidden>
username: <defaulted>
-----
profile-name: rcat
-----
enable-password: <hidden>
password: <hidden>
username: cisco default:
-----
profile-name: nopass
-----
enable-password: <hidden>
password: <hidden>
username: <defaulted>
-----
profile-name: test
-----
enable-password: <hidden>
password: <hidden>
username: pix default:
-----
profile-name: sshswitch
-----
enable-password: <hidden>
password: <hidden>
username: cisco default:
-----
cat6k-devices (min: 0, max: 250, current: 1)
-----
ip-address: 192.0.2.12
-----

```

```

communication: telnet default: ssh-3des
nat-address: 0.0.0.0 <defaulted>
profile-name: cat
block-vlans (min: 0, max: 100, current: 1)
-----
vlan: 1
-----
pre-vacl-name: <defaulted>
post-vacl-name: <defaulted>
-----
-----
router-devices (min: 0, max: 250, current: 1)
-----
ip-address: 192.0.2.25
-----
communication: telnet default: ssh-3des
nat-address: 0.0.0.0 <defaulted>
profile-name: r7200
block-interfaces (min: 0, max: 100, current: 1)
-----
interface-name: fa0/0
direction: in
-----
pre-acl-name: <defaulted>
post-acl-name: <defaulted>
-----
-----
firewall-devices (min: 0, max: 250, current: 2)
-----
ip-address: 192.0.2.30
-----
communication: telnet default: ssh-3des
nat-address: 0.0.0.0 <defaulted>
profile-name: insidePix
-----
ip-address: 192.0.2.3
-----
communication: ssh-3des <defaulted>
nat-address: 0.0.0.0 <defaulted>
profile-name: f1
-----
-----
sensor (config-net)#

```

Step 3 Show the ARC settings in terse mode.

```

sensor(config-net)# show settings terse
general
-----
log-all-block-events-and-errors: true <defaulted>
enable-nvram-write: false <defaulted>
enable-acl-logging: false <defaulted>
allow-sensor-block: false <defaulted>
block-enable: true <defaulted>
block-max-entries: 250 <defaulted>
max-interfaces: 250 default: 250
master-blocking-sensors (min: 0, max: 100, current: 0)
-----
never-block-hosts (min: 0, max: 250, current: 0)

```

```

-----
-----
never-block-networks (min: 0, max: 250, current: 0)
-----
-----
block-hosts (min: 0, max: 250, current: 0)
-----
-----
block-networks (min: 0, max: 250, current: 0)
-----
-----
user-profiles (min: 0, max: 250, current: 11)
-----
profile-name: 2admin
profile-name: r7200
profile-name: insidePix
profile-name: qatest
profile-name: fwsm
profile-name: outsidePix
profile-name: cat
profile-name: rcat
profile-name: nopass
profile-name: test
profile-name: sshswitch
-----
cat6k-devices (min: 0, max: 250, current: 1)
-----
ip-address: 192.0.2.12
-----
router-devices (min: 0, max: 250, current: 1)
-----
ip-address: 192.0.2.25
-----
firewall-devices (min: 0, max: 250, current: 2)
-----
ip-address: 192.0.2.30
ip-address: 192.0.2.3
-----
sensor(config-net)#

```

Step 4 You can use the **include** keyword to show settings in a filtered output, for example, to show only profile names and IP addresses in the ARC configuration.

```

sensor(config-net)# show settings | include profile-name|ip-address
profile-name: 2admin
profile-name: r7200
profile-name: insidePix
profile-name: qatest
profile-name: fwsm
profile-name: outsidePix
profile-name: cat
profile-name: rcat
profile-name: nopass
profile-name: test
profile-name: sshswitch
ip-address: 192.0.2.12
  profile-name: cat
ip-address: 192.0.2.25
  profile-name: r7200
ip-address: 192.0.2.30
  profile-name: insidePix
ip-address: 192.0.2.3
  profile-name: test

```

```
sensor(config-net)#
```



Configuring the ASA 5500-X IPS SSP

This chapter contains procedures that are specific to configuring the ASA 5500-X IPS SSP. It contains the following sections:

- [Notes and Caveats for ASA 5500-X IPS SSP, page 18-1](#)
- [Configuration Sequence for the ASA 5500-X IPS SSP, page 18-2](#)
- [Verifying Initialization for the ASA 5500-X IPS SSP, page 18-3](#)
- [Creating Virtual Sensors for the ASA 5500-X IPS SSP, page 18-3](#)
- [The ASA 5500-X IPS SSP and Bypass Mode, page 18-9](#)
- [The ASA 5500-X IPS SSP and the Normalizer Engine, page 18-9](#)
- [The ASA 5500-X IPS SSP and Memory Usage, page 18-11](#)
- [The ASA 5500-X IPS SSP and Jumbo Packets, page 18-10](#)
- [TCP Reset Differences Between IPS Appliances and the ASA 5500-X IPS SSP, page 18-11](#)
- [Reloading IPS Messages, page 18-11](#)
- [Reloading, Shutting Down, Resetting, and Recovering the ASA 5500-X IPS SSP, page 18-12](#)
- [Health and Status Information, page 18-13](#)
- [ASA 5500-X IPS SSP Failover Scenarios, page 18-20](#)
- [New and Modified Commands, page 18-21](#)

Notes and Caveats for ASA 5500-X IPS SSP

The following notes and caveats apply to configuring the ASA 5500-X IPS SSP:

- The ASA 5500-X IPS SSP is supported in ASA 8.6.1 and later.
- For the ASA 5500-X IPS SSP, normalization is performed by the adaptive security appliance and not the IPS.
- The ASA 5500-X IPS SSP does not support the inline TCP session tracking mode.
- The ASA 5500-X IPS SSP does not support CDP mode.
- Anomaly detection is disabled by default.
- All IPS platforms allow ten concurrent CLI sessions.

- The ASA 5500-X IPS SSP does not support bypass mode. The adaptive security appliance will either fail open, fail close, or fail over depending on the configuration of the adaptive security appliance and the type of activity being done on the IPS.
- The ASA 5500-X IPS SSP (except the ASA 5512-X IPS SSP and the ASA 5515-X IPS SSP) supports the String ICMP XL, String TCP XL, and String UDP XL engines. These engines provide optimized operation for these platforms.

Configuration Sequence for the ASA 5500-X IPS SSP

Perform the following tasks to configure the ASA 5500-X IPS SSP:

1. Obtain and install the current IPS software if your software is not up to date.
2. Obtain and install the license key.
3. Log (session) in to the ASA 5500-X IPS SSP.
4. Run the **setup** command to initialize the ASA 5500-X IPS SSP.
5. Verify initialization for the ASA 5500-X IPS SSP.
6. Configure the adaptive security appliance to send IPS traffic to the ASA 5500-X IPS SSP.
7. Perform other initial tasks, such as adding users, trusted hosts, and so forth.
8. Configure intrusion prevention.
9. Configure global correlation.
10. Perform miscellaneous tasks to keep your ASA 5500-X IPS SSP running smoothly.
11. Upgrade the IPS software with new signature updates and service packs as they become available.
12. Reimage the ASA 5500-X IPS SSP when needed.

For More Information

- For the procedure for logging in to the ASA 5500-X IPS SSP, see [Chapter 2, “Logging In to the Sensor.”](#)
- For the procedure for running the **setup** command, see [Advanced Setup for the ASA 5500-X IPS SSP, page 3-14.](#)
- For the procedure for verifying initialization for the ASA 5500-X IPS SSP, see [Verifying Initialization for the ASA 5500-X IPS SSP, page 18-3.](#)
- For the procedure for creating virtual sensors, see [Creating Virtual Sensors for the ASA 5500-X IPS SSP, page 18-3.](#)
- For the procedures for setting up the ASA 5500-X IPS SSP, see [Chapter 4, “Setting Up the Sensor.”](#)
- For the procedures for configuring intrusion prevention, see [Chapter 8, “Configuring Event Action Rules,”](#) [Chapter 7, “Defining Signatures,”](#) [Chapter 9, “Configuring Anomaly Detection,”](#) and [Chapter 14, “Configuring Attack Response Controller for Blocking and Rate Limiting.”](#)
- For the procedures for configuring global correlation, see [Chapter 10, “Configuring Global Correlation.”](#)
- For the procedures for keeping your ASA 5500-X IPS SSP running smoothly, see [Chapter 17, “Administrative Tasks for the Sensor.”](#)

- For more information on how to obtain Cisco IPS software, see [Chapter 20, “Obtaining Software.”](#)
- For the procedure for reimaging the ASA 5500-X IPS SSP, see [Installing the System Image for the ASA 5500-X IPS SSP, page 21-22.](#)

Verifying Initialization for the ASA 5500-X IPS SSP

You can use the **show module slot details** command to verify that you have initialized the ASA 5500-X IPS SSP and to verify that you have the correct software version.

To verify initialization, follow these steps:

-
- Step 1** Log in to the adaptive security appliance.
 - Step 2** Obtain the details about the ASA 5500-X IPS SSPs.

```
asa# show module ips details
Getting details from the Service Module, please wait...

Card Type:          ASA 5555-X IPS Security Services Processor
Model:              ASA5555-IPS
Hardware version:   N/A
Serial Number:      FCH151070GW
Firmware version:   N/A
Software version:   7.2(1)E4
MAC Address Range: 503d.e59c.7ca0 to 503d.e59c.7ca0
App. name:          IPS
App. Status:        Up
App. Status Desc:   Normal Operation
App. version:       7.2(1)E4
Data Plane Status:  Up
Status:             Up
License:            IPS Module Enabled perpetual
Mgmt IP addr:       192.0.2.2
Mgmt Network mask: 255.255.255.0
Mgmt Gateway:       192.0.2.254
Mgmt Access List:   0.0.0.0/0
Mgmt web ports:     443
Mgmt TLS enabled:   true
asa#
```

- Step 3** Confirm the information.
-

Creating Virtual Sensors for the ASA 5500-X IPS SSP

This section describes how to create virtual sensors on the ASA 5500-X IPS SSP, and contains the following topics:

- [The ASA 5500-X IPS SSP and Virtualization, page 18-4](#)
- [Virtual Sensor Configuration Sequence for ASA 5500-X IPS SSP, page 18-4](#)
- [Creating Virtual Sensors, page 18-4](#)
- [Assigning Virtual Sensors to Adaptive Security Appliance Contexts, page 18-6](#)

The ASA 5500-X IPS SSP and Virtualization

The ASA 5500-X IPS SSP has one sensing interface, PortChannel 0/0. When you create multiple virtual sensors, you must assign this interface to only one virtual sensor. For the other virtual sensors you do not need to designate an interface.

After you create virtual sensors, you must map them to a security context on the adaptive security appliance using the **allocate-ips** command. You can map many security contexts to many virtual sensors.



Note

The **allocate-ips** command does not apply to single mode. In this mode, the adaptive security appliance accepts any virtual sensor named in a **policy-map** command.

The **allocate-ips** command adds a new entry to the security context database. A warning is issued if the specified virtual sensor does not exist; however, the configuration is allowed. The configuration is checked again when the **service-policy** command is processed. If the virtual sensor is not valid, the **fail-open** policy is enforced.

Virtual Sensor Configuration Sequence for ASA 5500-X IPS SSP

Follow this sequence to create virtual sensors on the ASA 5500-X IPS SSP, and to assign them to adaptive security appliance contexts:

1. Configure up to four virtual sensors.
2. Assign the ASA 5500-X IPS SSP sensing interface (PortChannel 0/0) to one of the virtual sensors.
3. (Optional) Assign virtual sensors to different contexts on the adaptive security appliance.
4. Use MPF to direct traffic to the targeted virtual sensor.

Creating Virtual Sensors



Note

You can create four virtual sensors.

Use the **virtual-sensor name** command in service analysis engine submode to create virtual sensors on the ASA 5500-X IPS SSP. You assign policies (anomaly detection, event action rules, and signature definition) to the virtual sensor. You can use the default policies, **ad0**, **rules0**, or **sig0**, or you can create new policies. Then you assign the sensing interface, PortChannel 0/0 for the ASA 5500-X IPS SSP to one virtual sensor.

The following parameters apply:

- **anomaly-detection**—Specifies the anomaly detection parameters:
 - **anomaly-detection-name name**—Specifies the name of the anomaly detection policy.
 - **operational-mode**—Specifies the anomaly detection mode (**inactive**, **learn**, **detect**).



Note

Anomaly detection is disabled by default. You must enable it to configure or apply an anomaly detection policy. Enabling anomaly detection results in a decrease in performance.

- **description**—Provides a description of the virtual sensor.
- **event-action-rules**—Specifies the name of the event action rules policy.
- **signature-definition**—Specifies the name of the signature definition policy.
- **physical-interfaces**—Specifies the name of the physical interface.
- **no**—Removes an entry or selection.

Creating Virtual Sensors

To create a virtual sensor on the ASA 5500-X IPS SSP, follow these steps:

-
- Step 1** Log in to the CLI using an account with administrator privileges.
- Step 2** Enter service analysis mode.
- ```
sensor# configure terminal
sensor(config)# service analysis-engine
sensor(config-ana)#
```
- Step 3** Add a virtual sensor.
- ```
sensor(config-ana)# virtual-sensor vs1
sensor(config-ana-vir)#
```
- Step 4** Add a description for this virtual sensor.
- ```
sensor(config-ana-vir)# description virtual sensor 1
```
- Step 5** Assign an anomaly detection policy and operational mode to this virtual sensor if you have enabled anomaly detection. If you do not want to use the default anomaly detection policy, ad0, you must create a new one using the **service anomaly-detection name** command, for example, ad1.
- ```
sensor(config-ana-vir)# anomaly-detection
sensor(config-ana-vir-ano)# anomaly-detection-name ad0
sensor(config-ana-vir-ano)# operational-mode learn
```
- Step 6** Assign an event action rules policy to this virtual sensor. If you do not want to use the default event action rules policy, rules0, you must create a new one using the **service event-action-rules name** command, for example, rules1
- ```
sensor(config-ana-vir-ano)# exit
sensor(config-ana-vir)# event-action-rules rules0
```
- Step 7** Assign a signature definition policy to this virtual sensor. If you do not want to use the default signature definition policy, sig0, you must create a new one using the **service signature-definition name** command, for example sig1.
- ```
sensor(config-ana-vir)# signature-definition sig0
```
- Step 8** Assign the interface to one virtual sensor. By default the sensing interface is already assigned to the default virtual sensor, vs0. You must remove it from the default virtual sensor to assign it to another virtual sensor that you create.
- ```
sensor(config-ana-vir)# physical-interface PortChannel10/0
```
- Step 9** Verify the virtual sensor settings.
- ```
sensor(config-ana-vir)# show settings
<protected entry>
name: vs1
-----
description: virtual sensor 1 default:
```

```

signature-definition: sig0 <protected>
event-action-rules: rules0 <protected>
anomaly-detection
-----
anomaly-detection-name: ad0 <protected>
operational-mode: inactive <defaulted>
-----
physical-interface (min: 0, max: 999999999, current: 1)
-----
name: PortChannel0/0
-----
inline-TCP-evasion-protection-mode: strict <defaulted>
-----
sensor(config-ana-vir)#

```

Step 10 Exit analysis engine mode.

```

sensor(config-ana-vir)# exit
sensor(config-ana)# exit
Apply Changes?[yes]:
sensor(config)#

```

Step 11 Press **Enter** to apply the changes or enter **no** to discard them.

For More Information

- For the procedure for enabling anomaly detection, see [Enabling Anomaly Detection, page 9-8](#).
- For the procedures for creating and configuring anomaly detection policies, see [Working With Anomaly Detection Policies, page 9-8](#).
- For the procedure for creating and configuring event action rules policies, see [Working With Event Action Rules Policies, page 8-8](#).
- For the procedure for creating and configuring signature definitions, [Working With Signature Definition Policies, page 7-2](#).

Assigning Virtual Sensors to Adaptive Security Appliance Contexts

After you create virtual sensors on the ASA 5500-X IPS SSP, you must assign the virtual sensors to a security context on the adaptive security appliance.

The following parameters apply:

- **[no] allocate-ips** *sensor_name* [*mapped_name*] [**default**]
—Allocates a virtual sensor to a security context. Supported modes are multiple mode, system context, and context submode.



Note You cannot allocate the same virtual sensor twice in a context.

- *sensor_name*—Specifies the name of the virtual sensor configured on the ASA 5500-X IPS SSP. You receive a warning message if the name is not valid.
- *mapped_name*—Specifies the name by which the security context knows the virtual sensor.



Note The mapped name is used to hide the real name of the virtual sensor from the context, usually done for reasons of security or convenience to make the context configuration more generic. If no mapped name is used, the real virtual sensor name is used. You cannot reuse a mapped name for two different virtual sensors in a context.

- **no**—De-allocates the sensor, looks through the policy map configurations, and deletes any IPS subcommand that refers to it.
- **default**—Specifies this virtual sensor as the default. All legacy IPS configurations that do not specify a virtual sensor are mapped to this virtual sensor.

**Caution**

You can only configure one default virtual sensor per context. You must turn off the default flag of an existing default virtual sensor before you can designate another virtual sensor as the default.

- **clear configure allocate-ips**—Removes the configuration.
- **allocate-ips?**—Displays the list of configured virtual sensors.
- **show context [detail]**—Updated to display information about virtual sensors. In user context mode, a new line is added to show the mapped names of all virtual sensors that have been allocated to this context. In system mode, two new lines are added to show the real and mapped names of virtual sensors allocated to this context.

You can assign multiple virtual sensors to a context. Multiple contexts can share one virtual sensor, and when sharing, the contexts can have different mapped names (aliases) for the same virtual sensor. The following procedure demonstrates how to add three security contexts in multiple mode and how to assign virtual sensors to these security contexts.

Assigning Virtual Sensors to Contexts

To assign virtual sensors to adaptive security appliance contexts in multiple mode for the ASA 5500-X IPS SSP, follow these steps:

Step 1 Log in to the adaptive security appliance.

Step 2 Display the list of available virtual sensors.

```
asa# show ips
Sensor Name      Sensor ID
-----
vs0              1
vs1              2
asa#
```

Step 3 Enter configuration mode.

```
asa# configure terminal
asa(config)#
```

Step 4 Enter multiple mode.

```
asa(config)# mode multiple
WARNING: This command will change the behavior of the device
WARNING: This command will initiate a Reboot
Proceed with change mode? [confirm] yes
asa(config)#
```

Step 5 Add three context modes to multiple mode.

```
asa(config)# admin-context admin
Creating context 'admin'... Done. (13)
asa(config)# context admin
asa(config-ctx)# allocate-interface GigabitEthernet0/0.101
asa(config-ctx)# allocate-interface GigabitEthernet0/1.102
asa(config-ctx)# allocate-interface Management0/0
asa(config-ctx)# config-url disk0:/admin.cfg
Cryptochecksum (changed): 0c34dc67 f413ad74 e297464a db211681
INFO: Context admin was created with URL disk0:/admin.cfg
INFO: Admin context will take some time to come up .... please wait.
asa(config-ctx)#
asa(config-ctx)# context c2
Creating context 'c2'... Done. (14)
asa(config-ctx)# allocate-interface GigabitEthernet0/0.103
asa(config-ctx)# allocate-interface GigabitEthernet0/1.104
asa(config-ctx)# config-url disk0:/c2.cfg

WARNING: Could not fetch the URL disk0:/c2.cfg
INFO: Creating context with default config
asa(config-ctx)#
asa(config-ctx)# context c3
Creating context 'c3'... Done. (15)
asa(config-ctx)# all
asa(config-ctx)# allocate-in
asa(config-ctx)# allocate-interface g0/2
asa(config-ctx)# allocate-interface g0/3
asa(config-ctx)# config-url disk0:/c3.cfg

WARNING: Could not fetch the URL disk0:/c3.cfg
INFO: Creating context with default config
asa(config-ctx)#
```

Step 6 Assign virtual sensors to the security contexts.

```
asa(config)# context admin
asa(config-ctx)# allocate-ips vs0 adminvs0
asa(config-ctx)# exit
asa(config)# context c2
asa(config-ctx)# allocate-ips vs1 c2vs1
asa(config)# context c3
asa(config-ctx)# allocate-ips vs0 c3vs0
asa(config-ctx)# allocate-ips vs1 c3vs1
asa(config-ctx)#
```

Step 7 Configure MPF for each context.

Note The following example shows context 3 (c3).

```
asa(config)# context c3
asa/c3(config)# class-map any
asa/c3(config-cmap)# match access-list any
asa/c3(config-cmap)# exit
asa/c3(config)# policy-map ips_out
asa/c3(config-pmap)# class any
asa/c3(config-pmap-c)# ips promiscuous fail-close sensor c3vs1
asa/c3(config-pmap-c)# policy-map ips_in
asa/c3(config-pmap)# class any
asa/c3(config-pmap-c)# ips inline fail-open sensor c3vs0
asa/c3(config-pmap-c)# service-policy ips_out interface outside
asa/c3(config)# service-policy ips_in interface inside
```

```
asa/c3(config)#
```

Step 8 Confirm the configuration.

```
asa/c3(config)# exit
asa(config)# show ips detail
Sensor Name      Sensor ID      Allocated To    Mapped Name
-----
vs0              1             admin           adminvs0
                 c3            c3vs0
vs1              2             c2              c2vs1
                 c3            c3vs1
asa(config)#
```

The ASA 5500-X IPS SSP and Bypass Mode

The ASA 5500-X IPS SSP does not support bypass mode. The adaptive security appliance will either fail open, fail close, or fail over depending on the configuration of the adaptive security appliance and the type of activity being done on the ASA 5500-X IPS SSP.

The SensorApp Fails

The following occurs when the SensorApp fails:

- If the adaptive security appliance is configured for failover, then the adaptive security appliance fails over.
- If the adaptive security appliance is not configured for failover or failover is not possible:
 - If set to fail-open, the adaptive security appliance passes traffic without sending it to the ASA IPS module.
 - If set to fail-close, the adaptive security appliance stops passing traffic until the ASA IPS module is restarted.

The SensorApp is Reconfigured

The following occurs when the SensorApp is reconfigured:

- If set to fail-open, the adaptive security appliance passes traffic without sending it to the ASA IPS module.
- If set to fail-close, the adaptive security appliance stops passing traffic until the ASA IPS module is restarted.



Note

The adaptive security appliance does not fail over unless the reconfiguration is not completed.

The ASA 5500-X IPS SSP and the Normalizer Engine

The majority of the features in the Normalizer engine are not used on the ASA 5500-X IPS SSP, because the ASA itself handles the normalization. Packets on the ASA IPS modules go through a special path in the Normalizer that only reassembles fragments and puts packets in the right order for the TCP stream. The Normalizer does not do any of the normalization that is done on an inline IPS appliance, because that causes problems in the way the ASA handles the packets.

The following Normalizer engine signatures are not supported:

- 1300.0
- 1304.0
- 1305.0
- 1307.0
- 1308.0
- 1309.0
- 1311.0
- 1315.0
- 1316.0
- 1317.0
- 1330.0
- 1330.1
- 1330.2
- 1330.9
- 1330.10
- 1330.12
- 1330.14
- 1330.15
- 1330.16
- 1330.17
- 1330.18

For More Information

For detailed information about the Normalizer engine, see [Normalizer Engine, page B-36](#).

The ASA 5500-X IPS SSP and Jumbo Packets

The jumbo packet count in the **show interface** command output from the lines `Total Jumbo Packets Received` and `Total Jumbo Packets Transmitted` for ASA IPS modules may be larger than expected due to some packets that were almost jumbo size on the wire being counted as jumbo size by the IPS. This miscount is a result of header bytes added to the packet by the ASA before the packet is transmitted to the IPS. For IPv4, 58 bytes of header data are added. For IPv6, 78 bytes of header data are added. The ASA removes the added IPS header before the packet leaves the ASA.

The ASA 5500-X IPS SSP and Memory Usage

For the ASA 5500-X IPS SSP, the memory usage is 93%. The default health thresholds for the sensor are 80% for yellow and 91% for red, so the sensor health will be shown as red on these platforms even for normal operating conditions. You can tune the threshold percentage for memory usage so that it reads more accurately for these platforms by configuring the **memory-usage-policy** option in the sensor health metrics.



Note

Make sure you have the **memory-usage-policy** option in the sensor health metrics enabled.

Table 18-1 lists the yellow-threshold and the red-threshold health values.

Table 18-1 ASA 5500-X IPS SSP Memory Usage Values

Platform	Yellow	Red	Memory Used
ASA 5512-X IPS SSP	85%	91%	28%
ASA 5515-X IPS SSP	88%	92%	14%
ASA 5525-X IPS SSP	88%	92%	14%
ASA 5545-X IPS SSP	93%	96%	13%
ASA 5555-X IPS SSP	95%	98%	17%

TCP Reset Differences Between IPS Appliances and the ASA 5500-X IPS SSP

The IPS appliance sends TCP reset packets to both the attacker and victim when `reset-tcp-connection` is selected. The IPS appliance sends a TCP reset packet only to the victim under the following circumstances:

- When a `deny-packet-inline` or `deny-connection-inline` is selected
- When TCP-based signatures and `reset-tcp-connection` have NOT been selected

In the case of the ASA 5500-X IPS SSP, the TCP reset request is sent to the ASA, and then the ASA sends the TCP reset packets. The ASA sends TCP reset packets to both the attacker and victim when the `reset-tcp-connection` is selected. When `deny-packet-inline` or `deny-connection-inline` is selected, the ASA sends the TCP reset packet to either the attacker or victim depending on the configuration of the signature. Signatures configured to swap the attacker and victim when reporting the alert can cause the ASA to send the TCP reset packet to the attacker.

Reloading IPS Messages

The following messages generated during some IPS signature and global correlation updates for IPS 7.1 and later on the ASA 5500-X IPS SSP can cause confusion since the IPS is not reloading:

```
ASA5585-SSP-IPS20 Module in slot 1, application up "IPS", version "7.1(1)E4" Normal
Operation
ASA5585-SSP-IPS20 Module in slot 1, application reloading "IPS", version "7.1(1)E4" Config
Change
```

These messages are generated during some, but not all, of the global correlation updates that are attempted every five minutes. This is expected behavior. There is a global correlation check every five minutes, but there may not be an update available, thus the message appears every hour or so. When a global correlation update actually takes place, a message is sent from the IPS to the ASA indicating that a configuration change is taking place.

Reloading, Shutting Down, Resetting, and Recovering the ASA 5500-X IPS SSP



Note

You can enter the **sw-module** commands from privileged EXEC mode or from global configuration mode. You can enter the commands in single routed mode and single transparent mode. For adaptive security appliances operating in multi-mode (routed or transparent multi-mode) you can only execute the **sw-module** commands from the system context (not from administrator or user contexts).

Use the following commands to reload, shut down, reset, recover the password, and recover the ASA 5500-X IPS SSP directly from the adaptive security appliance:

- **sw-module module ips reload**—This command reloads the software on the ASA 5500-X IPS SSP without doing a hardware reset. It is effective only when the module is in the Up state.
- **sw-module module ips shutdown**—This command shuts down the software on the ASA 5500-X IPS SSP. It is effective only when the module is in Up state.
- **sw-module module ips reset**—This command performs a hardware reset of the ASA 5500-X IPS SSP. It is applicable when the module is in the Up/Down/Unresponsive/Recover states.
- **sw-module module ips password-reset**—This command restores the cisco CLI account password on the ASA 5500-X IPS SSP to the default **cisco**.
- **sw-module module ips recover image disk0:/image name**—This command starts the reimage process by setting the image location and name. You must first copy the IPS image to the ASA to disk0:/.
- **sw-module module ips recover boot**—This command reimages the ASA 5500-X IPS SSP. It is applicable only when the module is in the Up state.
- **sw-module module ips recover stop**—This command stops the reimage of the ASA 5500-X IPS SSP. It is applicable only when the module is in the Recover state.



Caution

If the ASA 5500-X IPS SSP recovery needs to be stopped, you must issue the **sw-module module ips recover stop** command within 30 to 45 seconds after starting the recovery. Waiting any longer can lead to unexpected consequences. For example, the module may come up in the Unresponsive state.

- **sw-module module ips recover configure**—Use this command to configure parameters for the ASA 5500-X IPS SSP recovery. The essential parameters are the IP address and recovery image TFTP URL location.

Example

```
asa-ips# sw-module module ips recover configure image
disk0://IPS-SSP_5555-K9-sys-1.1-a-7.2-1-E4.aip
Image URL [tftp://192.0.2.1/IPS-5545-K9-sys-1.1-a-7.2-1-E4.aip]:
```

```

Port IP Address [192.0.2.226]:
VLAN ID [0]:
Gateway IP Address [192.0.2.254]:

```

For More Information

For the procedure for recovering the ASA 5500-X IPS SSP system image, see [Installing the System Image for the ASA 5500-X IPS SSP, page 21-22](#).

Health and Status Information

To see the general health of the ASA 5500-X IPS SSP, use the **show module ips details** command.

```

asa# show module ips details
Getting details from the Service Module, please wait...

Card Type:           IPS 5555 Intrusion Prevention System
Model:               IPS5555
Hardware version:    N/A
Serial Number:       FCH1504V0CW
Firmware version:    N/A
Software version:    7.2(1)E4
MAC Address Range:   503d.e59c.7ca0 to 503d.e59c.7ca0
App. name:           IPS
App. Status:         Up
App. Status Desc:    Normal Operation
App. version:        7.2(1)E4
Data Plane Status:   Up
Status:              Up
License:             IPS Module Enabled perpetual
Mgmt IP addr:        192.168.1.2
Mgmt Network mask:   255.255.255.0
Mgmt Gateway:        192.168.1.1
Mgmt web ports:      443
Mgmt TLS enabled:    true
asa#

```

The output shows that the ASA 5500-X IPS SSP is up. If the status reads `Down`, you can reset it using the **sw-module module 1 reset** command.

If you have problems with reimaging the ASA 5500-X IPS SSP, use the **debug module-boot** command to see the output as it boots. Make sure you have the correct IP address for the TFTP server and you have the correct file on the TFTP server. Then use the **sw-module module ips recover** command again to reimage the module.

```

asa-ips# sw-module module ips recover configure image
disk0:/IPS-SSP_5555-K9-sys-1.1-a-7.2-1-E4.aip
Image URL [tftp://192.0.2.1/IPS-5545-K9-sys-1.1-a-7.2-1-E4.aip]:
Port IP Address [192.0.2.226]:
VLAN ID [0]:
Gateway IP Address [192.0.2.254]:

asa-ips# debug module-boot
debug module-boot enabled at level 1
asa-ips# sw-module module ips reload

Reload module ips? [confirm]
Reload issued for module ips.
asa-ips# Mod-ips 228> ***
Mod-ips 229> *** EVENT: The module is reloading.
Mod-ips 230> *** TIME: 08:07:36 CST Jan 17 2012

```

```

Mod-ips 231> ***
Mod-ips 232> Mod-ips 233> The system is going down NOW!
Mod-ips 234> Sending SIGTERM to all processes
Mod-ips 235> Sending SIGKILL to all processes
Mod-ips 236> Requesting system reboot
Mod-ips 237> e1000 0000:00:07.0: PCI INT A disabled
Mod-ips 238> e1000 0000:00:06.0: PCI INT A disabled
Mod-ips 239> e1000 0000:00:05.0: PCI INT A disabled
Mod-ips 240> Restarting system.
Mod-ips 241> machine restart
Mod-ips 242> IVSHMEM: addr = 4093640704 size = 67108864
Mod-ips 243> Booting 'Cisco IPS'
Mod-ips 244> root (hd0,0)
Mod-ips 245> Filesystem type is ext2fs, partition type 0x83
Mod-ips 246> kernel /ips-2.6.1d ro initfsDev=/dev/hda1 init=loader.run rootrw=/dev/hda2
init
Mod-ips 247> fs=runtime-image.cpio.bz2 hda=nodma console=ttyS0 plat=saleen htlblow=1
hugepag
Mod-ips 248> es=3223
Mod-ips 249> [Linux-bzImage, setup=0x2c00, size=0x2bad80]
Mod-ips 250> Linux version 2.6.29.1 (ipsbuild@seti-teambuilder-a) (gcc version 4.3.2
(crosstool
Mod-ips 251> -NG-1.4.1) ) #56 SMP Tue Dec 6 00:46:11 CST 2011
Mod-ips 252> Command line: ro initfsDev=/dev/hda1 init=loader.run rootrw=/dev/hda2
initfs=runti
Mod-ips 253> me-image.cpio.bz2 hda=nodma console=ttyS0 plat=saleen htlblow=1
hugepages=3223
Mod-ips 254> KERNEL supported cpus:
Mod-ips 255> Intel GenuineIntel
Mod-ips 256> AMD AuthenticAMD
Mod-ips 257> Centaur CentaurHauls
Mod-ips 258> BIOS-provided physical RAM map:
Mod-ips 259> BIOS-e820: 0000000000000000 - 000000000009f400 (usable)
Mod-ips 260> BIOS-e820: 000000000009f400 - 00000000000a0000 (reserved)
Mod-ips 261> BIOS-e820: 00000000000f0000 - 0000000000100000 (reserved)
Mod-ips 262> BIOS-e820: 0000000000100000 - 00000000dffffd00 (usable)
Mod-ips 263> BIOS-e820: 00000000dffffd00 - 00000000e0000000 (reserved)
Mod-ips 264> BIOS-e820: 00000000fffb0000 - 0000000100000000 (reserved)
Mod-ips 265> BIOS-e820: 0000000100000000 - 0000000201400000 (usable)
Mod-ips 266> DMI 2.4 present.
Mod-ips 267> last_pfn = 0x201400 max_arch_pfn = 0x100000000
Mod-ips 268> last_pfn = 0xdffffd max_arch_pfn = 0x100000000
Mod-ips 269> init_memory_mapping: 0000000000000000-00000000dffffd00
Mod-ips 270> last_map_addr: dffffd00 end: dffffd00
Mod-ips 271> init_memory_mapping: 0000000100000000-0000000201400000
Mod-ips 272> last_map_addr: 201400000 end: 201400000
Mod-ips 273> ACPI: RSDP 000F88D0, 0014 (r0 BOCHS )
Mod-ips 274> ACPI: RSDT DFFFDD00, 0034 (r1 BOCHS BXPCRSDT 1 BXPC 1)
Mod-ips 275> ACPI: FACP DFFFDD90, 0074 (r1 BOCHS BXPCFACP 1 BXPC 1)
Mod-ips 276> FADT: X_PM1a_EVT_BLK.bit_width (16) does not match PM1_EVT_LEN (4)
Mod-ips 277> ACPI: DSDT DFFFDF10, 1E22 (r1 BXPC BXDSDT 1 INTL 20090123)
Mod-ips 278> ACPI: FACS DFFFDD40, 0040
Mod-ips 279> ACPI: SSDT DFFFDE90, 0079 (r1 BOCHS BXPCSSDT 1 BXPC 1)
Mod-ips 280> ACPI: APIC DFFFDD80, 0090 (r1 BOCHS BXPCAPIC 1 BXPC 1)
Mod-ips 281> ACPI: HPET DFFFDD40, 0038 (r1 BOCHS BXPCHPET 1 BXPC 1)
Mod-ips 282> No NUMA configuration found
Mod-ips 283> Faking a node at 0000000000000000-0000000201400000
Mod-ips 284> Bootmem setup node 0 0000000000000000-0000000201400000
Mod-ips 285> NODE_DATA [0000000000011000 - 000000000001ffff]
Mod-ips 286> bootmap [0000000000020000 - 000000000006027f] pages 41
Mod-ips 287> (6 early reservations) ==> bootmem [0000000000 - 0201400000]
Mod-ips 288> #0 [000000000000 - 000000010000] BIOS data page ==> [000000000000 - 000000010000]
Mod-ips 289> #1 [000000060000 - 000000080000] TRAMPOLINE ==> [000000060000 - 000000080000]
Mod-ips 290> #2 [0000200000 - 0000d55754] TEXT DATA BSS ==> [0000200000 - 0000d55754]

```

```

Mod-ips 291> #3 [000009f400 - 0000100000] BIOS reserved ==> [000009f400 - 0000100000]
Mod-ips 292> #4 [0000008000 - 000000c000] PGTABLE ==> [0000008000 - 000000c000]
Mod-ips 293> #5 [000000c000 - 0000011000] PGTABLE ==> [000000c000 - 0000011000]
Mod-ips 294> found SMP MP-table at [ffff880000f8920] 000f8920
Mod-ips 295> Zone PFN ranges:
Mod-ips 296> DMA 0x00000000 -> 0x000001000
Mod-ips 297> DMA32 0x00001000 -> 0x001000000
Mod-ips 298> Normal 0x00100000 -> 0x00201400
Mod-ips 299> Movable zone start PFN for each node
Mod-ips 300> early_node_map[3] active PFN ranges
Mod-ips 301> 0: 0x00000000 -> 0x00000009f
Mod-ips 302> 0: 0x000000100 -> 0x000dffffd
Mod-ips 303> 0: 0x00100000 -> 0x00201400
Mod-ips 304> ACPI: PM-Timer IO Port: 0xb008
Mod-ips 305> ACPI: LAPIC (acpi_id[0x00] lapic_id[0x00] enabled)
Mod-ips 306> ACPI: LAPIC (acpi_id[0x01] lapic_id[0x01] enabled)
Mod-ips 307> ACPI: LAPIC (acpi_id[0x02] lapic_id[0x02] enabled)
Mod-ips 308> ACPI: LAPIC (acpi_id[0x03] lapic_id[0x03] enabled)
Mod-ips 309> ACPI: LAPIC (acpi_id[0x04] lapic_id[0x04] enabled)
Mod-ips 310> ACPI: LAPIC (acpi_id[0x05] lapic_id[0x05] enabled)
Mod-ips 311> ACPI: IOAPIC (id[0x06] address[0xfec00000] gsi_base[0])
Mod-ips 312> IOAPIC[0]: apic_id 6, version 0, address 0xfec00000, GSI 0-23
Mod-ips 313> ACPI: INT_SRC_OVR (bus 0 bus_irq 5 global_irq 5 high level)
Mod-ips 314> ACPI: INT_SRC_OVR (bus 0 bus_irq 9 global_irq 9 high level)
Mod-ips 315> ACPI: INT_SRC_OVR (bus 0 bus_irq 10 global_irq 10 high level)
Mod-ips 316> ACPI: INT_SRC_OVR (bus 0 bus_irq 11 global_irq 11 high level)
Mod-ips 317> Using ACPI (MADT) for SMP configuration information
Mod-ips 318> ACPI: HPET id: 0x8086a201 base: 0xfed00000
Mod-ips 319> SMP: Allowing 6 CPUs, 0 hotplug CPUs
Mod-ips 320> Allocating PCI resources starting at e2000000 (gap: e0000000:1ffbc000)
Mod-ips 321> NR_CPUS:32 nr_cpumask_bits:32 nr_cpu_ids:6 nr_node_ids:1
Mod-ips 322> PERCPU: Allocating 49152 bytes of per cpu data
Mod-ips 323> Built 1 zonelists in Zone order, mobility grouping on. Total pages: 1939347
Mod-ips 324> Policy zone: Normal
Mod-ips 325> Kernel command line: ro initfsDev=/dev/hda1 init=loader.run rootrw=/dev/hda2
initf
Mod-ips 326> s=runtime-image.cpio.bz2 hda=nodma console=ttyS0 plat=saleen htlblow=1
hugepages=3
Mod-ips 327> 223
Mod-ips 328> hugetlb_lowmem_setup: Allocated 2097152 huge pages (size=0x200000) from
lowmem are
Mod-ips 329> a at 0xffff88002ee00000 phys addr 0x000000002ee00000
Mod-ips 330> Initializing CPU#0
Mod-ips 331> PID hash table entries: 4096 (order: 12, 32768 bytes)
Mod-ips 332> Fast TSC calibration using PIT
Mod-ips 333> Detected 2792.965 MHz processor.
Mod-ips 334> Console: colour VGA+ 80x25
Mod-ips 335> console [ttyS0] enabled
Mod-ips 336> Checking aperture...
Mod-ips 337> No AGP bridge found
Mod-ips 338> PCI-DMA: Using software bounce buffering for IO (SWIOTLB)
Mod-ips 339> Placing 64MB software IO TLB between ffff880020000000 - ffff880024000000
Mod-ips 340> software IO TLB at phys 0x20000000 - 0x24000000
Mod-ips 341> Memory: 7693472k/8409088k available (3164k kernel code, 524688k absent,
190928k re
Mod-ips 342> served, 1511k data, 1032k init)
Mod-ips 343> Calibrating delay loop (skipped), value calculated using timer frequency..
5585.93
Mod-ips 344> Bogomips (lpj=2792965)
Mod-ips 345> Dentry cache hash table entries: 1048576 (order: 11, 8388608 bytes)
Mod-ips 346> Inode-cache hash table entries: 524288 (order: 10, 4194304 bytes)
Mod-ips 347> Mount-cache hash table entries: 256
Mod-ips 348> CPU: L1 I cache: 32K, L1 D cache: 32K
Mod-ips 349> CPU: L2 cache: 4096K

```

```

Mod-ips 350> CPU 0/0x0 -> Node 0
Mod-ips 351> Freeing SMP alternatives: 29k freed
Mod-ips 352> ACPI: Core revision 20081204
Mod-ips 353> Setting APIC routing to flat
Mod-ips 354> ..TIMER: vector=0x30 apic1=0 pin1=0 apic2=-1 pin2=-1
Mod-ips 355> CPU0: Intel QEMU Virtual CPU version 0.12.5 stepping 03
Mod-ips 356> Booting processor 1 APIC 0x1 ip 0x6000
Mod-ips 357> Initializing CPU#1
Mod-ips 358> Calibrating delay using timer specific routine.. 5585.16 BogoMIPS
(lpj=2792581)
Mod-ips 359> CPU: L1 I cache: 32K, L1 D cache: 32K
Mod-ips 360> CPU: L2 cache: 4096K
Mod-ips 361> CPU 1/0x1 -> Node 0
Mod-ips 362> CPU1: Intel QEMU Virtual CPU version 0.12.5 stepping 03
Mod-ips 363> checking TSC synchronization [CPU#0 -> CPU#1]:
Mod-ips 364> Measured 1453783140569731 cycles TSC warp between CPUs, turning off TSC
clock.
Mod-ips 365> Marking TSC unstable due to check_tsc_sync_source failed
Mod-ips 366> Booting processor 2 APIC 0x2 ip 0x6000
Mod-ips 367> Initializing CPU#2
Mod-ips 368> Calibrating delay using timer specific routine.. 5580.51 BogoMIPS
(lpj=2790259)
Mod-ips 369> CPU: L1 I cache: 32K, L1 D cache: 32K
Mod-ips 370> CPU: L2 cache: 4096K
Mod-ips 371> CPU 2/0x2 -> Node 0
Mod-ips 372> CPU2: Intel QEMU Virtual CPU version 0.12.5 stepping 03
Mod-ips 373> Booting processor 3 APIC 0x3 ip 0x6000
Mod-ips 374> Initializing CPU#3
Mod-ips 375> Calibrating delay using timer specific routine.. 5585.18 BogoMIPS
(lpj=2792594)
Mod-ips 376> CPU: L1 I cache: 32K, L1 D cache: 32K
Mod-ips 377> CPU: L2 cache: 4096K
Mod-ips 378> CPU 3/0x3 -> Node 0
Mod-ips 379> CPU3: Intel QEMU Virtual CPU version 0.12.5 stepping 03
Mod-ips 380> Booting processor 4 APIC 0x4 ip 0x6000
Mod-ips 381> Initializing CPU#4
Mod-ips 382> Calibrating delay using timer specific routine.. 5585.15 BogoMIPS
(lpj=2792579)
Mod-ips 383> CPU: L1 I cache: 32K, L1 D cache: 32K
Mod-ips 384> CPU: L2 cache: 4096K
Mod-ips 385> CPU 4/0x4 -> Node 0
Mod-ips 386> CPU4: Intel QEMU Virtual CPU version 0.12.5 stepping 03
Mod-ips 387> Booting processor 5 APIC 0x5 ip 0x6000
Mod-ips 388> Initializing CPU#5
Mod-ips 389> Calibrating delay using timer specific routine.. 5585.21 BogoMIPS
(lpj=2792609)
Mod-ips 390> CPU: L1 I cache: 32K, L1 D cache: 32K
Mod-ips 391> CPU: L2 cache: 4096K
Mod-ips 392> CPU 5/0x5 -> Node 0
Mod-ips 393> CPU5: Intel QEMU Virtual CPU version 0.12.5 stepping 03
Mod-ips 394> Brought up 6 CPUs
Mod-ips 395> Total of 6 processors activated (33507.17 BogoMIPS).
Mod-ips 396> net_namespace: 1312 bytes
Mod-ips 397> Booting paravirtualized kernel on bare hardware
Mod-ips 398> NET: Registered protocol family 16
Mod-ips 399> ACPI: bus type pci registered
Mod-ips 400> dca service started, version 1.8
Mod-ips 401> PCI: Using configuration type 1 for base access
Mod-ips 402> mtrr: your CPUs had inconsistent variable MTRR settings
Mod-ips 403> mtrr: your CPUs had inconsistent MTRRdefType settings
Mod-ips 404> mtrr: probably your BIOS does not setup all CPUs.
Mod-ips 405> mtrr: corrected configuration.
Mod-ips 406> bio: create slab <bio-0> at 0
Mod-ips 407> ACPI: Interpreter enabled

```

```
Mod-ips 408> ACPI: (supports S0 S5)
Mod-ips 409> ACPI: Using IOAPIC for interrupt routing
Mod-ips 410> ACPI: No dock devices found.
Mod-ips 411> ACPI: PCI Root Bridge [PCI0] (0000:00)
Mod-ips 412> pci 0000:00:01.3: quirk: region b000-b03f claimed by PIIX4 ACPI
Mod-ips 413> pci 0000:00:01.3: quirk: region b100-b10f claimed by PIIX4 SMB
Mod-ips 414> IVSHMEM: addr = 4093640704 size = 67108864
Mod-ips 415> ACPI: PCI Interrupt Link [LNKA] (IRQs 5 *10 11)
Mod-ips 416> ACPI: PCI Interrupt Link [LNKB] (IRQs 5 *10 11)
Mod-ips 417> ACPI: PCI Interrupt Link [LNKC] (IRQs 5 10 *11)
Mod-ips 418> ACPI: PCI Interrupt Link [LNKD] (IRQs 5 10 *11)
Mod-ips 419> SCSI subsystem initialized
Mod-ips 420> usbcore: registered new interface driver usbfs
Mod-ips 421> usbcore: registered new interface driver hub
Mod-ips 422> usbcore: registered new device driver usb
Mod-ips 423> PCI: Using ACPI for IRQ routing
Mod-ips 424> pnp: PnP ACPI init
Mod-ips 425> ACPI: bus type pnp registered
Mod-ips 426> pnp: PnP ACPI: found 9 devices
Mod-ips 427> ACPI: ACPI bus type pnp unregistered
Mod-ips 428> NET: Registered protocol family 2
Mod-ips 429> IP route cache hash table entries: 262144 (order: 9, 2097152 bytes)
Mod-ips 430> TCP established hash table entries: 524288 (order: 11, 8388608 bytes)
Mod-ips 431> TCP bind hash table entries: 65536 (order: 8, 1048576 bytes)
Mod-ips 432> TCP: Hash tables configured (established 524288 bind 65536)
Mod-ips 433> TCP reno registered
Mod-ips 434> NET: Registered protocol family 1
Mod-ips 435> Adding htlb page ffff88002ee00000 phys 000000002ee00000 page fffffe20000a41000
Mod-ips 436> HugeTLB registered 2 MB page size, pre-allocated 3223 pages
Mod-ips 437> report_hugepages: Using 1 pages from low memory at ffff88002ee00000 HugeTLB
FS
Mod-ips 438> msgmni has been set to 15026
Mod-ips 439> alg: No test for stdrng (krng)
Mod-ips 440> io scheduler noop registered
Mod-ips 441> io scheduler anticipatory registered
Mod-ips 442> io scheduler deadline registered
Mod-ips 443> io scheduler cfq registered (default)
Mod-ips 444> pci 0000:00:00.0: Limiting direct PCI/PCI transfers
Mod-ips 445> pci 0000:00:01.0: PIIX3: Enabling Passive Release
Mod-ips 446> pci 0000:00:01.0: Activating ISA DMA hang workarounds
Mod-ips 447> pci_hotplug: PCI Hot Plug PCI Core version: 0.5
Mod-ips 448> pciehp: PCI Express Hot Plug Controller Driver version: 0.4
Mod-ips 449> acpiphp: ACPI Hot Plug PCI Controller Driver version: 0.5
Mod-ips 450> acpiphp_glue: can't get bus number, assuming 0
Mod-ips 451> decode_hpp: Could not get hotplug parameters. Use defaults
Mod-ips 452> acpiphp: Slot [1] registered
Mod-ips 453> acpiphp: Slot [2] registered
Mod-ips 454> acpiphp: Slot [3] registered
Mod-ips 455> acpiphp: Slot [4] registered
Mod-ips 456> acpiphp: Slot [5] registered
Mod-ips 457> acpiphp: Slot [6] registered
Mod-ips 458> acpiphp: Slot [7] registered
Mod-ips 459> acpiphp: Slot [8] registered
Mod-ips 460> acpiphp: Slot [9] registered
Mod-ips 461> acpiphp: Slot [10] registered
Mod-ips 462> acpiphp: Slot [11] registered
Mod-ips 463> acpiphp: Slot [12] registered
Mod-ips 464> acpiphp: Slot [13] registered
Mod-ips 465> acpiphp: Slot [14] registered
Mod-ips 466> acpiphp: Slot [15] registered
Mod-ips 467> acpiphp: Slot [16] registered
Mod-ips 468> acpiphp: Slot [17] registered
Mod-ips 469> acpiphp: Slot [18] registered
Mod-ips 470> acpiphp: Slot [19] registered
```

```

Mod-ips 471> acpiphp: Slot [20] registered
Mod-ips 472> acpiphp: Slot [21] registered
Mod-ips 473> acpiphp: Slot [22] registered
Mod-ips 474> acpiphp: Slot [23] registered
Mod-ips 475> acpiphp: Slot [24] registered
Mod-ips 476> acpiphp: Slot [25] registered
Mod-ips 477> acpiphp: Slot [26] registered
Mod-ips 478> acpiphp: Slot [27] registered
Mod-ips 479> acpiphp: Slot [28] registered
Mod-ips 480> acpiphp: Slot [29] registered
Mod-ips 481> acpiphp: Slot [30] registered
Mod-ips 482> acpiphp: Slot [31] registered
Mod-ips 483> shpchp: Standard Hot Plug PCI Controller Driver version: 0.4
Mod-ips 484> fakephp: Fake PCI Hot Plug Controller Driver
Mod-ips 485> fakephp: pci_hp_register failed with error -16
Mod-ips 486> fakephp: pci_hp_register failed with error -16
Mod-ips 487> fakephp: pci_hp_register failed with error -16
Mod-ips 488> fakephp: pci_hp_register failed with error -16
Mod-ips 489> fakephp: pci_hp_register failed with error -16
Mod-ips 490> fakephp: pci_hp_register failed with error -16
Mod-ips 491> fakephp: pci_hp_register failed with error -16
Mod-ips 492> processor ACPI_CPU:00: registered as cooling_device0
Mod-ips 493> processor ACPI_CPU:01: registered as cooling_device1
Mod-ips 494> processor ACPI_CPU:02: registered as cooling_device2
Mod-ips 495> processor ACPI_CPU:03: registered as cooling_device3
Mod-ips 496> processor ACPI_CPU:04: registered as cooling_device4
Mod-ips 497> processor ACPI_CPU:05: registered as cooling_device5
Mod-ips 498> hpet_acpi_add: no address or irqs in _CRS
Mod-ips 499> Non-volatile memory driver v1.3
Mod-ips 500> Linux agpgart interface v0.103
Mod-ips 501> ipmi message handler version 39.2
Mod-ips 502> ipmi device interface
Mod-ips 503> IPMI System Interface driver.
Mod-ips 504> ipmi_si: Unable to find any System Interface(s)
Mod-ips 505> IPMI SMB Interface driver
Mod-ips 506> IPMI Watchdog: driver initialized
Mod-ips 507> Copyright (C) 2004 MontaVista Software - IPMI Powerdown via sys_reboot.
Mod-ips 508> Serial: 8250/16550 driver, 4 ports, IRQ sharing enabled
Mod-ips 509> ?serial8250: ttyS0 at I/O 0x3f8 (irq = 4) is a 16550A
Mod-ips 510> serial8250: ttyS1 at I/O 0x2f8 (irq = 3) is a 16550A
Mod-ips 511> 00:06: ttyS0 at I/O 0x3f8 (irq = 4) is a 16550A
Mod-ips 512> 00:07: ttyS1 at I/O 0x2f8 (irq = 3) is a 16550A
Mod-ips 513> brd: module loaded
Mod-ips 514> loop: module loaded
Mod-ips 515> lpc: version 0.1 (Nov 10 2011)
Mod-ips 516> tun: Universal TUN/TAP device driver, 1.6
Mod-ips 517> tun: (C) 1999-2004 Max Krasnyansky <maxk@qualcomm.com>
Mod-ips 518> Uniform Multi-Platform E-IDE driver
Mod-ips 519> piix 0000:00:01.1: IDE controller (0x8086:0x7010 rev 0x00)
Mod-ips 520> piix 0000:00:01.1: not 100native mode: will probe irqs later
Mod-ips 521>     ide0: BM-DMA at 0xc000-0xc007
Mod-ips 522>     ide1: BM-DMA at 0xc008-0xc00f
Mod-ips 523> hda: QEMU HARDDISK, ATA DISK drive
Mod-ips 524> Clocksource tsc unstable (delta = 2851415955127 ns)
Mod-ips 525> hda: MWDMA2 mode selected
Mod-ips 526> hdc: QEMU DVD-ROM, ATAPI CD/DVD-ROM drive
Mod-ips 527> hdc: MWDMA2 mode selected
Mod-ips 528> ide0 at 0x1f0-0x1f7,0x3f6 on irq 14
Mod-ips 529> ide1 at 0x170-0x177,0x376 on irq 15
Mod-ips 530> ide_generic: please use "probe_mask=0x3f" module parameter for probing all
legacy
Mod-ips 531> ISA IDE ports
Mod-ips 532> ide-gd driver 1.18
Mod-ips 533> hda: max request size: 512KiB

```



```
Mod-ips 534> hda: 7815168 sectors (4001 MB) w/256KiB Cache, CHS=7753/255/63
Mod-ips 535> hda: cache flushes supported
Mod-ips 536> hda: hda1 hda2 hda3 hda4
Mod-ips 537> Driver 'sd' needs updating - please use bus_type methods
Mod-ips 538> Driver 'sr' needs updating - please use bus_type methods
Mod-ips 539> ehci_hcd: USB 2.0 'Enhanced' Host Controller (EHCI) Driver
Mod-ips 540> ohci_hcd: USB 1.1 'Open' Host Controller (OHCI) Driver
Mod-ips 541> uhci_hcd: USB Universal Host Controller Interface driver
Mod-ips 542> Initializing USB Mass Storage driver...
Mod-ips 543> usbcore: registered new interface driver usb-storage
Mod-ips 544> USB Mass Storage support registered.
Mod-ips 545> PNP: PS/2 Controller [PNP0303:KBD,PNP0f13:MOU] at 0x60,0x64 irq 1,12
Mod-ips 546> serio: i8042 KBD port at 0x60,0x64 irq 1
Mod-ips 547> serio: i8042 AUX port at 0x60,0x64 irq 12
Mod-ips 548> mice: PS/2 mouse device common for all mice
Mod-ips 549> rtc_cmos 00:01: rtc core: registered rtc_cmos as rtc0
Mod-ips 550> rtc0: alarms up to one day, 114 bytes nvram
Mod-ips 551> input: AT Translated Set 2 keyboard as /class/input/input0
Mod-ips 552> i2c /dev entries driver
Mod-ips 553> piix4_smbus 0000:00:01.3: SMBus Host Controller at 0xb100, revision 0
Mod-ips 554> device-mapper: ioctl: 4.14.0-ioctl (2008-04-23) initialised:
dm-devel@redhat.com
Mod-ips 555> cpuidle: using governor ladder
Mod-ips 556> usbcore: registered new interface driver usbhid
Mod-ips 557> usbhid: v2.6:USB HID core driver
Mod-ips 558> TCP cubic registered
Mod-ips 559> IPv6: Loaded, but is disabled by default. IPv6 may be enabled on individual
interf
Mod-ips 560> aces.
Mod-ips 561> NET: Registered protocol family 10
Mod-ips 562> NET: Registered protocol family 17
Mod-ips 563> NET: Registered protocol family 5
Mod-ips 564> rtc_cmos 00:01: setting system clock to 2012-01-17 14:06:34 UTC (1326809194)
Mod-ips 565> Freeing unused kernel memory: 1032k freed
Mod-ips 566> Write protecting the kernel read-only data: 4272k
Mod-ips 567> Loader init started...
Mod-ips 568> kjournald starting. Commit interval 5 seconds
Mod-ips 569> EXT3-fs: mounted filesystem with ordered data mode.
Mod-ips 570> input: ImExPS/2 Generic Explorer Mouse as /class/input/input1
Mod-ips 571> 51216 blocks
Mod-ips 572> Checking rootrw fs: corrected filesystem
Mod-ips 573> kjournald starting. Commit interval 5 seconds
Mod-ips 574> EXT3 FS on hda2, internal journal
Mod-ips 575> EXT3-fs: mounted filesystem with ordered data mode.
Mod-ips 576> mkdir: cannot create directory '/lib/modules': File exists
Mod-ips 577> init started: BusyBox v1.13.1 (2011-11-01 07:21:34 CDT)
Mod-ips 578> starting pid 678, tty '': '/etc/init.d/rc.init'
Mod-ips 579> Checking system fs: no errors
Mod-ips 580> kjournald starting. Commit interval 5 seconds
Mod-ips 581> EXT3-fs: mounted filesystem with ordered data mode.
Mod-ips 582> /etc/init.d/rc.init: line 102: /proc/sys/vm/bdflush: No such file or
directory
Mod-ips 583> starting pid 728, tty '': '/etc/init.d/rcS'
Mod-ips 584> Initializing random number generator... done.
Mod-ips 585> Starting network... ifconfig lo 127.0.0.1 netmask 255.255.255.255 up
Mod-ips 586> starting inetd
Mod-ips 587> done
Mod-ips 588> Starting sshd:
Mod-ips 589> Starting nscd:
Mod-ips 590> Set Irq Affinity ... cpus:
Mod-ips 591> Checking kernel allocated memory: EXT3 FS on hda1, internal journal
Mod-ips 592> [ OK ]
Mod-ips 593> Unloading REGEX-CP drivers ...
Mod-ips 594> Loading REGEX-CP drivers ...
```

```

Mod-ips 595> ACPI: PCI Interrupt Link [LNKD] enabled at IRQ 11
Mod-ips 596> cpp_user_kvm 0000:00:04.0: PCI INT A -> Link[LNKD] -> GSI 11 (level, high) ->
IRQ
Mod-ips 597> 11
Mod-ips 598> Detected cpp_user_kvm device with 33554432 bytes of shared memory
Mod-ips 599> Device 0: model=LCPX8640, cpc=T2005, cpe0=None, cpe1=None
Mod-ips 600> Load cidmodcap:
Mod-ips 601> Create node:
Mod-ips 602> ln: /etc/modprobe.conf: File exists
Mod-ips 603> Shutting down network... ifconfig lo down
Mod-ips 604> ifconfig lo down
Mod-ips 605> done
Mod-ips 606> Load ihm:
Mod-ips 607> Create node:
Mod-ips 608> Load kvm_ivshmem: IVSHMEM: writing 0x0 to 0xc86cf8
Mod-ips 609> IVSHMEM: IntrMask write(w) val = 0xffff
Mod-ips 610> Create node:
Mod-ips 611> Create node:
Mod-ips 612> Create node:
Mod-ips 613> Set Irq Affinity ... cpus: 6
Mod-ips 614> Starting network... ifconfig lo 127.0.0.1 netmask 255.255.255.255 up
Mod-ips 615> done
Mod-ips 616> Creating boot.info[ OK ]
Mod-ips 617> Checking for system modifications since last boot[ OK ]
Mod-ips 618> Checking model identification[ OK ]
Mod-ips 619> Model: ASA-5555
Mod-ips 620> Model=ASA-5555
Mod-ips 621> Unable to set speed and duplex for user mode interfaces
Mod-ips 622> interface type 0x8086:0x100e at pci address 0:6.0(0) is currently named eth1
Mod-ips 623> Renaming eth1 --> ma0_0
Mod-ips 624> interface type 0x8086:0x100e at pci address 0:7.0(0) is currently named po0_0
Mod-ips 625> interface type 0x8086:0x100e at pci address 0:5.0(0) is currently named eth0
Mod-ips 626> Renaming eth0 --> sy0_0
Mod-ips 627> Initializing access list
Mod-ips 628> MGMT_INTFC_CIDS_NAME Management0/0
Mod-ips 629> MGMT_INTFC_OS_NAME ma0_0
Mod-ips 630> SYSTEM_PCI_IDS 0x0030,0x0028
Mod-ips 631> Load rebootkom:
Mod-ips 632> root: Starting SSM controlplane
Mod-ips 633> Starting CIDS:
Mod-ips 634> starting pid 1718, tty '/dev/ttyS0': '/sbin/getty -L ttyS0 9600 vt100'

```

ASA 5500-X IPS SSP Failover Scenarios

The following failover scenarios apply to the ASA 5500-X series in the event of configuration changes, signature/signature engine updates, service packs, and SensorApp crashes on the ASA 5500-X IPS SSP.

Single ASA in Fail-Open Mode

- If the ASA is configured in fail-open mode for the ASA 5500-X IPS SSP, and the ASA 5500-X IPS SSP experiences a configuration change or signature/signature engine update, traffic is passed through the ASA without being inspected.
- If the ASA is configured in fail-open mode for the ASA 5500-X IPS SSP, and the ASA 5500-X IPS SSP experiences a SensorApp crash or a service pack upgrade, traffic is passed through the ASA without being inspected.

Single ASA in Fail-Close Mode

- If the ASA is configured in fail-close mode for the ASA 5500-X IPS SSP, and the ASA 5500-X IPS SSP experiences a configuration change or a signature/signature engine update, traffic is stopped from passing through the ASA.
- If the ASA is configured in fail-close mode for the ASA 5500-X IPS SSP, and the ASA 5500-X IPS SSP experiences a SensorApp crash or a service pack upgrade, traffic is stopped from passing through the ASA.

Two ASAs in Fail-Open Mode

- If the ASAs are configured in fail-open mode and if the ASA 5500-X IPS SSP on the active ASA experiences a configuration change or a signature/signature engine update, traffic is still passed through the active ASA without being inspected. Failover is not triggered.
- If the ASAs are configured in fail-open mode, and if the ASA 5500-X IPS SSP on the active ASA experiences a SensorApp crash or a service pack upgrade, failover is triggered and traffic passes through the ASA 5500-X IPS SSP that was previously the standby ASA 5500-X IPS SSP.

Two ASAs in Fail-Close Mode

- If the ASAs are configured in fail-close mode, and if the ASA 5500-X IPS SSP on the active ASA experiences a configuration change or a signature/signature engine update, traffic is stopped from passing through the active ASA. No failover is triggered.
- If the ASAs are configured in fail-close mode, and if the ASA 5500-X IPS SSP on the active ASA experiences a SensorApp crash or a service pack upgrade, failover is triggered and traffic passes through the ASA 5500-X IPS SSP that was previously the standby for the ASA 5500-X IPS SSP.

Configuration Examples

Use the following configuration for the primary ASA:

```
interface GigabitEthernet0/7
  description LAN Failover Interface

failover
failover lan unit primary
failover lan interface folink GigabitEthernet0/7
failover interface ip folink 172.27.48.1 255.255.255.0 standby 172.27.48.2
```

Use the following configuration for the secondary ASA:

```
interface GigabitEthernet0/7
  description LAN Failover Interface

failover
failover lan unit secondary
failover lan interface folink GigabitEthernet0/7
failover interface ip folink 172.27.48.1 255.255.255.0 standby 172.27.48.2
```

New and Modified Commands

This section describes the new and modified Cisco ASA commands that support the ASA 5500-X IPS SSP and are used to configure the ASA 5500-X IPS SSP.

**Note**

All other Cisco ASA CLI commands are documented in the *Cisco Security Appliance Command Reference* on Cisco.com at http://www.cisco.com/en/US/products/ps6120/prod_command_reference_list.html.

This section contains the following topic:

- [allocate-ips](#), page 18-23

allocate-ips

To allocate an IPS virtual sensor to a security context if you have the ASA 5500-X IPS SSP installed, use the **allocate-ips** command in context configuration mode. To remove a virtual sensor from a context, use the **no** form of this command.

allocate-ips *sensor_name* [*mapped_name*] [**default**]

no allocate-ips *sensor_name* [*mapped_name*] [**default**]

Syntax Description

default	(Optional) Sets one sensor per context as the default sensor; if the context configuration does not specify a sensor name, the context uses this default sensor. You can only configure one default sensor per context. If you want to change the default sensor, enter the no allocate-ips sensor_name command to remove the current default sensor before you allocate a new default sensor. If you do not specify a sensor as the default, and the context configuration does not include a sensor name, then traffic uses the default sensor on the ASA 5500-X IPS SSP.
<i>mapped_name</i>	(Optional) Sets a mapped name as an alias for the sensor name that can be used within the context instead of the actual sensor name. If you do not specify a mapped name, the sensor name is used within the context. For security purposes, you might not want the context administrator to know which sensors are being used by the context. Or you might want to genericize the context configuration. For example, if you want all contexts to use sensors called “sensor1” and “sensor2,” then you can map the “highsec” and “lowsec” sensors to sensor1 and sensor2 in context A, but map the “medsec” and “lowsec” sensors to sensor1 and sensor2 in context B.
<i>sensor_name</i>	Sets the sensor name configured on the ASA 5500-X IPS SSP. To view the sensors that are configured on the ASA 5500-X IPS SSP, enter allocate-ips ? . All available sensors are listed. You can also enter the show ips command. In the system execution space, the show ips command lists all available sensors; if you enter it in the context, it shows the sensors you already assigned to the context. If you specify a sensor name that does not yet exist on the ASA 5500-X IPS SSP, you get an error, but the allocate-ips command is entered as is. Until you create a sensor of that name on the ASA 5500-X IPS SSP, the context assumes the sensor is down.

Defaults

No default behavior or values.

Command Modes

The following table shows the modes in which you can enter the command:

Command Mode	Firewall Mode		Security Context		
	Routed	Transparent	Single	Multiple	
				Context	System
Context configuration	•	•	—	—	•

Command History

Release	Modification
8.0(2)	This command was introduced.

Usage Guidelines

You can assign one or more IPS virtual sensors to each context. Then, when you configure the context to send traffic to the ASA 5500-X IPS SSP using the **ips** command, you can specify a sensor that is assigned to the context; you cannot specify a sensor that you did not assign to the context. If you do not assign any sensors to a context, then the default sensor configured on the ASA 5500-X IPS SSP is used. You can assign the same sensor to multiple contexts.

**Note**

You do not need to be in multiple context mode to use virtual sensors; you can be in single mode and use different sensors for different traffic flows.

Examples

The following example assigns sensor1 and sensor2 to context A, and sensor1 and sensor3 to context B. Both contexts map the sensor names to “ips1” and “ips2.” In context A, sensor1 is set as the default sensor, but in context B, no default is set so the default that is configured on the ASA 5500-X IPS SSP is used.

```
hostname(config-ctx) # context A
hostname(config-ctx) # allocate-interface gigabitethernet0/0.100 int1
hostname(config-ctx) # allocate-interface gigabitethernet0/0.102 int2
hostname(config-ctx) # allocate-interface gigabitethernet0/0.110-gigabitethernet0/0.115
int3-int8
hostname(config-ctx) # allocate-ips sensor1 ips1 default
hostname(config-ctx) # allocate-ips sensor2 ips2
hostname(config-ctx) # config-url ftp://user1:passw0rd@10.1.1.1/configlets/test.cfg
hostname(config-ctx) # member gold

hostname(config-ctx) # context sample
hostname(config-ctx) # allocate-interface gigabitethernet0/1.200 int1
hostname(config-ctx) # allocate-interface gigabitethernet0/1.212 int2
hostname(config-ctx) # allocate-interface gigabitethernet0/1.230-gigabitethernet0/1.235
int3-int8
hostname(config-ctx) # allocate-ips sensor1 ips1
hostname(config-ctx) # allocate-ips sensor3 ips2
hostname(config-ctx) # config-url ftp://user1:passw0rd@10.1.1.1/configlets/sample.cfg
hostname(config-ctx) # member silver
```

Related Commands

Command	Description
context	Creates a security context in the system configuration and enters context configuration mode.
ips	Diverts traffic to the ASA 5500-X IPS SSP for inspection.
show context	Shows a list of contexts (system execution space) or information about the current context.
show ips	Shows the virtual sensors configured on the ASA 5500-X IPS SSP.



Configuring the ASA 5585-X IPS SSP

This chapter contains procedures that are specific to configuring the ASA 5585-X IPS SSP. It contains the following sections:

- [ASA 5585-X IPS SSP Notes and Caveats, page 19-1](#)
- [Configuration Sequence for the ASA 5585-X IPS SSP, page 19-2](#)
- [Verifying Initialization for the ASA 5585-X IPS SSP, page 19-3](#)
- [Creating Virtual Sensors for the ASA 5585-X IPS SSP, page 19-4](#)
- [The ASA 5585-X IPS SSP and the Normalizer Engine, page 19-10](#)
- [The ASA 5585-X IPS SSP and Bypass Mode, page 19-10](#)
- [ASA 5585-X IPS SSP and Jumbo Packets, page 19-11](#)
- [TCP Reset Differences Between IPS Appliances and the ASA 5585-X IPS SSP, page 19-11](#)
- [Reloading IPS Messages, page 19-12](#)
- [Reloading, Shutting Down, Resetting, and Recovering the ASA 5585-X IPS SSP, page 19-12](#)
- [Health and Status Information, page 19-13](#)
- [Traffic Flow Stopped on IPS Switchports, page 19-16](#)
- [Failover Scenarios, page 19-16](#)

ASA 5585-X IPS SSP Notes and Caveats

The following notes and caveats apply to configuring the ASA 5585-X IPS SSP:

- The ASA 5585-X IPS SSP is supported in ASA 8.2(4.4) and later as well as ASA 8.4(2) and later. It is not supported in ASA 8.3(x).
- All IPS platforms allow ten concurrent CLI sessions.
- Anomaly detection is disabled by default.
- The ASA 5585-X IPS SSP does not support CDP mode.
- The ASA 5585-X IPS SSP does not support the inline TCP session tracking mode.
- For the ASA 5585-X IPS SSP, normalization is performed by the adaptive security appliance and not the IPS.

- The ASA 5585-X IPS SSP does not support bypass mode. The adaptive security appliance will either fail open, fail close, or fail over depending on the configuration of the adaptive security appliance and the type of activity being done on the IPS.
- The ASA 5585-X IPS SSP supports the String ICMP XL, String TCP XL, and String UDP XL engines. These engines provide optimized operation for these platforms.
- The ASA 5585-X IPS SSP has four types of ports (console, management, GigabitEthernet, and 10GE). The console and management ports (on the right front panel of the ASA 5585-X IPS SSP) are configured and controlled by IPS software. The GigabitEthernet and 10GE ports (on the left front panel of the ASA 5585-X IPS SSP) are configured and controlled by ASA software rather than IPS software. However, when you reset or shut down the ASA 5585-X IPS SSP, the GigabitEthernet and 10GE ports will also link down. You should reset or shut down the ASA 5585-X IPS SSP during scheduled maintenance windows to minimize the effect of the link down on these ports.

Configuration Sequence for the ASA 5585-X IPS SSP

Perform the following tasks to configure the ASA 5585-X IPS SSP:

1. Obtain and install the current IPS software if your software is not up to date.
2. Obtain and install the license key.
3. Log (session) in to the ASA 5585-X IPS SSP.
4. Run the **setup** command to initialize the ASA 5585-X IPS SSP.
5. Verify initialization for the ASA 5585-X IPS SSP.
6. Configure the adaptive security appliance to send IPS traffic to the ASA 5585-X IPS SSP.
7. Perform other initial tasks, such as adding users, trusted hosts, and so forth.
8. Configure intrusion prevention.
9. Configure global correlation.
10. Perform miscellaneous tasks to keep your ASA 5585-X IPS SSP running smoothly.
11. Upgrade the IPS software with new signature updates and service packs as they become available.
12. Reimage the ASA 5585-X IPS SSP when needed.

For More Information

- For the procedure for logging in to the ASA 5585-X IPS SSP, see [Chapter 2, “Logging In to the Sensor.”](#)
- For the procedure for running the **setup** command, see [Advanced Setup for the ASA 5585-X IPS SSP, page 3-15.](#)
- For the procedure for verifying ASA 5585-X IPS SSP initialization, see [Verifying Initialization for the ASA 5585-X IPS SSP, page 19-3.](#)
- For the procedure for creating virtual sensors, see [Creating Virtual Sensors for the ASA 5585-X IPS SSP, page 19-4.](#)
- For the procedures for setting up the ASA 5585-X IPS SSP, see [Chapter 4, “Setting Up the Sensor.”](#)
- For the procedures for configuring intrusion prevention, see [Chapter 8, “Configuring Event Action Rules,”](#) [Chapter 7, “Defining Signatures,”](#) [Chapter 9, “Configuring Anomaly Detection,”](#) and [Chapter 14, “Configuring Attack Response Controller for Blocking and Rate Limiting.”](#)

- For the procedures for configuring global correlation, see [Chapter 10, “Configuring Global Correlation.”](#)
- For the procedures for keeping your ASA 5585-X IPS SSP running smoothly, see [Chapter 17, “Administrative Tasks for the Sensor.”](#)
- For more information on how to obtain Cisco IPS software, see [Chapter 20, “Obtaining Software.”](#)
- For the procedure for reimaging the ASA 5585-X IPS SSP, see [Installing the System Image for the ASA 5585-X IPS SSP, page 21-23.](#)

Verifying Initialization for the ASA 5585-X IPS SSP

You can use the **show module slot details** command to verify that you have initialized the ASA 5585-X IPS SSP and to verify that you have the correct software version.

To verify initialization, follow these steps:

-
- Step 1** Log in to the adaptive security appliance.
 - Step 2** Obtain the details about the ASA 5585-X IPS SSP.

```
asa# show module 1 details
Getting details from the Service Module, please wait...
ASA 5585-X IPS Security Services Processor-10 with 8GE
Model:          ASA5585-SSP-IPS10
Hardware version: 1.0
Serial Number:  JAF1350ABSL
Firmware version: 2.0(1)3
Software version: 7.2(1)E4
MAC Address Range: 8843.e12f.5414 to 8843.e12f.541f
App. name:      IPS
App. Status:    Up
App. Status Desc: Normal Operation
App. version:   7.2(1)E4
Data plane Status: Up
Status:         Up
Mgmt IP addr:   192.0.2.3
Mgmt Network mask: 255.255.255.0
Mgmt Gateway:   192.0.2.254
Mgmt Access List: 10.0.0.0/8
Mgmt Access List: 64.0.0.0/8
Mgmt web ports: 443
Mgmt TLS enabled true
asa
```

- Step 3** Confirm the information.
-

Creating Virtual Sensors for the ASA 5585-X IPS SSP

This section describes how to create virtual sensors on the ASA 5585-X IPS SSP, and contains the following topics:

- [The ASA 5585-X IPS SSP and Virtualization, page 19-4](#)
- [The ASA 5585-X IPS SSP Virtual Sensor Configuration Sequence, page 19-5](#)
- [Creating Virtual Sensors, page 19-5](#)
- [Assigning Virtual Sensors to Adaptive Security Appliance Contexts, page 19-7](#)

The ASA 5585-X IPS SSP and Virtualization

The ASA 5585-X IPS SSP has two interfaces, the management interface (command and control) and the sensing interface. The command and control interface has an IP address and is used for configuring the ASA 5585-X IPS SSP. It is used by the ASA 5585-X IPS SSP to transmit security and status events to the IDM or IME. The ASA 5585-X IPS SSP command and control interface is named Management 0/0.



Caution

The ASA 5585-X IPS SSP has four types of ports (console, management, GigabitEthernet, and 10GE). The console and management ports (on the right front panel of the ASA 5585-X IPS SSP) are configured and controlled by IPS software. The GigabitEthernet and 10GE ports (on the left front panel of the ASA 5585-X IPS SSP) are configured and controlled by ASA software rather than IPS software. However, when you reset or shut down the ASA 5585-X IPS SSP, the GigabitEthernet and 10GE ports will also link down. You should reset or shut down the ASA 5585-X IPS SSP during scheduled maintenance windows to minimize the effect of the link down on these ports.

The command and control interface is permanently enabled. It is permanently mapped to a specific physical interface. You cannot use the command and control interface as either a sensing or alternate TCP reset interface.

Sensing interfaces are used to analyze traffic for security violations. There is only one sensing interface on the ASA 5585-X IPS SSP. It is named PortChannel 0/0 and is a backplane interface. All backplane interfaces have fixed speed, duplex, and state settings. These settings are protected in the default configuration on all backplane interfaces. You configure the ASA 5585-X IPS SSP interface by security context on the adaptive security appliance. The sensing interface is permanently enabled. When you create multiple virtual sensors, you must assign the sensing interface to only one virtual sensor. For the other virtual sensors you do not need to designate an interface.

After you create virtual sensors, you must map them to a security context on the adaptive security appliance using the **allocate-ips** command. You can map many security contexts to many virtual sensors.



Note

The **allocate-ips** command does not apply to single mode. In this mode, the adaptive security appliance accepts any virtual sensor named in a **policy-map** command.

The **allocate-ips** command adds a new entry to the security context database. A warning is issued if the specified virtual sensor does not exist; however, the configuration is allowed. The configuration is checked again when the **service-policy** command is processed. If the virtual sensor is not valid, the **fail-open** policy is enforced.

The ASA 5585-X IPS SSP Virtual Sensor Configuration Sequence

Follow this sequence to create virtual sensors on the ASA 5585-X IPS SSP, and to assign them to adaptive security appliance contexts:

1. Configure up to four virtual sensors.
2. Assign the ASA 5585-X IPS SSP sensing interface (PortChannel 0/0), to one of the virtual sensors.
3. (Optional) Assign virtual sensors to different contexts on the adaptive security appliance.
4. Use MPF to direct traffic to the targeted virtual sensor.

Creating Virtual Sensors



Note

You can create four virtual sensors.

Use the **virtual-sensor** *name* command in service analysis engine submode to create virtual sensors on the ASA 5585-X IPS SSP. You assign policies (anomaly detection, event action rules, and signature definition) to the virtual sensor. You can use the default policies, `ad0`, `rules0`, or `sig0`, or you can create new policies. Then you assign the sensing interface, PortChannel 0/0 for the ASA 5500-X IPS SSP, to one virtual sensor.

The following parameters apply:

- **anomaly-detection**—Specifies the anomaly detection parameters:
 - **anomaly-detection-name** *name*—Specifies the name of the anomaly detection policy.
 - **operational-mode**—Specifies the anomaly detection mode (**inactive**, **learn**, **detect**).



Note

Anomaly detection is disabled by default. You must enable it to configure or apply an anomaly detection policy. Enabling anomaly detection results in a decrease in performance.

- **description**—Provides a description of the virtual sensor.
- **event-action-rules**—Specifies the name of the event action rules policy.
- **signature-definition**—Specifies the name of the signature definition policy.
- **physical-interfaces**—Specifies the name of the physical interface.
- **no**—Removes an entry or selection.

Creating Virtual Sensors

To create a virtual sensor on the ASA 5585-X IPS SSP, follow these steps:

Step 1 Log in to the CLI using an account with administrator privileges.

Step 2 Enter service analysis mode.

```
sensor# configure terminal
sensor(config)# service analysis-engine
sensor(config-ana)#
```

Step 3 Add a virtual sensor.

```
sensor(config-ana)# virtual-sensor vs1
sensor(config-ana-vir)#
```

Step 4 Add a description for this virtual sensor.

```
sensor(config-ana-vir)# description virtual sensor 1
```

Step 5 Assign an anomaly detection policy and operational mode to this virtual sensor if you have enabled anomaly detection. If you do not want to use the default anomaly detection policy, ad0, you must create a new one using the **service anomaly-detection name** command, for example, ad1.

```
sensor(config-ana-vir)# anomaly-detection
sensor(config-ana-vir-ano)# anomaly-detection-name ad0
sensor(config-ana-vir-ano)# operational-mode learn
```

Step 6 Assign an event action rules policy to this virtual sensor. If you do not want to use the default event action rules policy, rules0, you must create a new one using the **service event-action-rules name** command, for example, rules1

```
sensor(config-ana-vir-ano)# exit
sensor(config-ana-vir)# event-action-rules rules0
```

Step 7 Assign a signature definition policy to this virtual sensor. If you do not want to use the default signature definition policy, sig0, you must create a new one using the **service signature-definition name** command, for example sig1.

```
sensor(config-ana-vir)# signature-definition sig0
```

Step 8 Assign the interface to one virtual sensor. By default the sensing interface is already assigned to the default virtual sensor, vs0. You must remove it from the default virtual sensor to assign it to another virtual sensor that you create.

```
sensor(config-ana-vir)# physical-interface PortChannel0/0
```

Step 9 Verify the virtual sensor settings.

```
sensor(config-ana-vir)# show settings
name: vs1
-----
description: virtual sensor 1 default:
signature-definition: sig1 default: sig0
event-action-rules: rules1 default: rules0
anomaly-detection
-----
anomaly-detection-name: ad1 default: ad0
operational-mode: learn default: detect
-----
physical-interface (min: 0, max: 999999999, current: 2)
-----
name: PortChannel0/0
subinterface-number: 0 <defaulted>
-----
logical-interface (min: 0, max: 999999999, current: 0)
-----
-----
sensor(config-ana-vir)#
```

Step 10 Exit analysis engine mode.

```
sensor(config-ana-vir)# exit
```

```
sensor(config-ana)# exit
Apply Changes:[yes]:
sensor(config)#
```

Step 11 Press **Enter** to apply the changes or enter **no** to discard them.

For More Information

- For the procedures for creating and configuring anomaly detection policies, see [Working With Anomaly Detection Policies, page 9-8](#).
- For the procedure for creating and configuring event action rules policies, see [Working With Event Action Rules Policies, page 8-8](#).
- For the procedure for creating and configuring signature definitions, [Working With Signature Definition Policies, page 7-2](#).
- For the procedure for enabling anomaly detection, see [Enabling Anomaly Detection, page 9-8](#).

Assigning Virtual Sensors to Adaptive Security Appliance Contexts

After you create virtual sensors on the ASA 5585-X IPS SSP, you must assign the virtual sensors to a security context on the adaptive security appliance.

The following parameters apply:

- **[no] allocate-ips *sensor_name* [*mapped_name*] [default]**—Allocates a virtual sensor to a security context. Supported modes are multiple mode, system context, and context submode.



Note You cannot allocate the same virtual sensor twice in a context.

- *sensor_name*—Specifies the name of the virtual sensor configured on the ASA 5585-X IPS SSP. You receive a warning message if the name is not valid.
- *mapped_name*—Specifies the name by which the security context knows the virtual sensor.



Note The mapped name is used to hide the real name of the virtual sensor from the context, usually done for reasons of security or convenience to make the context configuration more generic. If no mapped name is used, the real virtual sensor name is used. You cannot reuse a mapped name for two different virtual sensors in a context.

- **no**—De-allocates the sensor, looks through the policy map configurations, and deletes any IPS subcommand that refers to it.
- **default**—Specifies this virtual sensor as the default. All legacy IPS configurations that do not specify a virtual sensor are mapped to this virtual sensor.



Caution

You can only configure one default virtual sensor per context. You must turn off the default flag of an existing default virtual sensor before you can designate another virtual sensor as the default.

- **clear configure allocate-ips**—Removes the configuration.
- **allocate-ips?**—Displays the list of configured virtual sensors.

- **show context [detail]**—Updated to display information about virtual sensors. In user context mode, a new line is added to show the mapped names of all virtual sensors that have been allocated to this context. In system mode, two new lines are added to show the real and mapped names of virtual sensors allocated to this context.

You can assign multiple virtual sensors to a context. Multiple contexts can share one virtual sensor, and when sharing, the contexts can have different mapped names (aliases) for the same virtual sensor. The following procedure demonstrates how to add three security contexts in multiple mode and how to assign virtual sensors to these security contexts.

Assigning Virtual Sensors to Contexts

To assign virtual sensors to adaptive security appliance contexts in multiple mode for the ASA 5585-X IPS SSP, follow these steps:

Step 1 Log in to the adaptive security appliance.

Step 2 Display the list of available virtual sensors.

```
asa# show ips
Sensor Name      Sensor ID
-----
vs0              1
vs1              2
asa#
```

Step 3 Enter configuration mode.

```
asa# configure terminal
asa(config)#
```

Step 4 Enter multiple mode.

```
asa(config)# mode multiple
WARNING: This command will change the behavior of the device
WARNING: This command will initiate a Reboot
Proceed with change mode? [confirm] yes
asa(config)#
```

Step 5 Add three context modes to multiple mode.

```
asa(config)# admin-context admin
Creating context 'admin'... Done. (13)
asa(config)# context admin
asa(config-ctx)# allocate-interface GigabitEthernet0/0.101
asa(config-ctx)# allocate-interface GigabitEthernet0/1.102
asa(config-ctx)# allocate-interface Management0/0
asa(config-ctx)# config-url disk0:/admin.cfg
Cryptochecksum (changed): 0c34dc67 f413ad74 e297464a db211681
INFO: Context admin was created with URL disk0:/admin.cfg
INFO: Admin context will take some time to come up .... please wait.
asa(config-ctx)#
asa(config-ctx)# context c2
Creating context 'c2'... Done. (14)
asa(config-ctx)# allocate-interface GigabitEthernet0/0.103
asa(config-ctx)# allocate-interface GigabitEthernet0/1.104
asa(config-ctx)# config-url disk0:/c2.cfg

WARNING: Could not fetch the URL disk0:/c2.cfg
INFO: Creating context with default config
asa(config-ctx)#
asa(config-ctx)# context c3
Creating context 'c3'... Done. (15)
```

```
asa(config-ctx)# all
asa(config-ctx)# allocate-in
asa(config-ctx)# allocate-interface g0/2
asa(config-ctx)# allocate-interface g0/3
asa(config-ctx)# config-url disk0:/c3.cfg

WARNING: Could not fetch the URL disk0:/c3.cfg
INFO: Creating context with default config
asa(config-ctx)#
```

Step 6 Assign virtual sensors to the security contexts.

```
asa(config)# context admin
asa(config-ctx)# allocate-ips vs0 adminvs0
asa(config-ctx)# exit
asa(config)# context c2
asa(config-ctx)# allocate-ips vs1 c2vs1
asa(config)# context c3
asa(config-ctx)# allocate-ips vs0 c3vs0
asa(config-ctx)# allocate-ips vs1 c3vs1
asa(config-ctx)#
```

Step 7 Configure MPF for each context.



Note The following example shows context 3 (c3).

```
asa(config)# context c3
asa/c3(config)# class-map any
asa/c3(config-cmap)# match access-list any
asa/c3(config-cmap)# exit
asa/c3(config)# policy-map ips_out
asa/c3(config-pmap)# class any
asa/c3(config-pmap-c)# ips promiscuous fail-close sensor c3vs1
asa/c3(config-pmap-c)# policy-map ips_in
asa/c3(config-pmap)# class any
asa/c3(config-pmap-c)# ips inline fail-open sensor c3vs0
asa/c3(config-pmap-c)# service-policy ips_out interface outside
asa/c3(config)# service-policy ips_in interface inside
asa/c3(config)#
```

Step 8 Confirm the configuration.

```
asa/c3(config)# exit
asa(config)# show ips detail
```

Sensor Name	Sensor ID	Allocated To	Mapped Name
vs0	1	admin	adminvs0
		c3	c3vs0
vs1	2	c2	c2vs1
		c3	c3vs1

```
asa(config)#
```

The ASA 5585-X IPS SSP and the Normalizer Engine

The majority of the features in the Normalizer engine are not used on the ASA 5585-X IPS SSP, because the ASA itself handles the normalization. Packets on the ASA IPS modules go through a special path in the Normalizer that only reassembles fragments and puts packets in the right order for the TCP stream. The Normalizer does not do any of the normalization that is done on an inline IPS appliance, because that causes problems in the way the ASA handles the packets.

The following Normalizer engine signatures are not supported:

- 1300.0
- 1304.0
- 1305.0
- 1307.0
- 1308.0
- 1309.0
- 1311.0
- 1315.0
- 1316.0
- 1317.0
- 1330.0
- 1330.1
- 1330.2
- 1330.9
- 1330.10
- 1330.12
- 1330.14
- 1330.15
- 1330.16
- 1330.17
- 1330.18

For More Information

For detailed information about the Normalizer engine, see [Normalizer Engine](#), page B-36.

The ASA 5585-X IPS SSP and Bypass Mode

The ASA 5585-X IPS SSP does not support bypass mode. The adaptive security appliance will either fail open, fail close, or fail over depending on the configuration of the adaptive security appliance and the type of activity being done on the ASA 5585-X IPS SSP.

The SensorApp Fails

The following occurs when the SensorApp fails:

- If the adaptive security appliance is configured for failover, then the adaptive security appliance fails over.
- If the adaptive security appliance is not configured for failover or failover is not possible:
 - If set to fail-open, the adaptive security appliance passes traffic without sending it to the ASA IPS module.
 - If set to fail-close, the adaptive security appliance stops passing traffic until the ASA IPS module is restarted.

The SensorApp is Reconfigured

The following occurs when the SensorApp is reconfigured:

- If set to fail-open, the adaptive security appliance passes traffic without sending it to the ASA IPS module.
- If set to fail-close, the adaptive security appliance stops passing traffic until the ASA IPS module is restarted.

**Note**

The adaptive security appliance does not failover unless the reconfiguration is not completed.

ASA 5585-X IPS SSP and Jumbo Packets

The jumbo packet count in the **show interface** command output from the lines `Total Jumbo Packets Received` and `Total Jumbo Packets Transmitted` for ASA IPS modules may be larger than expected due to some packets that were almost jumbo size on the wire being counted as jumbo size by the IPS. This miscount is a result of header bytes added to the packet by the ASA before the packet is transmitted to the IPS. For IPv4, 58 bytes of header data are added. For IPv6, 78 bytes of header data are added. The ASA removes the added IPS header before the packet leaves the ASA.

TCP Reset Differences Between IPS Appliances and the ASA 5585-X IPS SSP

The IPS appliance sends TCP reset packets to both the attacker and victim when `reset-tcp-connection` is selected. The IPS appliance sends a TCP reset packet only to the victim under the following circumstances:

- When a `deny-packet-inline` or `deny-connection-inline` is selected
- When TCP-based signatures and `reset-tcp-connection` have NOT been selected

In the case of the ASA 5585-X IPS SSP, the TCP reset request is sent to the ASA, and then the ASA sends the TCP reset packets. The ASA sends TCP reset packets to both the attacker and victim when the `reset-tcp-connection` is selected. When `deny-packet-inline` or `deny-connection-inline` is selected, the ASA sends the TCP reset packet to either the attacker or victim depending on the configuration of the signature. Signatures configured to swap the attacker and victim when reporting the alert can cause the ASA to send the TCP reset packet to the attacker.

Reloading IPS Messages

The following messages generated during some IPS signature and global correlation updates for IPS 7.1 and later on the ASA 5585-X IPS SSP can cause confusion since the IPS is not reloading:

```
ASA5585-SSP-IPS20 Module in slot 1, application up "IPS", version "7.1(1)E4" Normal
Operation
ASA5585-SSP-IPS20 Module in slot 1, application reloading "IPS", version "7.1(1)E4" Config
Change
```

These messages are generated during some, but not all, of the global correlation updates that are attempted every five minutes. This is expected behavior. There is a global correlation check every five minutes, but there may not be an update available, thus the message appears every hour or so. When a global correlation update actually takes place, a message is sent from the IPS to the ASA indicating that a configuration change is taking place.

Reloading, Shutting Down, Resetting, and Recovering the ASA 5585-X IPS SSP



Note

You can enter the **hw-module** commands from privileged EXEC mode or from global configuration mode. You can enter the commands in single routed mode and single transparent mode. For adaptive security appliances operating in multi-mode (routed or transparent multi-mode) you can only execute the **hw-module** commands from the system context (not from administrator or user contexts).

Use the following commands to reload, shut down, reset, recover the password, and recover the ASA 5585-X IPS SSP directly from the adaptive security appliance:

- **hw-module module slot_number reload**—This command reloads the software on the ASA 5585-X IPS SSP without doing a hardware reset. It is effective only when the module is in the Up state.
- **hw-module module slot_number shutdown**—This command shuts down the software on the ASA 5585-X IPS SSP. It is effective only when the module is in Up state.
- **hw-module module slot_number reset**—This command performs a hardware reset of the ASA 5585-X IPS SSP. It is applicable when the module is in the Up/Down/Unresponsive/Recover states.
- **hw-module module slot_number password-reset**—This command restores the cisco CLI account password on the ASA 5585-X IPS SSP to the default **cisco**.
- **hw-module module slot_number recover [boot | stop | configure]**—The **recover** command displays a set of interactive options for setting or changing the recovery parameters. To change the parameter or keep the existing setting, press **Enter**.
 - **hw-module module slot_number recover boot**—This command initiates recovery of the ASA 5585-X IPS SSP. It is applicable only when the module is in the Up state.
 - **hw-module module slot_number recover stop**—This command stops recovery of the ASA 5585-X IPS SSP. It is applicable only when the module is in the Recover state.

**Caution**

If the ASA 5585-X IPS SSP recovery needs to be stopped, you must issue the **hw-module module 1 recover stop** command within 30 to 45 seconds after starting the recovery. Waiting any longer can lead to unexpected consequences. For example, the module may come up in the Unresponsive state.

- **hw-module module 1 recover configure**—Use this command to configure parameters for the ASA 5585-X IPS SSP recovery. The essential parameters are the IP address and recovery image TFTP URL location.

Example

```
ips-ssp# hardware-module module 1 recover configure
Image URL [tftp://10.89.146.1/IPS-SSP_10-K9-sys-1.1-a-7.2-1-E4.img]:
Port IP Address [10.89.149.226]:
VLAN ID [0]:
Gateway IP Address [10.89.149.254]:
```

For More Information

For the procedure for recovering the ASA 5585-X IPS SSP system image, see [Installing the System Image for the ASA 5585-X IPS SSP, page 21-23](#).

Health and Status Information

To see the general health of the ASA 5585-X IPS SSP, use the **show module 1 details** command.

```
asa# show module 1 details
Getting details from the Service Module, please wait...
ASA 5585-X IPS Security Services Processor-20 with 8GE
Model:                ASA5585-SSP-IPS20
Hardware version:     1.0
Serial Number:        ABC1234DEFG
Firmware version:     2.0(1)3
Software version:     7.2(1)E4
MAC Address Range:    8843.e12f.5414 to 8843.e12f.541f
App. name:            IPS
App. Status:          Up
App. Status Desc:     Normal Operation
App. version:         7.2(1)E4
Data plane Status:    Up
Status:               Up
Mgmt IP addr:         192.0.2.3
Mgmt Network mask:    255.255.255.0
Mgmt Gateway:         192.0.2.254
Mgmt Access List:     10.0.0.0/8
Mgmt Access List:     64.0.0.0/8
Mgmt web ports:       443
Mgmt TLS enabled:     true
asa
```

The output shows that the ASA 5585-X IPS SSP is up. If the status reads `Down`, you can reset it using the **hw-module module 1 reset** command.

```
asa# hw-module module 1 reset
The module in slot 1 should be shut down before
resetting it or loss of configuration may occur.
Reset module in slot 1? [confirm]
Reset issued for module in slot 1
asa# show module 1 details
Getting details from the Service Module, please wait...
```

```

Unable to read details from slot 1
ASA 5585-X IPS Security Services Processor-20 with 8GE
Model:          ASA5585-SSP-IPS20
Hardware version: 1.0
Serial Number:  ABC1234DEFG
Firmware version: 2.0(7)0
Software version: 7.2(1)E4
MAC Address Range: 5475.d029.7f9c to 5475.d029.7fa7
App. name:      IPS
App. Status:    Not Applicable
App. Status Desc: Not Applicable
App. version:   7.2(1)E4
Data plane Status: Not Applicable
Status:         Shutting Down
asa# show module 1 details
Getting details from the Service Module, please wait...
Unable to read details from slot 1
ASA 5585-X IPS Security Services Processor-20 with 8GE
Model:          ASA5585-SSP-IPS20
Hardware version: 1.0
Serial Number:  ABC1234DEFG
Firmware version: 2.0(7)0
Software version: 7.2(1)E4
MAC Address Range: 5475.d029.7f9c to 5475.d029.7fa7
App. name:      IPS
App. Status:    Not Applicable
App. Status Desc: Not Applicable
App. version:   7.2(1)E4
Data plane Status: Not Applicable
Status:         Down
asa# show module 1 details
Getting details from the Service Module, please wait...
Unable to read details from slot 1
ASA 5585-X IPS Security Services Processor-20 with 8GE
Model:          ASA5585-SSP-IPS20
Hardware version: 1.0
Serial Number:  ABC1234DEFG
Firmware version: 2.0(7)0
Software version: 7.2(1)E4
MAC Address Range: 5475.d029.7f9c to 5475.d029.7fa7
App. name:      IPS
App. Status:    Not Applicable
App. Status Desc: Not Applicable
App. version:   7.2(1)E4
Data plane Status: Not Applicable
Status:         Init
asa# show module 1 details
Getting details from the Service Module, please wait...
ASA 5585-X IPS Security Services Processor-20 with 8GE
Model:          ASA5585-SSP-IPS20
Hardware version: 1.0
Serial Number:  ABC1234DEFG
Firmware version: 2.0(7)0
Software version: 7.2(1)E4
MAC Address Range: 5475.d029.7f9c to 5475.d029.7fa7
App. name:      IPS
App. Status:    Reload
App. Status Desc: Starting up
App. version:   7.2(1)E4
Data plane Status: Down
Status:         Up
Mgmt IP addr:   192.0.2.3
Mgmt Network mask: 255.255.255.0
Mgmt Gateway:  192.0.2.254

```

```

Mgmt Access List: 0.0.0.0/0
Mgmt web ports: 443
Mgmt TLS enabled: true
asa# show module 1 details
Getting details from the Service Module, please wait...
ASA 5585-X IPS Security Services Processor-20 with 8GE
Model: ASA5585-SSP-IPS20
Hardware version: 1.0
Serial Number: ABC1234DEFG
Firmware version: 2.0(7)0
Software version: 7.2(1)E4
MAC Address Range: 5475.d029.7f9c to 5475.d029.7fa7
App. name: IPS
App. Status: Up
App. Status Desc: Normal Operation
App. version: 7.2(1)E4
Data plane Status: Up
Status: Up
Mgmt IP addr: 192.0.2.3
Mgmt Network mask: 255.255.255.0
Mgmt Gateway: 192.0.2.254
Mgmt Access List: 0.0.0.0/0
Mgmt web ports: 443
Mgmt TLS enabled: true
asa#

```

If you have problems with reimaging the ASA 5585-X IPS SSP, use the **debug module-boot** command to see the output as it boots. Make sure you have the correct IP address for the TFTP server and you have the correct file on the TFTP server. Then use the **hw-module module 1 recover** command again to reimage the module.

```

ips-ssp# hw-module module 1 recover configure
Image URL [tftp://0.0.0.0/]: tftp://10.10.10.10//IPS-SSP_20-K9-sys-1.1-a-7.2-1-E4.img
Port IP Address [0.0.0.0]: 10.10.10.11
VLAN ID [0]:
Gateway IP Address [0.0.0.0]: 10.10.10.254

asa(config)# debug module-boot
debug module-boot enabled at level 1
asa(config)# hw-module module 1 recover boot
The module in slot 1 will be recovered. This may erase all configuration and all data on
that device and attempt to download a new image for it.
Recover module in slot 1? [confirm]
Recover issued for module in slot 1
asa(config)# Slot-1 140> Cisco Systems ROMMON Version (1.0(10)0) #0: Fri Mar 25 23:02:10
PST 2010
Slot-1 141> Platform ASA5585-SSP-IPS20
Slot-1 142> GigabitEthernet0/0
Slot-1 143> Link is UP
Slot-1 144> MAC Address: 000b.fcf8.0176
Slot-1 145> ROMMON Variable Settings:
Slot-1 146> ADDRESS=192.0.2.3
Slot-1 147> SERVER=192.0.2.15
Slot-1 148> GATEWAY=192.0.2.254
Slot-1 149> PORT=GigabitEthernet0/0
Slot-1 150> VLAN=untagged
Slot-1 151> IMAGE=IPS-SSP-K9-sys-1.1-a-7.2-1.img
Slot-1 152> CONFIG=
Slot-1 153> LINKTIMEOUT=20
Slot-1 154> PKTTIMEOUT=4
Slot-1 155> RETRY=20
Slot-1 156> tftp IPS-SSP_10-K9-sys-1.1-a-7.2-1.img@192.0.2.15 via 192.0.2.254
Slot-1 157> TFTP failure: Packet verify failed after 20 retries

```

```

Slot-1 158> Rebooting due to Autoboot error ...
Slot-1 159> Rebooting...
Slot-1 160> Cisco Systems ROMMON Version (1.0(10)0) #0: Fri Mar 25 23:02:10 PST 2010
Slot-1 161> Platform ASA5585-SSP-IPS20
Slot-1 162> GigabitEthernet0/0
Slot-1 163> Link is UP
Slot-1 164> MAC Address: 000b.fcf8.0176
Slot-1 165> ROMMON Variable Settings:
Slot-1 166>   ADDRESS=192.0.2.3
Slot-1 167>   SERVER=192.0.2.15
Slot-1 168>   GATEWAY=192.0.2.254
Slot-1 169>   PORT=GigabitEthernet0/0
Slot-1 170>   VLAN=untagged
Slot-1 171>   IMAGE=IPS-SSP_10-K9-sys-1.1-a-7.2-1.img
Slot-1 172>   CONFIG=
Slot-1 173>   LINKTIMEOUT=20
Slot-1 174>   PKTIMEOUT=4
Slot-1 175>   RETRY=20
Slot-1 176> tftp IPS-SSP_10-K9-sys-1.1-a-7.2-1.img@192.0.2.15 via 192.0.2.254

```

Traffic Flow Stopped on IPS Switchports

Problem Traffic on any port located on the ASA 5585-X IPS SSP (1/x) no longer passes through the adaptive security appliance when the ASA 5585-X IPS SSP is reset or shut down. This affects all traffic through these ports regardless of whether or not the traffic would have been monitored by the IPS. The link on the ports will link down when the ASA 5585-X IPS SSP is reset or shut down.

Possible Cause Using the ports located on the ASA 5585-X IPS SSP (1/x), and resetting or shutting it down via any mechanism.

Solution Use the ports on the adaptive security appliance (0/x) instead because those ports do not lose their link when the ASA 5585-X IPS SSP is reset or shut down.

Failover Scenarios

The following failover scenarios apply to the ASA 5585-X in the event of configuration changes, signature/signature engine updates, service packs, and SensorApp crashes on the ASA 5585-X IPS SSP.

Single ASA 5585-X in Fail-Open Mode

- If the ASA is configured in fail-open mode for the ASA 5585-X IPS SSP, and the ASA 5585-X IPS SSP experiences a configuration change or signature/signature engine update, traffic is passed through the ASA without being inspected.
- If the ASA is configured in fail-open mode for the ASA 5585-X IPS SSP, and the ASA 5585-X IPS SSP experiences a SensorApp crash or a service pack upgrade, traffic is passed through the ASA without being inspected.

Single ASA 5585-X in Fail-Close Mode

- If the ASA is configured in fail-close mode for the ASA 5585-X IPS SSP, and the ASA 5585-X IPS SSP experiences a configuration change or a signature/signature engine update, traffic is stopped from passing through the ASA.

- If the ASA is configured in fail-close mode for the ASA 5585-X IPS SSP, and the ASA 5585-X IPS SSP experiences a SensorApp crash or a service pack upgrade, traffic is stopped from passing through the ASA.

Two ASA 5585-Xs in Fail-Open Mode

- If the ASAs are configured in fail-open mode and if the ASA 5585-X IPS SSP on the active ASA experiences a configuration change or a signature/signature engine update, traffic is still passed through the active ASA without being inspected. Failover is not triggered.
- If the ASAs are configured in fail-open mode, and if the ASA 5585-X IPS SSP on the active ASA experiences a SensorApp crash or a service pack upgrade, failover is triggered and traffic passes through the ASA 5585-X IPS SSP that was previously the standby ASA 5585-X IPS SSP.

Two ASA 5585-Xs in Fail-Close Mode

- If the ASAs are configured in fail-close mode, and if the ASA 5585-X IPS SSP on the active ASA experiences a configuration change or a signature/signature engine update, traffic is stopped from passing through the active ASA. No failover is triggered.
- If the ASAs are configured in fail-close mode, and if the ASA 5585-X IPS SSP on the active ASA experiences a SensorApp crash or a service pack upgrade, failover is triggered and traffic passes through the ASA 5585-X IPS SSP that was previously the standby for the ASA 5585-X IPS SSP.

Configuration Examples

Use the following configuration for the primary ASA:

```
interface GigabitEthernet0/7
  description LAN Failover Interface

failover
failover lan unit primary
failover lan interface folink GigabitEthernet0/7
failover interface ip folink 172.27.48.1 255.255.255.0 standby 172.27.48.2
```

Use the following configuration for the secondary ASA:

```
interface GigabitEthernet0/7
  description LAN Failover Interface

failover
failover lan unit secondary
failover lan interface folink GigabitEthernet0/7
failover interface ip folink 172.27.48.1 255.255.255.0 standby 172.27.48.2
```




Obtaining Software

This chapter provides information on obtaining the latest Cisco IPS software. It contains the following sections:

- [IPS 7.3 File List, page 20-1](#)
- [Obtaining Cisco IPS Software, page 20-1](#)
- [IPS Software Versioning, page 20-3](#)
- [Accessing IPS Documentation, page 20-7](#)
- [Cisco Security Intelligence Operations, page 20-7](#)

IPS 7.3 File List

The currently supported IPS 7.3(x) versions are 7.3(1)E4 and 7.3(2)E4.

For a list of the specific IPS filenames and the IPS versions that each sensor supports, refer to the Release Notes for your IPS version found at this URL:

http://www.cisco.com/en/US/products/hw/vpndevc/ps4077/prod_release_notes_list.html

Obtaining Cisco IPS Software

You can find major and minor updates, service packs, signature and signature engine updates, system and recovery files, firmware upgrades, and Readmes on the Download Software site on Cisco.com. Signature updates are posted to Cisco.com approximately every week, more often if needed. Service packs are posted to Cisco.com in a release train format, a new release every three months. Major and minor updates are also posted periodically. Check Cisco.com regularly for the latest IPS software.

You must have an account with cryptographic access before you can download software. You set this account up the first time you download IPS software from the Download Software site.



Note

You must be logged in to Cisco.com to download software. You must have an active IPS maintenance contract and a Cisco.com password to download software. You must have a sensor license to apply signature updates.

Downloading Cisco IPS Software

To download software on Cisco.com, follow these steps:

-
- Step 1** Log in to Cisco.com.
 - Step 2** From the Support drop-down menu, choose **Download Software**.
 - Step 3** Under Select a Software Product Category, choose **Security Software**.
 - Step 4** Choose **Intrusion Prevention System (IPS)**.
 - Step 5** Enter your username and password.
 - Step 6** In the Download Software window, choose **IPS Appliances > Cisco Intrusion Prevention System** and then click the version you want to download.



Note You must have an IPS subscription service license to download software.

- Step 7** Click the type of software file you need. The available files appear in a list in the right side of the window. You can sort by file name, file size, memory, and release date. And you can access the Release Notes and other product documentation.
 - Step 8** Click the file you want to download. The file details appear.
 - Step 9** Verify that it is the correct file, and click **Download**.
 - Step 10** Click **Agree** to accept the software download rules. The File Download dialog box appears. The first time you download a file from Cisco.com, you must fill in the Encryption Software Export Distribution Authorization form before you can download the software.
 - a.** Fill out the form and click **Submit**. The Cisco Systems Inc. Encryption Software Usage Handling and Distribution Policy appears.
 - b.** Read the policy and click **I Accept**. The Encryption Software Export/Distribution Form appears.If you previously filled out the Encryption Software Export Distribution Authorization form, and read and accepted the Cisco Systems Inc. Encryption Software Usage Handling and Distribution Policy, these forms are not displayed again.
 - Step 11** Open the file or save it to your computer.
 - Step 12** Follow the instructions in the Readme or the Release Notes to install the update.
-

For More Information

For an explanation of the IPS file versioning scheme, see [IPS Software Versioning, page 20-3](#).

IPS Software Versioning

When you download IPS software images from Cisco.com, you should understand the versioning scheme so that you know which files are base files, which are cumulative, and which are incremental.

Major Update

A major update contains new functionality or an architectural change in the product. For example, the Cisco IPS 7.3 base version includes everything (except deprecated features) since the previous major release (the minor update features, service pack fixes, and signature updates) plus any new changes. Major update 7.3(1) requires 5.1(6) and later. With each major update there are corresponding system and recovery packages.



Note

The 7.3(1) major update is used to upgrade 5.1(6) and later sensors to 7.3(1). If you are reinstalling 7.3(1) on a sensor that already has 7.3(1) installed, use the system image or recovery procedures rather than the major update.

Minor Update

A minor update is incremental to the major version. Minor updates are also base versions for service packs. The first minor update for 7.3 is 7.4. Minor updates are released for minor enhancements to the product. Minor updates contain all previous minor features (except deprecated features), service pack fixes, signature updates since the last major version, and the new minor features being released. You can install the minor updates on the previous major or minor version (and often even on earlier versions). The minimum supported version needed to upgrade to the newest minor version is listed in the Readme that accompanies the minor update. With each minor update there are corresponding system and recovery packages.

Service Pack

A service pack is cumulative following a base version release (minor or major). Service packs are released in a train release format with several new features per train. Service packs contain all service pack fixes since the last base version (minor or major) and the new features and defect fixes being released. Service packs require the minor version. The minimum supported version needed to upgrade to the newest service pack is listed in the Readme that accompanies the service pack. Service packs also include the latest engine update. For example, if service pack 7.3(3) is released, and E4 is the latest engine level, the service pack is released as 7.3(3)E4.

Patch Release

A patch release is used to address defects that are identified in the upgrade binaries after a software release. Rather than waiting until the next major or minor update, or service pack to address these defects, a patch can be posted. Patches include all prior patch releases within the associated service pack level. The patches roll into the next official major or minor update, or service pack.

Before you can install a patch release, the most recent major or minor update, or service pack must be installed. For example, patch release 7.3(1p1) requires 7.3(1).

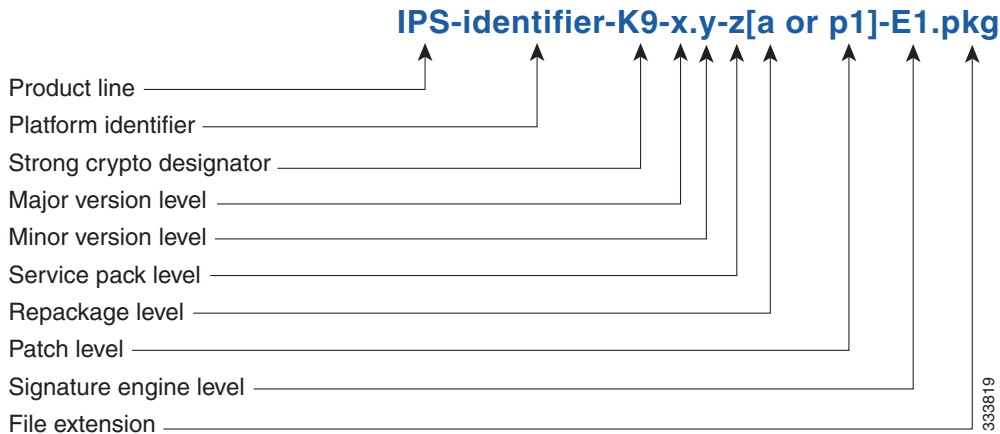


Note

Upgrading to a newer patch does not require you to uninstall the old patch. For example, you can upgrade from patch 7.3(1p1) to 7.3(1p2) without first uninstalling 7.3(1p1).

Figure 20-1 illustrates what each part of the IPS software file represents for major and minor updates, service packs, and patch releases.

Figure 20-1 IPS Software File Name for Major and Minor Updates, Service Packs, and Patch Releases



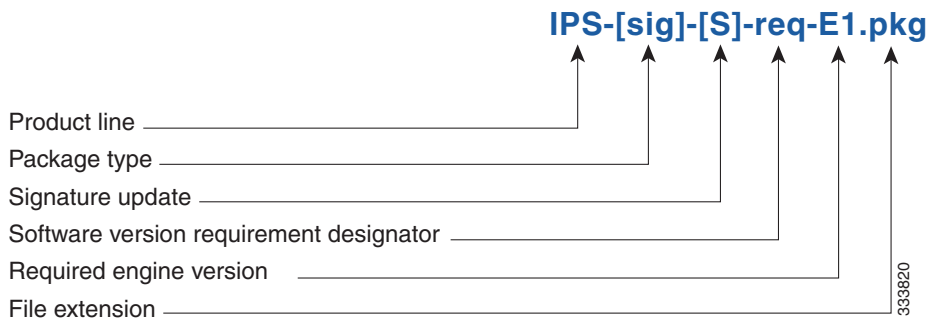
Signature Update and Signature Engine Update

A signature update is a package file containing a set of rules designed to recognize malicious network activities. Signature updates are released independently from other software updates. Each time a major or minor update is released, you can install signature updates on the new version and the next oldest version for a period of at least six months. Signature updates are dependent on a required signature engine version. Because of this, a *req* designator lists the signature engine required to support a particular signature update. Signature updates also contain the latest threat profile updates. If there is a new threat profile, it will be installed along with the signature update.

The signature engine update is contained in the signature updates.

Figure 20-2 illustrates what each part of the IPS software file represents for signature updates and signature engine updates.

Figure 20-2 IPS Software File Name for Signature Updates and Signature Engine Updates



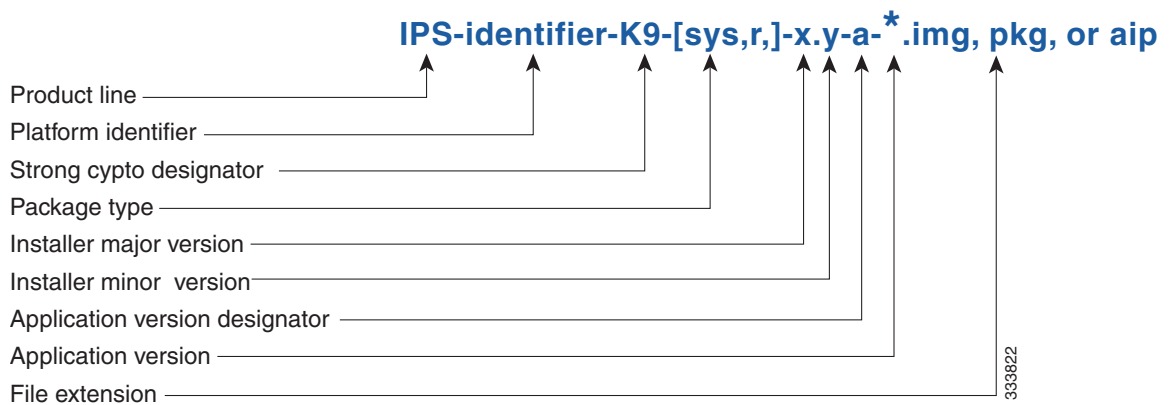
Recovery and System Image Files

Recovery and system image files contain separate versions for the installer and the underlying application. The installer version contains a major and minor version field. The major version is incremented by one of any major changes to the image installer, for example, switching from .tar to rpm or changing kernels. The minor version can be incremented by any one of the following:

- Minor change to the installer, for example, a user prompt added.
- Repackages require the installer minor version to be incremented by one if the image file must be repackaged to address a defect or problem with the installer.

Figure 20-3 illustrates what each part of the IPS software file represents for recovery and system image filenames.

Figure 20-3 IPS Software File Name for Recovery and System Image Files



IPS Software Release Examples

Table 20-1 lists the Cisco IPS software release examples.

Table 20-1 IPS Software Release Examples

Release	Target Frequency	Identifier	Example Version	Example Filename
Signature update and signature engine update ¹	Weekly for signature updates, as needed for signature engine updates	sig	S669 E4	IPS-sig-S669-req-E4.pkg
Service packs ²	Every three months	—	7.3(2)	IPS-identifier-K9-7.3-2-E4.pkg
Minor version update ³	Annually	—	7.3(1)	IPS-identifier-K9-7.3-1-E4.pkg
Major version update ⁴	Annually	—	8.0(1)	IPS-identifier-K9-8.0-1-E4.pkg
Patch release ⁵	As needed	patch	7.3(1p1)	IPS-identifier-K9-patch-7.3-1pl-E4.pkg

Table 20-1 IPS Software Release Examples (continued)

Release	Target Frequency	Identifier	Example Version	Example Filename
Recovery package ⁶	Annually or as needed	r	1.1-7.3(1)	IPS-identifier-K9-r-1.1-a-7.3-1-E4.pkg
System image ⁷	Annually	sys	Separate file per sensor platform	IPS-identifier-K9-sys-1.1-a-7.3-2-E4.img IPS-identifier-K9-sys-1.1-a-7.3-2-E4.aip

- Signature updates include the latest cumulative IPS signatures and threat profiles. Signature engine updates add new engines or engine parameters that are used by new signatures in later signature updates.
- Service packs include new features and defect fixes.
- Minor versions include new minor version features and/or minor version functionality.
- Major versions include new major version functionality or new architecture.
- Patch releases are for interim fixes.
- The r 1.1 can be revised to r 1.2 if it is necessary to release a new recovery package that contains the same underlying application image. If there are defect fixes for the installer, for example, the underlying application version may still be 7.3(1), but the recovery partition image will be r 1.2.
- The system image includes the combined recovery and application image used to reimage an entire sensor.

Table 20-2 describes the platform identifiers used in filenames.

Table 20-2 Platform Identifiers

Sensor Family	Identifier
ASA 5500-X series	SSP_5512 SSP_5515 SSP_5525 SSP_5545 SSP_5555
ASA 5585-X series	SSP_10 SSP_20 SSP_40 SSP_60
IPS 4345 series	4345
IPS 4360 series	4360
IPS 4510 series	4510
IPS 4520 series	4520

For More Information

For instructions on how to access these files on Cisco.com, see [Obtaining Cisco IPS Software, page 20-1](#).

Accessing IPS Documentation

You can find IPS documentation at this URL:

http://www.cisco.com/en/US/products/hw/vpndevc/ps4077/tsd_products_support_series_home.html

Or to access IPS documentation from Cisco.com, follow these steps:

-
- Step 1** Log in to [Cisco.com](http://www.cisco.com).
- Step 2** Click **Support**.
- Step 3** Under Support at the bottom of the page, click **Documentation**.
- Step 4** Choose **Products > Security > Intrusion Prevention System (IPS) > IPS Appliances > Cisco IPS 4200 Series Sensors**. The Cisco IPS 4200 Series Sensors page appears. All of the most up-to-date IPS documentation is on this page.



Note Although you will see references to other IPS documentation sites on Cisco.com, this is the site with the most complete and up-to-date IPS documentation.

- Step 5** Click one of the following categories to access Cisco IPS documentation:
- **Download Software**—Takes you to the Download Software site.



Note You must be logged into Cisco.com to access the software download site.

- **Release and General Information**—Contains documentation roadmaps and release notes.
 - **Reference Guides**—Contains command references and technical references.
 - **Design**—Contains design guide and design tech notes.
 - **Install and Upgrade**—Contains hardware installation and regulatory guides.
 - **Configure**—Contains configuration guides for IPS CLI, IDM, and IME.
 - **Troubleshoot and Alerts**—Contains TAC tech notes and field notices.
-

Cisco Security Intelligence Operations

The Cisco Security Intelligence Operations site on Cisco.com provides intelligence reports about current vulnerabilities and security threats. It also has reports on other security topics that help you protect your network and deploy your security systems to reduce organizational risk.

You should be aware of the most recent security threats so that you can most effectively secure and manage your network. Cisco Security Intelligence Operations contains the top ten intelligence reports listed by date, severity, urgency, and whether there is a new signature available to deal with the threat.

Cisco Security Intelligence Operations contains a Security News section that lists security articles of interest. There are related security tools and links.

You can access Cisco Security Intelligence Operations at this URL:

<http://tools.cisco.com/security/center/home.x>

Cisco Security Intelligence Operations is also a repository of information for individual signatures, including signature ID, type, structure, and description.

You can search for security alerts and signatures at this URL:

<http://tools.cisco.com/security/center/search.x>



Upgrading, Downgrading, and Installing System Images

This chapter describes how to upgrade, downgrade, and install system images. It contains the following sections:

- [Upgrade Notes and Caveats, page 21-1](#)
- [Upgrades, Downgrades, and System Images, page 21-2](#)
- [Supported FTP and HTTP/HTTPS Servers, page 21-3](#)
- [Upgrading the Sensor, page 21-3](#)
- [Configuring Automatic Upgrades, page 21-7](#)
- [Downgrading the Sensor, page 21-12](#)
- [Recovering the Application Partition, page 21-14](#)
- [Installing System Images, page 21-15](#)

Upgrade Notes and Caveats

Pay attention to the following upgrade notes and caveats when upgrading your sensor:

- Anomaly detection has been disabled by default. If you did not configure the operation mode manually before the upgrade, it defaults to inactive after you upgrade. If you configured the operation mode to detect, learn, or inactive, the tuned value is preserved after the upgrade.
- You must have a valid maintenance contract per sensor to download software upgrades from Cisco.com.
- You must be running the following versions to upgrade the following platforms to IPS 7.3(1)E4:
 - For the IPS 4300 series sensors, you must be running IPS 7.1(3)E4 or later.
 - For the IPS 4500 series sensors, you must be running IPS 7.1(4)E4 or later.
- This service pack automatically reboots the sensor to apply the changes. During reboot, inline network traffic is disrupted.
- You cannot uninstall IPS 7.3(1)E4. To revert to a previous version, you must reimage the sensor using the appropriate system image file.
- You cannot use the **downgrade** command to revert to a previous major or minor version, for example, from Cisco IPS 7.3 to 7.2. You can only use the **downgrade** command to downgrade from the latest signature update or signature engine update. To revert to 7.2, you must reimage the sensor.

- All user configuration settings are lost when you install the system image. Before trying to recover the sensor by installing the system image, try to recover by using the **recover application-partition** command or by selecting the recovery partition during sensor bootup.

For More Information

- For the procedure for accessing downloads on Cisco.com, see [Obtaining Cisco IPS Software, page 20-1](#).
- For the procedure for using the **upgrade** command to upgrade the sensor, see [Upgrading the Sensor, page 21-3](#).
- For the procedure for configuring automatic upgrades on the sensor, see [Configuring Automatic Upgrades, page 21-7](#).
- For the procedure for using the **recover** command, see [Recovering the Application Partition, page 21-14](#).
- For the procedure for installing the IPS 4345 and IPS 4360 system image, see [Installing the System Image for the IPS 4300 Series Sensors, page 21-16](#).
- For the procedure for installing the IPS 4510 and IPS 4520 system image, see [Installing the System Image for the IPS 4500 Series Sensors, page 21-20](#).
- For the procedure for installing the ASA 5500-X IPS SSP system image, see [Installing the System Image for the ASA 5500-X IPS SSP, page 21-22](#).
- For the procedure for installing the ASA 5585-X IPS SSP system image, see [Installing the System Image for the ASA 5585-X IPS SSP, page 21-23](#).

Upgrades, Downgrades, and System Images

You can upgrade and downgrade the software on the sensor. Upgrading applies a service pack, signature update (which includes threat profile updates), signature engine update, minor version, major version, or recovery partition file. Downgrading removes the last applied service pack or signature update from the sensor. With a signature downgrade, the threat profile is downgraded to the version available in the previous signature version.

You can recover the application partition image on your sensor if it becomes unusable. Using the **recover** command lets you retain your host settings while other settings revert to the factory defaults.

When you install a new system image on your sensor, all accounts are removed and the default cisco account is reset to use the default password **cisco**. After installing the system image, you must initialize the sensor again.

After you reimage and initialize your sensor, upgrade your sensor with the most recent service pack, signature update, signature engine update, minor update, major update, and recovery partition files.

Pay attention to the following:

- You cannot use the **downgrade** command to revert to a previous major or minor version, for example, from Cisco IPS 7.3 to 7.2. You can only use the **downgrade** command to downgrade from the latest signature update or signature engine update. To revert to 7.2, you must reimage the sensor.
- After you upgrade any IPS software on your sensor, you must restart the IDM to see the latest software features.
- You cannot downgrade the sensor using the recovery partition. To downgrade to an earlier version, you must install the appropriate system image file (.img file).

- During a signature upgrade all signature configurations are retained, both the signature tunings as well as the custom signatures. During a signature downgrade the current signature configuration is replaced with the old signature configuration. So if the last signature set had custom signatures and/or signature tunings, these are restored during the downgrade. Threat profiles may also be upgraded during signature upgrades.

For More Information

- For the procedure for initializing the sensor, see [Basic Sensor Setup, page 3-5](#).
- For the procedure for locating software on Cisco.com, see [Obtaining Cisco IPS Software, page 20-1](#).

Supported FTP and HTTP/HTTPS Servers

The following FTP servers are supported for IPS software updates:

- WU-FTPD 2.6.2 (Linux)
- Solaris 2.8
- Sambar 6.0 (Windows 2000)
- Serv-U 5.0 (Windows 2000)
- MS IIS 5.0 (Windows 2000)

The following HTTP/HTTPS servers are supported for IPS software updates:

- CSM - Apache Server (Tomcat)
- CSM - Apache Server (JRun)

For More Information

- For the procedure for downloading IPS software updates from Cisco.com, see [Obtaining Cisco IPS Software, page 20-1](#).
- For the procedure for configuring automatic updates, see [Configuring Automatic Upgrades, page 21-7](#).

Upgrading the Sensor

This section explains how to use the **upgrade** command to upgrade the software on the sensor. It contains the following topics:

- [IPS 7.3\(1\)E4 Files, page 21-3](#)
- [Manually Upgrading the Sensor, page 21-4](#)
- [Upgrading the Recovery Partition, page 21-6](#)

IPS 7.3(1)E4 Files

The currently supported IPS 7.3(x) versions are 7.3(1)E4 and 7.3(2)E4.

For a list of the specific IPS filenames and the IPS versions that each sensor supports, refer to the Release Notes for your IPS version found at this URL:

http://www.cisco.com/en/US/products/hw/vpndevc/ps4077/prod_release_notes_list.html

For More Information

For the procedure for obtaining these files on Cisco.com, see [Obtaining Cisco IPS Software, page 20-1](#).

Manually Upgrading the Sensor

**Caution**

You must log in to Cisco.com using an account with cryptographic privileges to download software. The first time you download software on Cisco.com, you receive instructions for setting up an account with cryptographic privileges.

**Note**

Do not change the filename. You must preserve the original filename for the sensor to accept the update.

**Note**

During a signature upgrade all signature configurations are retained, both the signature tunings as well as the custom signatures. During a signature downgrade the current signature configuration is replaced with the old signature configuration. So if the last signature set had custom signatures and/or signature tunings, these are restored during the downgrade. Threat profiles may also be upgraded during signature upgrades.

Use the **upgrade** *source-url* command to apply service pack, signature update, threat profile update, engine update, minor version, major version, or recovery partition file upgrades. The following commands apply:

- *source-url*—Specifies the location of the source file to be copied:
 - ftp:—Source URL for an FTP network server. The syntax for this prefix is:
ftp://[[username@]location][relativeDirectory]/filename
ftp://[[username@]location][absoluteDirectory]/filename



Note You are prompted for a password.

- scp:—Source URL for the SCP network server. The syntax for this prefix is:
scp://[[username@]location][relativeDirectory]/filename
scp://[[username@]location][absoluteDirectory]/filename



Note You are prompted for a password. You must add the remote host to the SSH known hosts list.

- http:—Source URL for the web server. The syntax for this prefix is:
http://[[username@]location][directory]/filename



Note The directory specification should be an absolute path to the desired file.

- https:—Source URL for the web server. The syntax for this prefix is:
https://[[username@]location][directory]/filename



Note The directory specification should be an absolute path to the desired file. The remote host must be a TLS trusted host.

Upgrading the Sensor



Note The CLI output is an example of what your configuration may look like. It will not match exactly due to the optional setup choices, sensor model, and IPS version you have installed.

To upgrade the sensor, follow these steps:

- Step 1** Download the appropriate file to an FTP, SCP, HTTP, or HTTPS server that is accessible from your sensor.
- Step 2** Log in to the CLI using an account with administrator privileges.
- Step 3** Enter configuration mode.

```
sensor# configure terminal
```

- Step 4** Upgrade the sensor.

```
sensor(config)# upgrade url/IPS-4360-K9-7.3-1-E4.pkg
```

The URL points to where the update file is located, for example, to retrieve the update using FTP, enter the following:

```
sensor(config)# upgrade ftp://username@ip_address//directory/IPS-4360-K9-7.3-1-E4.pkg
```

- Step 5** Enter the password when prompted.

```
Enter password: *****
```

- Step 6** Enter **yes** to complete the upgrade.



Note Major updates, minor updates, and service packs may force a restart of the IPS processes or even force a reboot of the sensor to complete installation.



Note The operating system is reimaged and all files that have been placed on the sensor through the service account are removed.

- Step 7** Verify your new sensor version.

```
sensor# show version
```

```
Application Partition:
```

```
Cisco Intrusion Prevention System, Version 7.3(1)E4
```

```
Host:
```

```
  Realm Keys                key1.0
```

```
Signature Definition:
```

```
  Signature Update          S741.0          2013-09-10
```

```

    Threat Profile Version    2
OS Version:                 2.6.29.1
Platform:                   IPS-4360
Serial Number:              FGL1702401M
Licensed, expires:         21-Nov-2014 UTC
Sensor up-time is 22:09.
Using 14372M out of 15943M bytes of available memory (90% usage)
system is using 32.4M out of 160.0M bytes of available disk space (20% usage)
application-data is using 85.6M out of 376.4M bytes of available disk space (24% usage)
boot is using 63.1M out of 70.2M bytes of available disk space (95% usage)
application-log is using 494.0M out of 513.0M bytes of available disk space (96% usage)

MainApp                      C-2013_12_16_14_00_7_3_0_143    (Release)    2013-12-16T14:06:20-0600
Running
AnalysisEngine              C-2013_12_16_14_00_7_3_0_143    (Release)    2013-12-16T14:06:20-0600
Running
CollaborationApp           C-2013_12_16_14_00_7_3_0_143    (Release)    2013-12-16T14:06:20-0600
Running
CLI                          C-2013_12_16_14_00_7_3_0_143    (Release)    2013-12-16T14:06:20-0600

Upgrade History:

    IPS-K9-7.3-1-E4    11:22:07 UTC Sat Jan 19 2013

Recovery Partition Version 1.1 - 7.3(1)E4

Host Certificate Valid from: 09-Oct-2014 to 09-Oct-2016

sensor#

```

For More Information

- For a list of supported FTP and HTTP/HTTPS servers, see [Supported FTP and HTTP/HTTPS Servers, page 21-3](#).
- For a list of the specific upgrade files, see [IPS 7.3\(1\)E4 Files, page 21-3](#).
- For the procedure for locating software on Cisco and obtaining an account with cryptographic privileges, see [Obtaining Cisco IPS Software, page 20-1](#).
- For the IDM procedure for upgrading the sensor, refer to [Manually Updating the Sensor](#). For the IME procedure, refer to [Manually Updating the Sensor](#).

Upgrading the Recovery Partition

**Note**

You cannot downgrade the sensor using the recovery partition. To downgrade to an earlier version, you must install the appropriate system image file (.img file).

Use the **upgrade** command to upgrade the recovery partition with the most recent version so that it is ready if you need to recover the application partition on your sensor. Recovery partition images are generated for major and minor updates and only in rare situations for service packs or signature updates.

To upgrade the recovery partition on your sensor, follow these steps:

Step 1 Download the appropriate recovery partition image file to an FTP, SCP, HTTP, or HTTPS server that is accessible from your sensor.



Caution

Some browsers add an extension to the filename. The filename of the saved file must match what is displayed on the download page or you cannot use it to upgrade the recovery partition.

Step 2 Log in to the CLI using an account with administrator privileges.

Step 3 Enter configuration mode.

```
sensor# configure terminal
```

Step 4 Upgrade the recovery partition.

```
sensor(config)#  
upgrade scp://user@server_ipaddress//upgrade_path/IPS-4360-K9-r-1.1-a-7.3-1-E4.pkg  
  
sensor(config)#  
upgrade ftp://user@server_ipaddress//upgrade_path/IPS-4360-K9-r-1.1-a-7.3-1-E4.pkg
```

Step 5 Enter the server password. The upgrade process begins.



Note

This procedure only reimages the recovery partition. The application partition is not modified by this upgrade. To reimage the application partition after the recovery partition, use the **recover application-partition** command.

For More Information

- For a list of supported FTP and HTTP/HTTPS servers, see [Supported FTP and HTTP/HTTPS Servers, page 21-3](#).
- For a list of the specific recovery filenames, see [IPS 7.3\(1\)E4 Files, page 21-3](#).
- For the procedure for locating software on Cisco.com, see [Obtaining Cisco IPS Software, page 20-1](#).
- For the procedure for using the **recover** command, see [Recovering the Application Partition Image, page 21-14](#).

Configuring Automatic Upgrades

This section describes how to configure automatic updates and how to perform an immediate update, and contains the following topics:

- [Configuring Automatic Updates, page 21-8](#)
- [Applying an Immediate Update, page 21-11](#)

Configuring Automatic Updates


Note

For the IDM procedure for automatically upgrading the sensor, refer to [Configuring Automatic Update](#). For the IME procedure, refer to [Configuring Automatic Update](#).


Note

Support for either a proxy or DNS server to resolve the host name and download updates from Cisco.com is available. You configure these servers in service host network-settings mode.

You can configure the sensor to look for new upgrade files in your upgrade directory automatically. For example, several sensors can point to the same remote FTP server directory with different update schedules, such as every 24 hours, or Monday, Wednesday, and Friday at 11:00 pm.

You specify the following information to schedule automatic upgrades:

- Server IP address
- Path of the directory on the file server where the sensor checks for upgrade files
- File copy protocol (SCP or FTP)
- Username and password
- Upgrade schedule

You must download the software upgrade from Cisco.com and copy it to the upgrade directory before the sensor can poll for automatic upgrades.

Use the **auto-upgrade-option enabled** command in the service host mode to configure automatic upgrades.

The following commands apply:

- **cisco-server {disabled | enabled}**—Enables automatic signature and engine updates from Cisco.com.
- **cisco-url** *cisco_url*—Specifies the Cisco server locator service. You do not need to change this unless the www.cisco.com IP address changes.
- **default**— Sets the value back to the system default setting.
- **directory** *directory*— Specifies the directory where upgrade files are located on the file server. A leading '/' indicates an absolute path.
- **file-copy-protocol {ftp | scp}**— Specifies the file copy protocol used to download files from the file server.


Note

If you use SCP, you must use the **ssh host-key** command to add the server to the SSH known hosts list so the sensor can communicate with it through SSH.

- **ip-address** *ip_address*—Specifies the IP address of the file server.
- **password** *password*—Specifies the user password for Cisco server authentication.

- **schedule-option**—Specifies the schedules for when Cisco server automatic upgrades occur. Calendar scheduling starts upgrades at specific times on specific days. Periodic scheduling starts upgrades at specific periodic intervals.
 - **calendar-schedule**—Configures the days of the week and times of day that automatic upgrades will be performed.
 - **days-of-week**—Specifies the days of the week on which auto-upgrades will be performed. You can select multiple days: *sunday* through *saturday* are the valid values.
 - **no**—Removes an entry or selection setting.
 - **times-of-day**—Specifies the times of day at which auto-upgrades will begin. You can select multiple times. The valid value is *hh:mm[:ss]*.
 - **periodic-schedule**—Specifies the time that the first automatic upgrade should occur, and how long to wait between automatic upgrades.
 - **interval**—Specifies the number of hours to wait between automatic upgrades. Valid values are 1 to 8760. The default value is 24.
 - **start-time**—Specifies the time of day to start the first automatic upgrade. The valid value is *hh:mm[:ss]*.
- **user-name** *user_name*—Specifies the username for server authentication.
- **user-server {disabled | enabled}**—Enables automatic upgrades from a user-defined server.

Configuring Automatic Upgrades

If you get an unauthorized error message while configuring an automatic update, make sure you have the correct ports open on any firewalls between the sensor and Cisco.com. For example, you need port 443 for the initial automatic update connection to www.cisco.com, and you need port 80 to download the chosen package from a Cisco file server. The IP address may change for the Cisco file server, but you can find it in the lastDownloadAttempt section in the output of the **show statistics host** command.



Note

To check the status of the last automatic update or the next scheduled automatic update, run the **show statistics host** command and check the Auto Update Statistics section.

To schedule automatic upgrades, follow these steps:

Step 1 Log in to the CLI using an account with administrator privileges.

Step 2 Enter automatic upgrade submode.

```
sensor# configure terminal
sensor(config)# service host
sensor(config-hos)# auto-upgrade
sensor(config-hos-aut)#
```

Step 3 Configure the sensor to automatically look for new upgrades either on Cisco.com or on your file server:

- a. On Cisco.com. Continue with Step 4.

```
sensor(config-hos-aut)# cisco-server enabled
```

- b. From your server.

```
sensor(config-hos-aut)# user-server enabled
```

- c. Specify the IP address of the file server.

```
sensor(config-hos-ena)# ip-address 10.1.1.1
```

- d. Specify the directory where the upgrade files are located on the file server.

```
sensor(config-hos-ena)# directory /tftpboot/sensor_updates
```

- e. Specify the file server protocol.

```
sensor(config-hos-ena)# file-copy-protocol ftp
```



Note If you use SCP, you must use the `ssh host-key` command to add the server to the SSH known hosts list so the sensor can communicate with it through SSH.

- Step 4** Specify the username for authentication.

```
sensor(config-hos-ena)# user-name tester
```

- Step 5** Specify the password of the user.

```
sensor(config-hos-ena)# password
Enter password[:] : *****
Re-enter password: *****
```

- Step 6** Specify the scheduling:

- a. For calendar scheduling (starts upgrades at specific times on specific day):

```
sensor(config-hos-ena)# schedule-option calendar-schedule
sensor(config-hos-ena-cal)# days-of-week sunday
sensor(config-hos-ena-cal)# times-of-day 12:00:00
```

- b. For periodic scheduling (starts upgrades at specific periodic intervals):

```
sensor(config-hos-ena)# schedule-option periodic-schedule
sensor(config-hos-ena-per)# interval 24
sensor(config-hos-ena-per)# start-time 13:00:00
```

- Step 7** Verify the settings.

```
sensor(config-hos-ena)# show settings
enabled
-----
schedule-option
-----
periodic-schedule
-----
start-time: 13:00:00
interval: 24 hours
-----
-----
ip-address: 10.1.1.1
directory: /tftpboot/update/6.1_dummy_updates
user-name: tester
password: <hidden>
file-copy-protocol: ftp default: scp
-----
sensor(config-hos-ena)#
```

- Step 8** Exit automatic upgrade submode.

```
sensor(config-hos-ena)# exit
```

```
sensor(config-hos)# exit
Apply Changes:[yes]:
```

Step 9 Press **Enter** to apply the changes or type **no** to discard them.

For More Information

- For a list of supported FTP and HTTP/HTTPS servers, see [Supported FTP and HTTP/HTTPS Servers, page 21-3](#).
- For the CLI procedure for adding the SCP server to the SSH known hosts list, see [Adding Hosts to the SSH Known Hosts List, page 4-47](#).
- For the procedure for configuring proxy and DNS servers, see [Configuring the DNS and Proxy Servers for Global Correlation and Automatic Update, page 4-10](#).
- For the output of the **show statistics host** command, see [Displaying Statistics, page 17-30](#).
- For the IDM procedure for automatically upgrading the sensor, refer to [Configuring Automatic Update](#). For the IME procedure, refer to [Configuring Automatic Update](#).

Applying an Immediate Update



Note

You can use the IDM, IME, and CLI immediately after you begin an automatic update because the automatic update is now executed as background process.



Note

You must have either a DNS or HTTP proxy server configured to download automatic updates from cisco.com.



Note

You must have automatic update configured and a valid license to apply updates.

Use the **autoupdatenow** command to perform an immediate update on the sensor. You receive a warning that this command performs an update on the sensor immediately. After executing this command, disable the **user-server/cisco-server** options in the auto-upgrade settings in the service host submode, if you do not want scheduled automatic updates.

To perform an immediate update on the sensor, follow these steps:

Step 1 Log in to the sensor using an account with administrator privileges.

Step 2 Start immediate automatic update.

```
sensor# autoupdatenow
Warning: Executing this command will perform an auto-upgrade on the sensor immediately.
Before executing this command, you must have a valid license to apply the Signature
AutoUpdates and auto-upgrade settings configured.After executing this command please
disable user-server/cisco-server inside 'auto-upgrade' settings, if you don't want
scheduled auto-updates
Continue? []:
```

Step 3 Enter **yes** to continue. The update is applied.

Step 4 Check to see if the update was applied.

```
sensor# show statistics host
Auto Update Statistics
  lastDirectoryReadAttempt = N/A
  lastDownloadAttempt = N/A
  lastInstallAttempt = N/A
  nextAttempt = N/A
```

For More Information

For the procedure for configuring automatic update, see [Configuring Automatic Updates, page 21-8](#).

Downgrading the Sensor



Caution

You cannot use the **downgrade** command to revert to a previous major or minor version, for example, from Cisco IPS 7.3 to 7.2. You can only use the **downgrade** command to downgrade from the latest signature update or signature engine update. To revert to 7.2, you must reimagine the sensor.



Note

You cannot downgrade the sensor using the recovery partition. To downgrade to an earlier version, you must install the appropriate system image file (.img file).

Use the **downgrade** command to remove the last applied signature upgrade or signature engine upgrade from the sensor. The signature upgrade includes threat profiles. When you downgrade the signature update, the threat profile reverts to the one found in the previous signature upgrade.

To remove the last applied signature update or signature engine update from the sensor, follow these steps:

Step 1 Log in to the sensor using an account with administrator privileges.

Step 2 Enter global configuration mode.

```
sensor# configure terminal
```

Step 3 Downgrade the sensor.

```
sensor(config)# downgrade
Warning: Executing this command will downgrade the system to IPS-sig-S741-req-E4.
Configuration changes made since the last upgrade will be lost and the system may be
rebooted. Signature threat profile mapping to signature instances will be reverted to the
previous configuration.
Continue with downgrade? []: y
% Please answer `yes` or `no`.
Continue with downgrade? []: yes

Broadcast Message from root@qa-ff-4360-188-85
(somewhere) at 19:23 ...

Un-installing IPS-sig-S741-req-E4.
```

```
Broadcast Message from root@qa-ff-4360-188-85
(somewhere) at 19:23 ...
```

```
Un-install complete.
```

```
sensor#
```

Step 4 Display the current version after the downgrade.

```
sensor# show version
Application Partition:

Cisco Intrusion Prevention System, Version 7.3(1)E4

Host:
  Realm Keys          key1.0
Signature Definition:
  Signature Update    S741.0          2013-09-10
  Threat Profile Version 1
OS Version:          2.6.29.1
Platform:            IPS-4360
Serial Number:       FGL1702401M
Licensed, expires:   21-Nov-2014 UTC
Sensor up-time is 22:09.
Using 14372M out of 15943M bytes of available memory (90% usage)
system is using 32.4M out of 160.0M bytes of available disk space (20% usage)
application-data is using 85.6M out of 376.4M bytes of available disk space (24% usage)
boot is using 63.1M out of 70.2M bytes of available disk space (95% usage)
application-log is using 494.0M out of 513.0M bytes of available disk space (96% usage)

MainApp              C-2013_12_16_14_00_7_3_0_143  (Release)  2013-12-16T14:06:20-0600
Running
AnalysisEngine       C-2013_12_16_14_00_7_3_0_143  (Release)  2013-12-16T14:06:20-0600
Running
CollaborationApp     C-2013_12_16_14_00_7_3_0_143  (Release)  2013-12-16T14:06:20-0600
Running
CLI                   C-2013_12_16_14_00_7_3_0_143  (Release)  2013-12-16T14:06:20-0600

Upgrade History:

* IPS-K9-7.3-1-E4          07:40:07 UTC Sun Jan 20 2013
  IPS-sig-S769-req-E4.pkg  16:57:01 UTC Sat Jan 11 2014

Recovery Partition Version 1.1 - 7.3(1)E4

Host Certificate Valid from: 09-Oct-2014 to 09-Oct-2016

sensor#
```

Step 5 If there is no recently applied service pack or signature update, the **downgrade** command is not available.

```
sensor(config)# downgrade
No downgrade available.
sensor(config)#
```

Recovering the Application Partition

You can recover the application partition image for the sensor if it becomes unusable. Some network configuration information is retained when you use this method, which lets you have network access after the recovery is performed. Use the **recover application-partition** command to boot to the recovery partition, which automatically recovers the application partition on your sensor. If you have upgraded your recovery partition to the most recent version before you recover the application partition image, you can install the most up-to-date software image.

Because you can execute the **recover application-partition** command through a Telnet or SSH connection, we recommend using this command to recover sensors that are installed at remote locations.



Note

When you reconnect to the sensor after recovery, you must log in with the default username and password **cisco**.

Recovering the Application Partition Image

To recover the application partition image, follow these steps:

-
- Step 1** Download the recovery partition image file to an FTP, HTTP, or HTTPS server that is accessible from your sensor.
- Step 2** Log in to the CLI using an account with administrator privileges.
- Step 3** Enter configuration mode.
- ```
sensor# configure terminal
```
- Step 4** Recover the application partition image.
- ```
sensor(config)# recover application-partition
Warning: Executing this command will stop all applications and re-image the node to
version 7.1(x)E4. All configuration changes except for network settings will be reset to
default.
Continue with recovery? []:
```
- Step 5** Enter **yes** to continue. Shutdown begins immediately after you execute the **recover** command. Shutdown can take a while, and you will still have access to the CLI, but access will be terminated without warning. The application partition is reimaged using the image stored on the recovery partition. You must now initialize the sensor with the **setup** command. The IP address, netmask, access lists, time zone, and offset are saved and applied to the reimaged application partition. If you executed the **recover application-partition** command remotely, you can SSH to the sensor with the default username and password (**cisco/cisco**) and then initialize the sensor again with the **setup** command. You cannot use Telnet until you initialize the sensor because Telnet is disabled by default.
-

For More Information

- For more information about TFTP servers, see [TFTP Servers, page 21-15](#).
- For the procedure for locating software on Cisco.com, see [Obtaining Cisco IPS Software, page 20-1](#).
- For the procedure for using the **setup** command to initialize the sensor, see [Basic Sensor Setup, page 3-5](#).
- For the procedure for upgrading the recovery partition to the most recent version, see [Upgrading the Recovery Partition, page 21-6](#).

Installing System Images

**Caution**

All user configuration settings are lost when you install the system image. Before trying to recover the sensor by installing the system image, try to recover by using the **recover application-partition** command or by selecting the recovery partition during sensor bootstrap.

This section contains the procedures for installing system images on the appliances and modules. It contains the following topics:

- [ROMMON, page 21-15](#)
- [TFTP Servers, page 21-15](#)
- [Connecting an Appliance to a Terminal Server, page 21-16](#)
- [Installing the System Image for the IPS 4300 Series Sensors, page 21-16](#)
- [Installing the System Image for the IPS 4500 Series Sensors, page 21-20](#)
- [Installing the System Image for the ASA 5500-X IPS SSP, page 21-22](#)
- [Installing the System Image for the ASA 5585-X IPS SSP, page 21-23](#)

ROMMON

Some Cisco sensors include a preboot CLI called ROMMON, which lets you boot images on sensors where the image on the primary device is missing, corrupt, or otherwise unable to boot the normal application. ROMMON is particularly useful for recovering remote sensors as long as the serial console port is available.

Access to ROMMON is available only through the serial console port, a Cisco-standard asynchronous RS-232C DTE available in an RJ-45F connector on the sensor chassis. The serial port is configured for 9600 baud, 8 data bits, 1 stop bit, no parity, and no flow control.

For More Information

For the procedure for using a terminal server, see [Connecting an Appliance to a Terminal Server, page 21-16](#).

TFTP Servers

ROMMON uses TFTP to download an image and launch it. TFTP does not address network issues such as latency or error recovery. It does implement a limited packet integrity check so that packets arriving in sequence with the correct integrity value have an extremely low probability of error. But TFTP does not offer pipelining so the total transfer time is equal to the number of packets to be transferred times the network average RTT. Because of this limitation, we recommend that the TFTP server be located on the same LAN segment as the sensor. Any network with an RTT less than a 100 milliseconds should provide reliable delivery of the image. Be aware that some TFTP servers limit the maximum file size that can be transferred to ~32 MB.

Connecting an Appliance to a Terminal Server

A terminal server is a router with multiple, low speed, asynchronous ports that are connected to other serial devices. You can use terminal servers to remotely manage network equipment, including appliances.

To set up a Cisco terminal server with RJ-45 or hydra cable assembly connections, follow these steps:

-
- Step 1** Connect to a terminal server using one of the following methods:
- For terminal servers with RJ-45 connections, connect a rollover cable from the console port on the appliance to a port on the terminal server.
 - For hydra cable assemblies, connect a straight-through patch cable from the console port on the appliance to a port on the terminal server.

- Step 2** Configure the line and port on the terminal server. In enable mode, enter the following configuration, where # is the line number of the port to be configured.

```
config t
line #
login
transport input all
stopbits 1
flowcontrol hardware
speed 9600
exit
exit
wr mem
```

- Step 3** Be sure to properly close a terminal session to avoid unauthorized access to the appliance. If a terminal session is not stopped properly, that is, if it does not receive an exit(0) signal from the application that initiated the session, the terminal session can remain open. When terminal sessions are not stopped properly, authentication is not performed on the next session that is opened on the serial port.

**Caution**

Always exit your session and return to a login prompt before terminating the application used to establish the connection.

**Caution**

If a connection is dropped or terminated by accident, you should reestablish the connection and exit normally to prevent unauthorized access to the appliance.

Installing the System Image for the IPS 4300 Series Sensors

**Note**

This procedure is for IPS 4345, but is also applicable to IPS 4360. The system image for IPS 4360 has “4360” in the filename.

You can install the IPS 4345 and IPS 4360 system image by using the ROMMON on the appliance to TFTP the system image on to the compact flash device. To install the IPS 4345 and IPS 4360 system image, follow these steps:

- Step 1** Download the IPS 4345 system image file to the tftp root directory of a TFTP server that is accessible from your IPS 4345.



Note Make sure you can access the TFTP server location from the network connected to the Ethernet port of your IPS 4345.

- Step 2** Boot the IPS 4345.

Booting system, please wait...

```
CISCO SYSTEMS
Embedded BIOS Version 1.0(5)0 09/14/04 12:23:35.90
```

```
Low Memory: 631 KB
High Memory: 2048 MB
PCI Device Table.
```

Bus	Dev	Func	VendID	DevID	Class	Irq
00	00	00	8086	2578	Host Bridge	
00	01	00	8086	2579	PCI-to-PCI Bridge	
00	03	00	8086	257B	PCI-to-PCI Bridge	
00	1C	00	8086	25AE	PCI-to-PCI Bridge	
00	1D	00	8086	25A9	Serial Bus	11
00	1D	01	8086	25AA	Serial Bus	10
00	1D	04	8086	25AB	System	
00	1D	05	8086	25AC	IRQ Controller	
00	1D	07	8086	25AD	Serial Bus	9
00	1E	00	8086	244E	PCI-to-PCI Bridge	
00	1F	00	8086	25A1	ISA Bridge	
00	1F	02	8086	25A3	IDE Controller	11
00	1F	03	8086	25A4	Serial Bus	5
00	1F	05	8086	25A6	Audio	5
02	01	00	8086	1075	Ethernet	11
03	01	00	177D	0003	Encrypt/Decrypt	9
03	02	00	8086	1079	Ethernet	9
03	02	01	8086	1079	Ethernet	9
03	03	00	8086	1079	Ethernet	9
03	03	01	8086	1079	Ethernet	9
04	02	00	8086	1209	Ethernet	11
04	03	00	8086	1209	Ethernet	5

```
Evaluating BIOS Options ...
Launch BIOS Extension to setup ROMMON
```

```
Cisco Systems ROMMON Version (1.0(5)0) #1: Tue Sep 14 12:20:30 PDT 2004
```

```
Platform IPS-4345-K9
Management0/0
```

```
MAC Address: 0000.c0ff.ee01
```

- Step 3** Press **Break** or **Esc** at the following prompt while the system is booting to interrupt boot. Press the spacebar to begin boot immediately.



Note You have ten seconds to press **Break** or **Esc**.

Use BREAK or ESC to interrupt boot.
Use SPACE to begin boot immediately.

The system enters ROMMON mode. The `rommon>` prompt appears.

Step 4 Check the current network settings.

```
rommon> set

ROMMON Variable Settings:
  ADDRESS=0.0.0.0
  SERVER=0.0.0.0
  GATEWAY=0.0.0.0
  PORT=Management0/0
  VLAN=untagged
  IMAGE=
  CONFIG=
```

The variables have the following definitions:

- Address—Local IP address of the IPS 4345.
- Server—TFTP server IP address where the application image is stored.
- Gateway—Gateway IP address used by the IPS 4345.
- Port—Ethernet interface used for the IPS 4345 management.
- VLAN—VLAN ID number (leave as untagged).
- Image—System image file/path name.
- Config—Unused by these platforms.



Note Not all values are required to establish network connectivity. The address, server, gateway, and image values are required. If you are not sure of the settings needed for your local environment, contact your system administrator.

Step 5 If necessary, change the interface used for the TFTP download.



Note The default interface used for TFTP downloads is Management 0/0, which corresponds to the MGMT interface of the IPS 4345.

```
rommon> PORT=interface_name
```

Step 6 If necessary, assign an IP address for the local port on the IPS 4345.

```
rommon> ADDRESS=ip_address
```



Note Use the same IP address that is assigned to the IPS 4345.

Step 7 Assign the TFTP server IP address.

```
rommon> SERVER=ip_address
```

Step 8 If necessary, assign the gateway IP address.

```
rommon> GATEWAY=ip_address
```

- Step 9** Verify that you have access to the TFTP server by pinging it from your local Ethernet port with one of the following commands:

```
rommon> ping server_ip_address
rommon> ping server
```

- Step 10** If necessary define the path and filename on the TFTP file server from which you are downloading the image.

```
rommon> IMAGE=path file_name
```

**Caution**

Make sure that you enter the **IMAGE** command in all uppercase. You can enter the other ROMMON commands in either lower case or upper case, but the **IMAGE** command specifically must be all uppercase.

UNIX Example

```
rommon> IMAGE=system_images/IPS-4345-K9-sys-1.1-a-7.3-1-E4.img
```



Note The path is relative to the default tftpboot directory of the UNIX TFTP server. Images located in the default tftpboot directory do not have any directory names or slashes in the IMAGE specification.

Windows Example

```
rommon> IMAGE=system_images/IPS-4345-K9-sys-1.1-a-7.3-1-E4.img
```

- Step 11** Enter **set** and press **Enter** to verify the network settings.



Note You can use the **sync** command to store these settings in NVRAM so they are maintained across boots. Otherwise, you must enter this information each time you want to boot an image from ROMMON.

- Step 12** Download and install the system image.

```
rommon> tftp
```

**Caution**

To avoid corrupting the system image, do not remove power from the IPS 4345 while the system image is being installed.



Note If the network settings are correct, the system downloads and boots the specified image on the IPS 4345. Be sure to use the IPS 4345 image.

For More Information

- For more information about TFTP servers, see [TFTP Servers](#), page 21-15.
- For a list of the specific system image files, see [IPS 7.3\(1\)E4 Files](#), page 21-3.
- For the procedure for locating software on Cisco.com, see [Obtaining Cisco IPS Software](#), page 20-1.

Installing the System Image for the IPS 4500 Series Sensors

**Note**

The following procedure references the IPS 4510 but it also refers to the IPS 4520 and IPS 4520-XL.

You can install the IPS 4500 series system image by using the ROMMON on the appliance to TFTP the system image onto the compact flash device.

To install the IPS 4510 system image, follow these steps:

- Step 1** Download the IPS 4510 system image file to the tftp root directory of a TFTP server that is accessible from your IPS 4510.

**Note**

Make sure you can access the TFTP server location from the network connected to the Management port of your IPS 4510.

- Step 2** Boot the IPS 4510.

- Step 3** Press **Break** or **Esc** at the following prompt while the system is booting to interrupt boot. Press the spacebar to begin boot immediately.

**Note**

You have ten seconds to press **Break** or **Esc**.

Use `BREAK` or `ESC` to interrupt boot.
Use `SPACE` to begin boot immediately.

The system enters ROMMON mode. The `rommon>` prompt appears.

- Step 4** Check the current network settings.

```
rommon> set
```

```
ROMMON Variable Settings:
ADDRESS=0.0.0.0
SERVER=0.0.0.0
GATEWAY=0.0.0.0
PORT=Management0/0
VLAN=untagged
IMAGE=
CONFIG=
LINKTIMEOUT=20
PKTTIMEOUT=2
RETRY=20
```

The variables have the following definitions:

- Address—Specifies the local IP address of the IPS 4510.
- Server—Specifies the TFTP server IP address where the application image is stored.
- Gateway—Specifies the gateway IP address used by the IPS 4510.
- Port—Specifies the Ethernet interface used for IPS 4510 management.
- VLAN—Specifies the VLAN ID number (leave as untagged).
- Image—Specifies the system image file/path name.
- Config—Unused by these platforms.



Note Not all values are required to establish network connectivity. The address, server, gateway, and image values are required. If you are not sure of the settings needed for your local environment, contact your system administrator.

Step 5 If necessary, assign an IP address for the Management port on the IPS 4510.

```
rommon> ADDRESS=ip_address
```



Note Use the same IP address that is assigned to the IPS 4510.

Step 6 If necessary, assign the TFTP server IP address.

```
rommon> SERVER=ip_address
```

Step 7 If necessary, assign the gateway IP address.

```
rommon> GATEWAY=ip_address
```

Step 8 Verify that you have access to the TFTP server by pinging it from your local Ethernet port with one of the following commands:

```
rommon> ping server_ip_address
rommon> ping server
```

Step 9 If necessary define the path and filename on the TFTP file server from which you are downloading the image.

```
rommon> IMAGE=path/file_name
```

UNIX Example

```
rommon> IMAGE=/system_images/IPS-4510-K9-sys-1.1-a-7.3-1-E4.img
```



Note The path is relative to the UNIX TFTP server default tftpboot directory. Images located in the default tftpboot directory do not have any directory names or slashes in the IMAGE specification.

Windows Example

```
rommon> IMAGE=\system_images\IPS-4510-K9-sys-1.1-a-7.3-1-E4.img
```

Step 10 Enter **set** and press **Enter** to verify the network settings.



Note You can use the **sync** command to store these settings in NVRAM so they are maintained across boots. Otherwise, you must enter this information each time you want to boot an image from ROMMON.

Step 11 Download and install the system image.

```
rommon> tftp
```



Caution To avoid corrupting the system image, do not remove power from the IPS 4510 while the system image is being installed.

**Note**

If the network settings are correct, the system downloads and boots the specified image on the IPS 4510. Be sure to use the IPS 4510 image.

For More Information

- For more information about TFTP servers, see [TFTP Servers, page 21-15](#).
- For a list of the specific system image files, see [IPS 7.3\(1\)E4 Files, page 21-3](#).
- For the procedure for locating software on Cisco.com, see [Obtaining Cisco IPS Software, page 20-1](#).

Installing the System Image for the ASA 5500-X IPS SSP

**Note**

Be sure the TFTP server that you specify can transfer files up to 60 MB in size.

**Note**

The CLI output is an example of what your configuration may look like. It will not match exactly due to the optional setup choices, sensor model, and IPS version you have installed.

To install the system image on the ASA 5500-X IPS SSP, follow these steps:

- Step 1** Download the IPS system image file corresponding to your ASA platform to the tftp root directory of a TFTP server that is accessible from your adaptive security appliance.

**Note**

Make sure you can access the TFTP server location from the network connected to the Ethernet port of the adaptive security appliance.

- Step 2** Log in to the adaptive security appliance.

- Step 3** Enter enable mode.

```
asa> enable
```

- Step 4** Copy the IPS image to the disk0 flash of the adaptive security appliance.

```
asa# copy tftp://192.0.2.0/directory/IPS-5545-K9-sys-1.1-a-7.1-3-E4.aip disk0:
```

- Step 5** Image the ASA 5500-X IPS SSP.

```
asa# sw-module module ips recover configure image  
disk0:/IPS-SSP_5545-K9-sys-1.1-a-7.3-1-E4.aip
```

- Step 6** Execute the recovery. This transfers the image from the TFTP server to the ASA 5500-X IPS SSP and restarts it.

```
asa# sw-module module ips recover boot
```

- Step 7** Periodically check the recovery until it is complete.

```
asa# show module
```

```

Mod Card Type                               Model                               Serial No.
-----
 0 Cisco ASA 5545 Appliance with 8 GE ports, 1 ASA5545                       ABC1234D56E
 1 IPS 5545 Intrusion Protection System      IPS5545                             ABC1234D56E

Mod MAC Address Range                       Hw Version  Fw Version  Sw Version
-----
0    503d.e59c.6dc1 to 503d.e59c.6dca  1.0
ips  503d.e59c.6dcb to 503d.e59c.6dcb  N/A      N/A      7.1(3)E4

Mod SSM Application Name                   Status           SSM Application Version
-----
 1 IPS                                     Up              7.1(3)E4

Mod Status           Data Plane Status  Compatibility
-----
 0 Up Sys           Not Applicable
 1 Up              Up
asa#

```



Note The Status field in the output indicates the operational status of the ASA 5500-X IPS SSP. An ASA 5500-X IPS SSP operating normally shows a status of “Up.” While the adaptive security appliance transfers an application image to the ASA 5500-X IPS SSP, the Status field in the output reads “Recover.” When the adaptive security appliance completes the image transfer and restarts the ASA 5500-X IPS SSP, the newly transferred image is running.



Note To debug any errors that may happen in the recovery process, use the **debug module-boot** command to enable debugging of the system reimaging process.

Step 8 Session to the ASA 5500-X IPS SSP and initialize it with the **setup** command.

For More Information

- For more information about TFTP servers, see [TFTP Servers, page 21-15](#).
- For a list of the specific system image files, see [IPS 7.3\(1\)E4 Files, page 21-3](#).
- For the procedure for locating software on Cisco.com, see [Obtaining Cisco IPS Software, page 20-1](#).
- For the procedure for initializing the ASA 5500-X IPS SSP, see [Advanced Setup for the ASA 5500-X IPS SSP, page 3-14](#).

Installing the System Image for the ASA 5585-X IPS SSP

This section describes how to install the ASA 5585-X IPS SSP system image using the `hw-module` command or `ROMMON`, and contains the following topics:

- [Installing the ASA 5585-X IPS SSP System Image Using the hw-module Command, page 21-24](#)
- [Installing the ASA 5585-X IPS SSP System Image Using ROMMON, page 21-26](#)

Installing the ASA 5585-X IPS SSP System Image Using the hw-module Command


Note

Be sure the TFTP server that you specify can transfer files up to 60 MB in size.


Note

This process can take approximately 15 minutes to complete, depending on your network and the size of the image.


Note

The CLI output is an example of what your configuration may look like. It will not match exactly due to the optional setup choices, sensor model, and IPS version you have installed.

To install the system image, transfer the software image from a TFTP server to the ASA 5585-X IPS SSP using the adaptive security appliance CLI. The adaptive security appliance can communicate with the ROMMON application of the ASA 5585-X IPS SSP to transfer the image. To install the ASA 5585-X IPS SSP software image, follow these steps:

- Step 1** Download the ASA 5585-X IPS SSP system image file to the tftp root directory of a TFTP server that is accessible from your adaptive security appliance.


Note

Make sure you can access the TFTP server location from the network connected to the Ethernet port of your adaptive security appliance.

- Step 2** Log in to the adaptive security appliance.

- Step 3** Enter enable mode.

```
asa# enable
```

- Step 4** Configure the recovery settings for the ASA 5585-X IPS SSP.

```
asa (enable)# hw-module module 1 recover configure
```


Note

If you make an error in the recovery configuration, use the **hw-module module 1 recover stop** command to stop the system reimaging and then you can correct the configuration.

- Step 5** Specify the TFTP URL for the software image.

```
Image URL [tftp://0.0.0.0/]:
```

Example

```
Image URL [tftp://0.0.0.0/]: tftp://192.0.2.0/IPS-SSP_40-K9-sys-1.1-a-7.1-3-E4.img
```

- Step 6** Specify the command and control interface of the ASA 5585-X IPS SSP.


Note

The port IP address is the management IP address of the ASA 5585-X IPS SSP.

```
Port IP Address [0.0.0.0]:
```


Example

```
Port IP Address [0.0.0.0]: 10.89.149.231
```

Step 7 Leave the VLAN ID at 0.

```
VLAN ID [0]:
```

Step 8 Specify the default gateway of the ASA 5585-X IPS SSP.

```
Gateway IP Address [0.0.0.0]:
```

Example

```
Gateway IP Address [0.0.0.0]: 10.89.149.254
```

Step 9 Execute the recovery. This transfers the software image from the TFTP server to the ASA 5585-X IPS SSP and restarts it.

```
asa# hw-module module 1 recover boot
```

Step 10 Periodically check the recovery until it is complete.



Note The status reads `Recovery` during recovery and reads `Up` when installation is complete.

```
asa# show module 1 details
Getting details from the Service Module, please wait...
ASA 5585-X IPS Security Services Processor-10 with 8GE
Model:          ASA5585-SSP-IPS40
Hardware version: 1.0
Serial Number:  JAF1350ABSL
Firmware version: 2.0(1)3
Software version: 7.1(3)E4
MAC Address Range: 8843.e12f.5414 to 8843.e12f.541f
App. name:      IPS
App. Status:    Up
App. Status Desc: Normal Operation
App. version:   7.1(3)E4
Data plane Status: Up
Status:         Up
Mgmt IP addr:   192.0.2.0
Mgmt Network mask: 255.255.255.0
Mgmt Gateway:   10.89.148.254
Mgmt Access List: 10.0.0.0/8
Mgmt Access List: 64.0.0.0/8
Mgmt web ports: 443
Mgmt TLS enabled true
asa#
```



Note The Status field in the output indicates the operational status of the ASA 5585-X IPS SSP. An ASA 5585-X IPS SSP operating normally shows a status of “Up.” While the adaptive security appliance transfers the software image to the ASA 5585-X IPS SSP, the Status field in the output reads “Recover.” When the adaptive security appliance completes the software image transfer and restarts the ASA 5585-X IPS SSP, the newly transferred image is running.



Note To debug any errors that may happen during this process, use the **debug module-boot** command to enable debugging of the software installation process.

- Step 11** Session to the ASA 5585-X IPS SSP.
- Step 12** Enter `cisco` three times and your new password twice.
- Step 13** Initialize the ASA 5585-X IPS SSP with the `setup` command.

For More Information

- For more information about TFTP servers, see [TFTP Servers, page 21-15](#).
- For a list of the specific system image files, see [IPS 7.3\(1\)E4 Files, page 21-3](#).
- For the procedure for locating software on Cisco.com, see [Obtaining Cisco IPS Software, page 20-1](#).
- For the procedure for initializing the ASA 5585-X IPS SSP, see [Advanced Setup for the ASA 5585-X IPS SSP, page 3-15](#).

Installing the ASA 5585-X IPS SSP System Image Using ROMMON

You can install the ASA 5585-X IPS SSP system image by using the ROMMON on the adaptive security appliance to TFTP the system image onto the ASA 5585-X IPS SSP.

To install the ASA 5585-X IPS SSP system image, follow these steps:

- Step 1** Download the ASA 5585-X IPS SSP system image file to the tftp root directory of a TFTP server that is accessible from your adaptive security appliance.



Note Make sure you can access the TFTP server location from the network connected to the Ethernet port of your adaptive security appliance.

- Step 2** Boot the ASA 5585-X IPS SSP.

```
Booting system, please wait...
```

```
CISCO SYSTEMS
Embedded BIOS Version 0.0(2)10 11:16:38 04/15/10
Com KbdBuf SMM UsbHid Msg0 Prompt Pmrt Cache1 LowM ExtM HugeM Cache2 Flg Siz0 Amrt PMM
PnpDsp Smbios Lpt0 Npx1 Apm Lpl Acpi Typ Dbg Enb Mp MemReduce MemSync1 CallRoms MemSync2
DriveInit
```

```
Total memory : 12 GB
Total number of CPU cores : 8
Com Lpl AdmGr2 Brd10 Plx2 OEM0=7EFF5C74
Cisco Systems ROMMON Version (1.0(12)10) #0: Thu Apr 8 00:12:33 CDT 2010
```

```
Use BREAK or ESC to interrupt boot.
Use SPACE to begin boot immediately.
Boot interrupted.
```

```
Management0/0
Link is UP
MAC Address: 5475.d029.7fa9
```

- Step 3** Press **Break** or **Esc** at the following prompt while the system is booting to interrupt boot. Press the spacebar to begin boot immediately.



Note You have ten seconds to press **Break** or **Esc**.

Use **BREAK** or **ESC** to interrupt boot.
Use **SPACE** to begin boot immediately.

The system enters ROMMON mode. The `rommon>` prompt appears.

Step 4 Check the current network settings.

```
rommon #0> set
ROMMON Variable Settings:
  ADDRESS=0.0.0.0
  SERVER=0.0.0.0
  GATEWAY=0.0.0.0
  PORT=Management0/0
  VLAN=untagged
  IMAGE=
  CONFIG=
  LINKTIMEOUT=20
  PKTTIMEOUT=4
  RETRY=20
```

The variables have the following definitions:

- Address—Specifies the local IP address of the ASA 5585-X IPS SSP.
- Server—Specifies the TFTP server IP address where the application image is stored.
- Gateway—Specifies the gateway IP address used by the ASA 5585-X IPS SSP.
- Port—Specifies the ethernet interface used for the ASA 5585-X IPS SSP management.
- VLAN—Specifies the VLAN ID number (leave as untagged).
- Image—Specifies the system image file/path name.
- Config—Specifies the unused by these platforms.



Note Not all values are required to establish network connectivity. The address, server, gateway, and image values are required. If you are not sure of the settings needed for your local environment, contact your system administrator.

Step 5 If necessary, change the interface used for the TFTP download.



Note The default interface used for TFTP downloads is Management 0/0, which corresponds to the management interface of the ASA 5585-X IPS SSP.

```
rommon> PORT=interface_name
```

Step 6 If necessary, assign an IP address for the local port on the ASA 5585-X IPS SSP.

```
rommon> ADDRESS=ip_address
```



Note Use the same IP address that is assigned to the ASA 5585-X IPS SSP.

Step 7 If necessary, assign the TFTP server IP address.

```
rommon> SERVER=ip_address
```

Step 8 If necessary, assign the gateway IP address.

```
rommon> GATEWAY=ip_address
```

Step 9 Verify that you have access to the TFTP server by pinging it from your local Ethernet port with one of the following commands.

```
rommon> ping server_ip_address
rommon> ping server
```

Step 10 If necessary define the path and filename on the TFTP file server from which you are downloading the image.

```
rommon> IMAGE=path/file_name
```



Caution

Make sure that you enter the **IMAGE** command in all uppercase. You can enter the other ROMMON commands in either lower case or upper case, but the **IMAGE** command specifically must be all uppercase.

UNIX Example

```
rommon> IMAGE=/system_images/IPS-SSP_10-K9-sys-1.1-a-7.1-3-E4.img
```



Note

The path is relative to the default tftpboot directory of the UNIX TFTP server. Images located in the default tftpboot directory do not have any directory names or slashes in the IMAGE specification.

Windows Example

```
rommon> IMAGE=\system_images\IPS-SSP_10-K9-sys-1.1-a-7.1-3-E4.img
```

Step 11 Enter **set** and press **Enter** to verify the network settings.



Note

You can use the **sync** command to store these settings in NVRAM so they are maintained across boots. Otherwise, you must enter this information each time you want to boot an image from ROMMON.

Step 12 Download and install the system image.

```
rommon> tftp
```



Note

If the network settings are correct, the system downloads and boots the specified image on the ASA 5585-X IPS SSP. Be sure to use the ASA 5585-X IPS SSP image.

**Caution**

To avoid corrupting the system image, do not remove power from the ASA 5585-X IPS SSP while the system image is being installed.

For More Information

- For more information about TFTP servers, see [TFTP Servers, page 21-15](#).
- For a list of the specific system image files, see [IPS 7.3\(1\)E4 Files, page 21-3](#).
- For the procedure for locating software on Cisco.com, see [Obtaining Cisco IPS Software, page 20-1](#).
- For the procedure for initializing the ASA 5585-X IPS SSP, see [Advanced Setup for the ASA 5585-X IPS SSP, page 3-15](#).



System Architecture

This appendix describes the IPS system architecture, and contains the following sections:

- [IPS System Design, page A-1](#)
- [System Applications, page A-3](#)
- [User Interaction, page A-5](#)
- [Security Features, page A-5](#)
- [MainApp, page A-6](#)
- [SensorApp, page A-22](#)
- [CollaborationApp, page A-27](#)
- [SwitchApp, page A-29](#)
- [CLI, page A-30](#)
- [Communications, page A-31](#)
- [Cisco IPS File Structure, page A-34](#)
- [Summary of Cisco IPS Applications, page A-35](#)

Understanding the IPS System Architecture

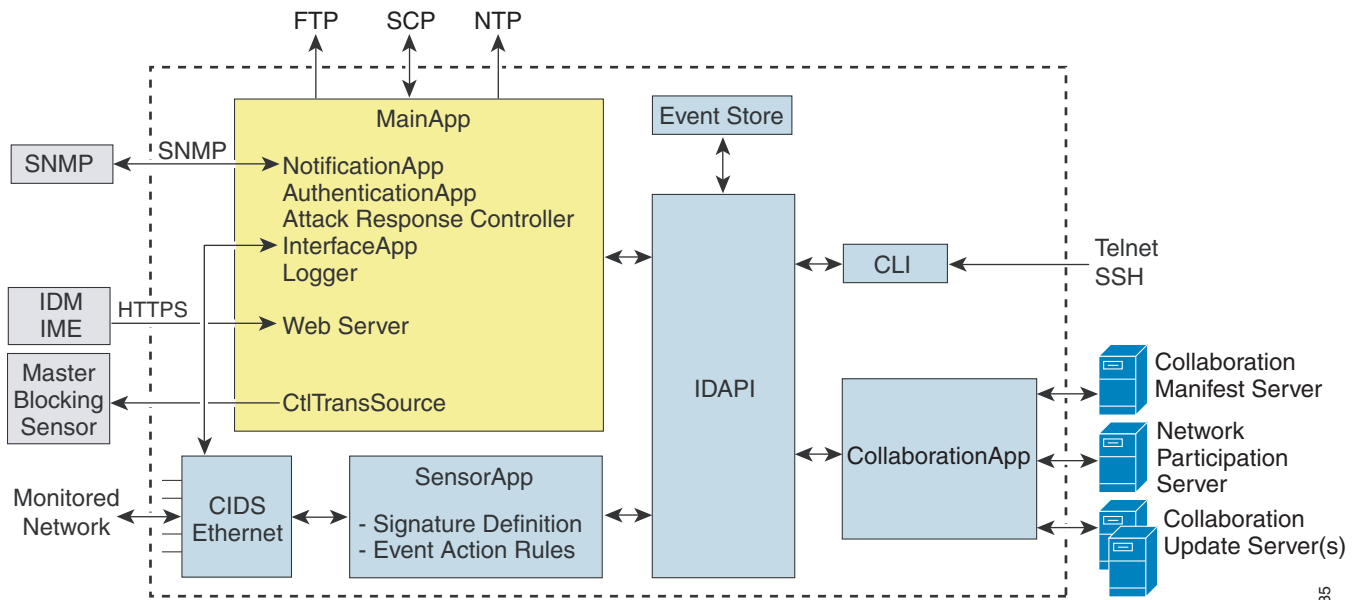
The purpose of the Cisco IPS is to detect and prevent malicious network activity. You can install the Cisco IPS software on two platforms: appliances and the modules. The Cisco IPS contains a management application and a monitoring application. The IDM is a network management JAVA application that you can use to manage and monitor the IPS. The IME is an IPS network monitoring JAVA application that you can use to view IPS events. The IME also contains the IDM configuration component. The IDM and the IME communicate with the IPS using HTTP or HTTPS and are hosted on your computer.

IPS System Design

The Cisco IPS software runs on the Linux operating system. We have hardened the Linux OS by removing unnecessary packages from the OS, disabling unused services, restricting network access, and removing access to the shell.

Figure A-1 illustrates the system design for IPS software.

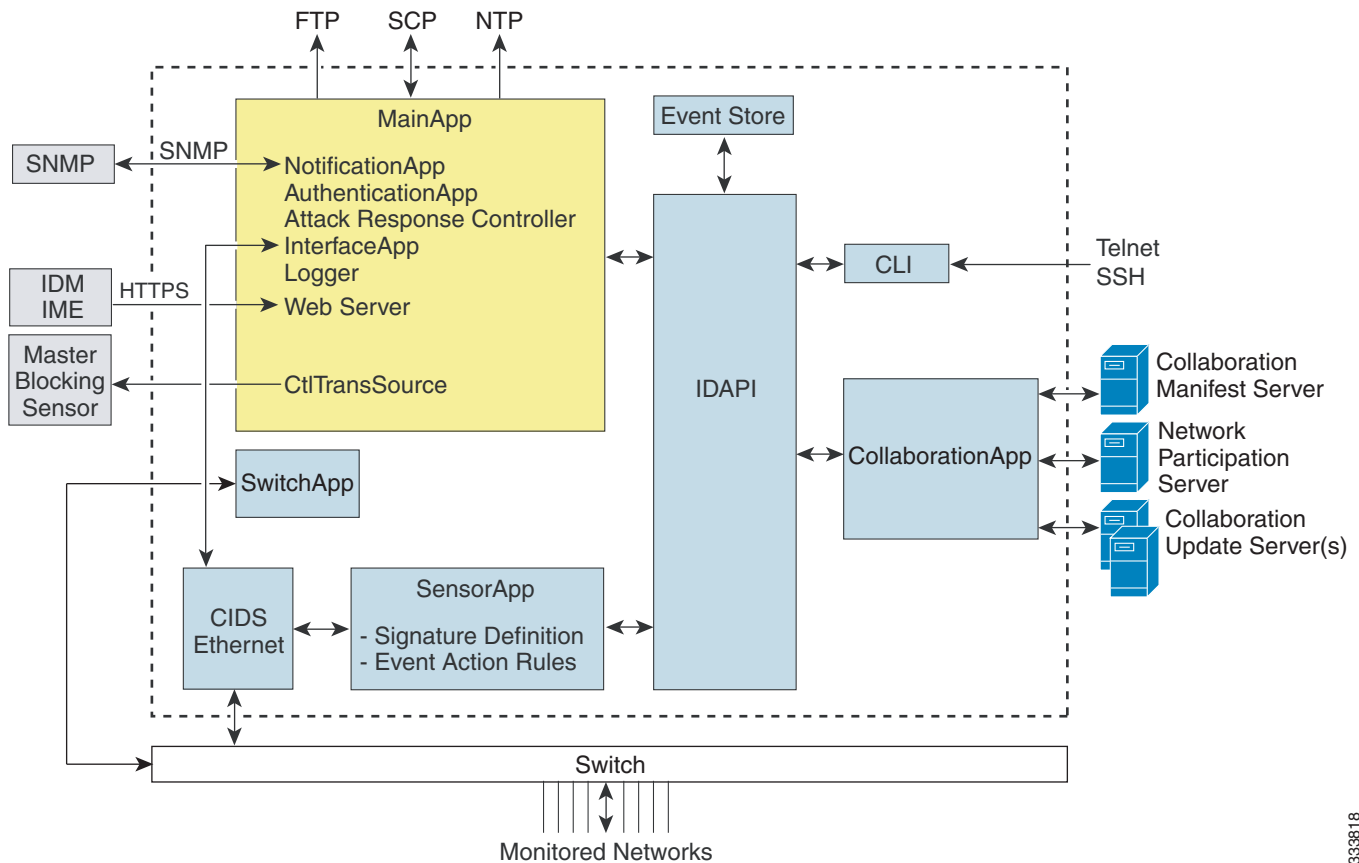
Figure A-1 System Design for the IPS



251235

Figure A-2 illustrates the system design for IPS software for the IPS 4500 series sensors.

Figure A-2 System Design for IPS 4500 Series Sensors



333818

For More Information

- For detailed information about the MainApp, see [MainApp](#), page A-6.
- For detailed information about the SensorApp, see [SensorApp](#), page A-22.
- For detailed information about the CollaborationApp, see [CollaborationApp](#), page A-27.
- For detailed information about the CLI, see [CLI](#), page A-30.
- For detailed information about the SwitchApp, see [SwitchApp](#), page A-29.

System Applications



Note

Each application has its own configuration file in XML format.

The Cisco IPS software includes the following applications:

- MainApp—Initializes the system, starts and stops the other applications, configures the OS, and performs upgrades. It contains the following components:
 - ctlTransSource (Control Transaction server)—Allows sensors to send control transactions. This is used to enable the master blocking sensor capability of Attack Response Controller (formerly known as Network Access Controller).
 - Event Store—An indexed store used to store IPS events (error, status, and alert system messages) that is accessible through the CLI, IDM, IME, ASDM, or SDEE.



Note

The Event Store has a fixed size of 30 MB for all platforms

- InterfaceApp—Handles bypass and physical settings and defines paired interfaces. Physical settings are speed, duplex, and administrative state.
- Logger—Writes all the log messages of the application to the log file and the error messages of the application to the Event Store.
- Attack Response Controller (formerly known as Network Access Controller) —Manages remote network devices (firewalls, routers, and switches) to provide blocking capabilities when an alert event has occurred. The ARC creates and applies ACLs on the controlled network device or uses the **shun** command (firewalls).
- NotificationApp—Sends SNMP traps when triggered by alert, status, and error events. The NotificationApp uses the public domain SNMP agent. SNMP GETs provide information about the general health of the sensor.
- Web server (HTTP SDEE server)—Provides a web interface and communication with the other IPS devices through the SDEE protocol using several servlets to provide the IPS services.
- AuthenticationApp—Verifies that users are authorized to perform CLI, IDM, IME, ASDM, or SDEE actions.
- SensorApp (Analysis Engine)—Performs packet capture and analysis.
- CollaborationApp—Interfaces with the MainApp and the SensorApp using various interprocess communication technologies including IDAPI control transactions, semaphores, shared memory, and file exchange.
- CLI—The interface that is run when you successfully log in to the sensor through Telnet or SSH. All accounts created through the CLI will use the CLI as their shell (except the service account—only one service account is allowed). Allowed CLI commands depend on the privilege of the user.

All Cisco IPS applications communicate with each other through a common API called the IDAPI. Remote applications (other sensors, management applications, and third-party software) communicate with sensors through the SDEE protocol.

The sensor has the following partitions:

- Application partition—A full IPS system image.
- Recovery partition—A special purpose image used for recovery of the sensor. Booting into the recovery partition enables you to completely reimage the application partition. Network settings are preserved, but all other configuration is lost.

User Interaction

You interact with the Cisco IPS in the following ways:

- Configure device parameters

You generate the initial configuration for the system and its features. This is an infrequent task, usually done only once. The system has reasonable default values to minimize the number of modifications you must make. You can configure Cisco IPS through the CLI, IDM, IME, CSM, ASDM, or through another application using SDEE.

- Tune

You make minor modifications to the configuration, primarily to Analysis Engine, which is the portion of the application that monitors network traffic. You can tune the system frequently after initially installing it on the network until it is operating efficiently and only producing information you find useful. You can create custom signatures, enable features, or apply a service pack or signature update. You can tune Cisco IPS through the CLI, IDM, IME, CSM, ASDM, or through another application using SDEE.

- Update

You can schedule automatic updates or apply updates immediately to the applications and signature data files. You can update Cisco IPS through the CLI, IDM, IME, CSM, ASDM, or through another application using SDEE.

- Retrieve information

You can retrieve data (status messages, errors, and alerts) from the system through the CLI, IDM, IME, CSM, ASDM, or another application using SDEE.

For More Information

For detailed information about SDEE, see [SDEE, page A-33](#).

Security Features

Cisco IPS has the following security features:

- Network access is restricted to hosts who are specifically allowed access.
- All remote hosts who attempt to connect through the web server, SSH and SCP or Telnet will be authenticated.
- By default Telnet access is disabled. You can choose to enable Telnet.
- By default SSH access is enabled.
- An FTP server does not run on the sensor. You can use SCP to remotely copy files.
- By default the web server uses TLS or SSL. You can choose to disable TLS and SSL.
- Unnecessary services are disabled.
- Only the SNMP set required by the Cisco MIB Police is allowed within the CISCO-CIDS-MIB. OIDs implemented by the public domain SNMP agent will be writeable when specified by the MIB.

For More Information

For detailed information about SNMP and Cisco MIBs, see [Chapter 15, “Configuring SNMP.”](#)

MainApp

This section describes the MainApp, and contains the following topics:

- [Understanding the MainApp, page A-6](#)
- [MainApp Responsibilities, page A-6](#)
- [Event Store, page A-7](#)
- [NotificationApp, page A-9](#)
- [CtlTransSource, page A-11](#)
- [Attack Response Controller, page A-12](#)
- [Logger, page A-19](#)
- [AuthenticationApp, page A-20](#)
- [Web Server, page A-22](#)

Understanding the MainApp

The MainApp includes all IPS components except SensorApp and the CLI. It is loaded by the operating system at startup and loads SensorApp. The MainApp then brings the following subsystem components up:

- Authentication
- Logger
- ARC
- Web Server
- Notification (SNMP)
- External Product Interface
- Interface manager
- Event Store
- Health and security monitoring

MainApp Responsibilities

The MainApp has the following responsibilities:

- Validate the Cisco-supported hardware platform
- Report software version and PEP information
- Start, stop, and report the version of the IPS components
- Configure the host system settings
- Manage the system clock
- Manage the Event Store
- Install and uninstall software upgrades



Note In the Cisco IPS, the MainApp can automatically download signature and signature engine updates from Cisco.com.

- Shut down or reboot the operating system

The MainApp responds to the **show version** command by displaying the following information:

- Sensor build version
- MainApp version
- Version of each running application
- Version and timestamp of each installed upgrade
- Next downgrade version of each installed upgrade
- Platform version
- Version of sensor build on the other partition

The MainApp also gathers the host statistics and reports the health and security monitoring status.

Event Store

This section describes the Event Store, and contains the following topics:

- [Understanding the Event Store, page A-7](#)
- [Event Data Structures, page A-8](#)
- [IPS Events, page A-9](#)

Understanding the Event Store



Note The Event Store has a fixed size of 30 MB for all platforms.

Each IPS event is stored in the Event Store with a time stamp and a unique, monotonic, ascending ID. This time stamp is the primary key used to index the event into the fixed-size, indexed Event Store. When the circular Event Store has reached its configured size, the oldest event or events are overwritten by the new event being stored. The SensorApp is the only application that writes alert events into the Event Store. All applications write log, status, and error events into the Event Store.

The fixed-sized, indexed Event Store allows simple event queries based on the time, type, priority, and a limited number of user-defined attributes. If each event is assigned a priority of low, medium, or high, a single event query can specify a list of desired event types, intrusion event priorities, and a time range.

Table A-1 shows some examples:

Table A-1 IPS Event Examples

IPS Event Type	Intrusion Event Priority	Start Time Stamp Value	Stop Time Stamp Value	Meaning
status	—	0	Maximum value	Get all status events that are stored.
error status	—	0	65743	Get all error and status events that were stored before time 65743.
status	—	65743	Maximum value	Get status events that were stored at or after time 65743.
intrusion attack response	low	0	Maximum value	Get all intrusion and attack response events with low priority that are stored.
attack response error status intrusion	medium high	4123000000	4123987256	Get attack response, error, status, and intrusion events with medium or high priority that were stored between time 4123000000 and 4123987256.

The size of the Event Store allows sufficient buffering of the IPS events when the sensor is not connected to an IPS event consumer. Sufficient buffering depends on your requirements and the capabilities of the nodes in use. The oldest events in the circular buffer are replaced by the newest events.

Event Data Structures

The various functional units communicate the following seven types of data:

- Intrusion events—Produced by the SensorApp. The sensor detects intrusion events.
- Error events—Caused by hardware or software malfunctions.
- Status events—Reports of a change in the status of the application, for example, that its configuration has been updated.
- Control transaction log events—The sensor logs the result of a control transaction.
- Attack response events—Actions for the ARC, for example, a block request.
- Debug events—Highly detailed reports of a change in the status of the application used for debugging.
- Control transaction data—Data associated with control transactions, for example, diagnostic data from an application, session logs, and configuration data to or from an application.

All seven types of data are referred to collectively as *IPS data*. The six event types—intrusion, error, status, control transaction log, network access, and debug—have similar characteristics and are referred to collectively as *IPS events*. IPS events are produced by the several different applications that make up the IPS and are subscribed to by other IPS applications. IPS events have the following characteristics:

- They are spontaneously generated by the application instances configured to do so. There is no request from another application instance to generate a particular event.
- They have no specific destination. They are stored and then retrieved by one or more application instances.

Control transactions involve the following types of requests:

- Request to update the configuration data of an application instance
- Request for the diagnostic data of an application instance
- Request to reset the diagnostic data of an application instance
- Request to restart an application instance
- Request for ARC, such as a block request

Control transactions have the following characteristics:

- They always consist of a request followed by a response.

The request and response may have an arbitrary amount of data associated with them. The response always includes at least a positive or negative acknowledgment.

- They are point-to-point transactions.

Control transactions are sent by one application instance (the initiator) to another application instance (the responder).

IPS data is represented in XML format as an XML document. The system stores user-configurable parameters in several XML files.

IPS Events

IPS applications generate IPS events to report the occurrence of some stimulus. The events are the data, such as the alerts generated by SensorApp or errors generated by any application. Events are stored in a local database known as the Event Store.

There are five types of events:

- evAlert—Alert event messages that report when a signature is triggered by network activity.
- evStatus—Status event messages that report the status and actions of the IPS applications.
- evError—Error event messages that report errors that occurred while attempting response actions.
- evLogTransaction—Log transaction messages that report the control transactions processed by each sensor application.
- evShunRqst—Block request messages that report when ARC issues a block request.

You can view the status and error messages using the CLI, and IME, and ASDM. The SensorApp and ARC log response actions (TCP resets, IP logging start and stop, blocking start and stop, trigger packet) as status messages.

NotificationApp

The NotificationApp allows the sensor to send alerts and system error messages as SNMP traps. It subscribes to events in the Event Store and translates them into SNMP MIBs and sends them to destinations through a public-domain SNMP agent. The NotificationApp supports sending sets and gets. The SNMP GETs provide information about basic sensor health.

The NotificationApp sends the following information from the evAlert event in sparse mode:

- Originator information
- Event ID
- Event severity

- Time (UTC and local time)
- Signature name
- Signature ID
- Subsignature ID
- Participant information
- Alarm traits

The NotificationApp sends the following information from the evAlert event in detail mode:

- Originator information
- Event ID
- Event severity
- Time (UTC and local time)
- Signature name
- Signature ID
- Subsignature ID
- Version
- Summary
- Interface group
- VLAN
- Participant information
- Actions
- Alarm traits
- Signature
- IP log IDs

The NotificationApp determines which evError events to send as a trap according to the filter that you define. You can filter based on error severity (error, fatal, and warning). The NotificationApp sends the following information from the evError event:

- Originator information
- Event ID
- Event severity
- Time (UTC and local time)
- Error message

The NotificationApp supports GETs for the following general health and system information from the sensor:

- Packet loss
- Packet denies
- Alarms generated
- Fragments in FRP
- Datagrams in FRP

- TCP streams in embryonic state
- TCP streams in established state
- TCP streams in closing state
- TCP streams in system
- TCP packets queued for reassembly
- Total nodes active
- TCP nodes keyed on both IP addresses and both ports
- UDP nodes keyed on both IP addresses and both ports
- IP nodes keyed on both IP addresses
- Sensor memory critical stage
- Interface status
- Command and control packet statistics
- Fail-over state
- System uptime
- CPU usage
- Memory usage for the system
- PEP



Note Not all IPS platforms support PEP.

The NotificationApp provides the following statistics:

- Number of error traps
- Number of event action traps
- Number of SNMP GET requests
- Number of SNMP SET requests

CtlTransSource

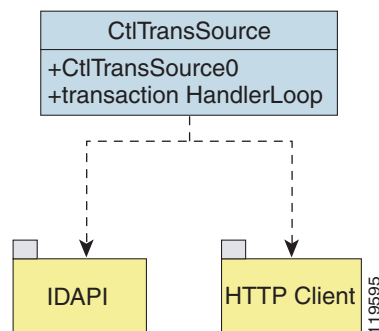
The CtlTransSource is an application that forwards locally initiated remote control transactions to their remote destinations using HTTP protocol. The CtlTransSource initiates either TLS or non-TLS connections and communicates remote control transactions to HTTP servers over these connections.

The CtlTransSource must establish sufficient credentials on the remote HTTP server to execute a remote control transaction. It establishes its credentials by presenting an identity to the HTTP server on the remote node in the form of a username and password (basic authentication). When the authentication is successful, the requestor is assigned a cookie containing a user authentication that must be presented with each request on that connection.

The transactionHandlerLoop method in the CtlTransSource serves as a proxy for remote control transaction. When a local application initiates a remote control transaction, IDAPI initially directs the transaction to the CtlTransSource. The transactionHandlerLoop method is a loop that waits on remote control transactions that are directed to the CtlTransSource.

Figure A-3 shows the transactionHandlerLoop method in the CtlTransSource.

Figure A-3 CtlTransSource



When the transactionHandlerLoop receives a remotely addressed transaction, it tries to forward the remote control transaction to its remote destination. The transactionHandlerLoop formats the transaction into a control transaction message. The transactionHandlerLoop uses the HttpClient classes to issue the control transaction request to the HTTP server on the remote node. The remote HTTP server handles the remote control transaction and returns the appropriate response message in an HTTP response. If the remote HTTP server is an IPS web server, the web server uses the CtlTransSource servlet to process the remote control transactions.

The transactionHandlerLoop returns either the response or a failure response as the response of the control transaction to the initiator of the remote control transaction. If the HTTP server returns an unauthorized status response (indicating the HTTP client has insufficient credentials on the HTTP server), the transactionHandlerLoop reissues the transaction request using the designated username and password of the CtlTransSource to authenticate the identity of the requestor. The transactionHandlerLoop continues to loop until it receives a control transaction that directs it to exit or until its exit event is signaled.

Attack Response Controller

This section describes the ARC, which is the IPS application that starts and stops blocking on routers, switches, and firewalls, and rate limits traffic on routers running Cisco IOS 12.3. A *block* is an entry in the configuration or ACL of a device to block incoming and outgoing traffic for a specific host IP address or network address. This section contains the following topics:

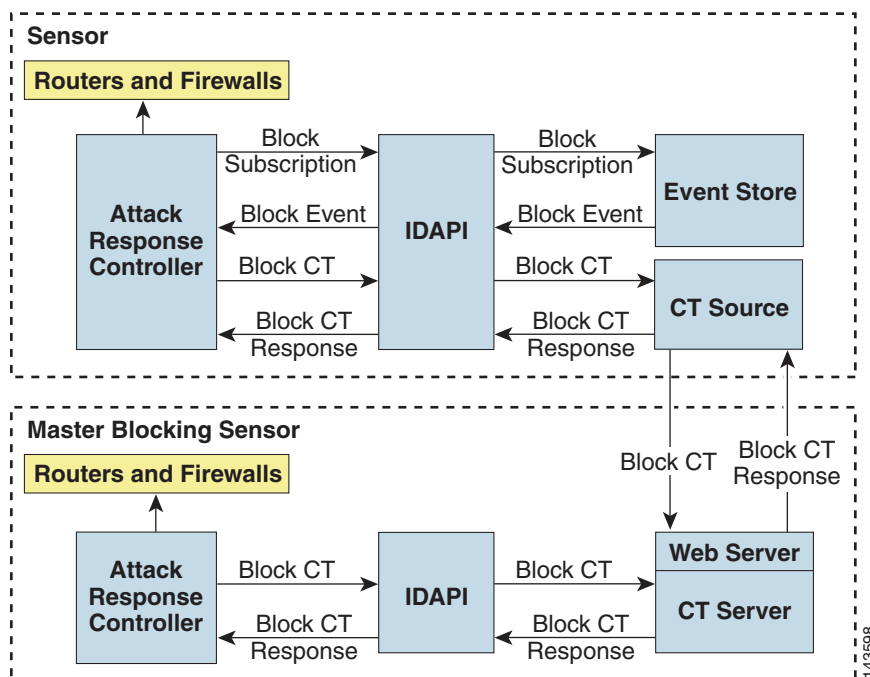
- [Understanding the ARC, page A-13](#)
- [ARC Features, page A-14](#)
- [Supported Blocking Devices, page A-15](#)
- [ACLs and VACLs, page A-16](#)
- [Maintaining State Across Restarts, page A-16](#)
- [Connection-Based and Unconditional Blocking, page A-17](#)
- [Blocking with Cisco Firewalls, page A-18](#)
- [Blocking with Catalyst Switches, page A-19](#)

Understanding the ARC

The main responsibility of the ARC is to block events. When it responds to a block, it either interacts with the devices it is managing directly to enable the block or it sends a block request through the Control Transaction Server to a master blocking sensor. The web server on the master blocking sensor receives the control transaction and passes it to the Control Transaction Server, which passes it to the ARC. The ARC on the master blocking sensor then interacts with the devices it is managing to enable the block.

Figure A-4 illustrates the ARC.

Figure A-4 ARC



Note

An ARC instance can control 0, 1, or many network devices. The ARC does not share control of any network device with other ARC applications, IPS management software, other network management software, or system administrators. Only one ARC instance is allowed to run on a given sensor.

The ARC initiates a block in response to one of the following:

- An alert event generated from a signature that is configured with a block action
- A block configured manually through the CLI, IDM, or IME, or ASDM.
- A block configured permanently against a host or network address

When you configure the ARC to block a device, it initiates either a Telnet or SSH connection with the device. The ARC maintains the connection with each device. After the block is initiated, the ARC pushes a new set of configurations or ACLs (one for each interface direction) to each controlled device. When a block is completed, all configurations or ACLs are updated to remove the block.

ARC Features

The ARC has the following features:

- Communication through Telnet and SSH 1.5 with 3DES (the default) or DES encryption

Only the protocol specified in the ARC configuration for that device is attempted. If the connection fails for any reason, the ARC attempts to reestablish it.

- Preexisting ACLs on routers and VACLs on switches

If a preexisting ACL exists on a router interface or direction that is controlled by the ARC, you can specify that this ACL be merged into the ARC-generated configuration, either before any blocks by specifying a preblock ACL or after any blocks by specifying a postblock ACL. The Catalyst 6000 VACL device types can have a preblock and postblock VACL specified for each interface that the ARC controls. The firewall device types use a different API to perform blocks and the ARC does not have any effect on preexisting ACLs on the firewalls.



Note Catalyst 5000 RSM and Catalyst 6000 MSFC2 network devices are supported in the same way as Cisco routers.

- Forwarding blocks to a list of remote sensors

The ARC can forward blocks to a list of remote sensors, so that multiple sensors can in effect collectively control a single network device. Such remote sensors are referred to as master blocking sensors.

- Specifying blocking interfaces on a network device

You can specify the interface and direction where blocking is performed in the ARC configuration for routers. You can specify the interface where blocking is performed in the VACL configuration. The ARC can simultaneously control up to 250 interfaces.



Note Cisco firewalls do not block based on interface or direction, so this configuration is never specified for them.

- Blocking hosts or networks for a specified time

The ARC can block a host or network for a specified number of minutes or indefinitely. The ARC determines when a block has expired and unblocks the host or network at that time.

- Logging important events

The ARC writes a confirmation event when block or unblock actions are completed successfully or if any errors occur. The ARC also logs important events such as loss and recovery of a network device communication session, configuration errors, and errors reported by the network device.

- Maintaining the blocking state across ARC restarts

The ARC reapplies blocks that have not expired when a shutdown or restart occurs. The ARC removes blocks that have expired while it was shut down.



Note The ARC can only maintain the blocking state successfully if no one changes the system time while the application is shut down.

- Maintaining blocking state across network device restarts
The ARC reapplies blocks and removes expired blocks as needed whenever a network device is shut down and restarted. The ARC is not affected by simultaneous or overlapping shutdowns and restarts of the ARC.
- Authentication and authorization
The ARC can establish a communications session with a network device that uses AAA authentication and authorization including the use of remote TACACS+ servers.
- Two types of blocking
The ARC supports host blocks and network blocks. Host blocks are connection based or unconditional. Network blocks are always unconditional.
- NAT addressing
The ARC can control network devices that use a NAT address for the sensor. If you specify a NAT address when you configure a network device, that address is used instead of the local IP address when the sensor address is filtered from blocks on that device.
- Single point of control
The ARC does not share control of network devices with administrators or other software. If you must update a configuration, shut down ARC until the change is complete. You can enable or disable the ARC through the CLI or any Cisco IPS manager. When the ARC is reenabled, it completely reinitializes itself, including rereading the current configuration for each controlled network device.



Note We recommend that you disable the ARC from blocking when you are configuring any network device, including firewalls.

- Maintains up to 250 active blocks at any given time
The ARC can maintain up to 250 active blocks at a time. Although the ARC can support up to 65535 blocks, we recommend that you allow no more than 250 at a time.



Note The number of blocks is not the same as the number of interface and directions.

Supported Blocking Devices

The ARC can control the following devices:

- Cisco routers running Cisco IOS 11.2 or later



Note To perform rate limiting, the routers must be running Cisco IOS 12.3 or later.

- Catalyst 5000 series switches with Supervisor Engine software 5.3(1) or later running on the supervisor engine, and IOS 11.2(9)P or later running on the RSM.



Note You must have the RSM because blocking is performed on the RSM.

- Catalyst 6000 series switches with PFC installed running Catalyst software 5.3 or later

- Catalyst 6000 MSFC2 with Catalyst software 5.4(3) or later and Cisco IOS 12.1(2)E or later on the MSFC2
- Cisco ASA 5500 series models: ASA 5510, ASA 5520, and ASA 5540
- FWSM



Note The FWSM cannot block in multi-mode admin context.

ACLs and VACLs

If you want to filter packets on an interface or direction that the ARC controls, you can configure the ARC to apply an ACL before any blocks (preblock ACL) and to apply an ACL after any blocks (postblock ACL). These ACLs are configured on the network device as inactive ACLs. You can define preblock and postblock ACLs for each interface and direction. The ARC retrieves and caches the lists and merges them with the blocking ACEs whenever it updates the active ACL on the network device. In most cases, you will want to specify a preexisting ACL as the postblock ACL so that it does not prevent any blocks from taking effect. ACLs work by matching a packet to the first ACE found. If this first ACE permits the packet, a subsequent deny statement will not be found.

You can specify different preblock and postblock ACLs for each interface and direction, or you can reuse the same ACLs for multiple interfaces and directions. If you do not want to maintain a preblock list, you can use the never block option and always block hosts and networks by using existing configuration statements. A forever block is a normal block with a timeout value of -1.

The ARC only modifies ACLs that it owns. It does not modify ACLs that you have defined. The ACLs maintained by ARC have a specific format that should not be used for user-defined ACLs. The naming convention is **IPS_<interface_name>_[in | out]_[0 | 1]**. <interface_name> corresponds to the name of the blocking interface as given in the ARC configuration.

For Catalyst switches, it is a blocking interface VLAN number. Do not use these names for preblock and postblock ACLs. For Catalyst 6000 VACLs, you can specify a preblock and postblock VACL and only the interface is specified (direction is not used in VLANs). For firewalls, you cannot use preblock or postblock ACLs because the firewall uses a different API for blocking. Instead you must create ACLs directly on the firewalls.

Maintaining State Across Restarts

When the sensor shuts down, the ARC writes all blocks and rate limits (with starting timestamps) to a local file (nac.shun.txt) that is maintained by the ARC. When the ARC starts, this file is used to determine if any block updates should occur at the controlled network devices. Any unexpired blocks found in the file are applied to the network devices at startup. When the ARC shuts down, no special actions on the ACLs are taken even if outstanding blocks are in effect. The nac.shun.txt file is accurate only if the system time is not changed while the ARC is not running.



Caution

Do not make manual changes to the nac.shun.txt file.

The following scenarios demonstrate how the ARC maintains state across restarts.

Scenario 1

There are two blocks in effect when the ARC stops and one of them expires before the ARC restarts. When the ARC restarts, it first reads the `nac.shun.txt` file. It then reads the preblock and postblock ACLs or VACLs. The active ACL or VACL is built in the following order:

1. The **allow** *sensor_ip_address* command (unless the **allow sensor shun** command has been configured)
2. Preblock ACL
3. The **always block** command entries from the configuration
4. Unexpired blocks from `nac.shun.txt`
5. Postblock ACL

When a host is specified as never block in the ARC configuration, it does not get translated into permit statements in the ACL. Instead, it is cached by the ARC and used to filter incoming `addShunEvent` events and `addShunEntry` control transactions.

Scenario 2

There are no preblock or postblock ACLs specified, but there is an existing active ACL. The new ACL is built in the following order:

1. The **allow** *sensor_ip_address* command (unless the **allow sensor shun** command has been configured)
2. The **always block** command entries from the configuration
3. Unexpired blocks from `nac.shun.txt`
4. The **permit IP any any** command

Connection-Based and Unconditional Blocking

The ARC supports two types of blocking for hosts and one type of blocking for networks. Host blocks are connection-based or unconditional. Network blocks are always unconditional.

When a host block is received, the ARC checks for the `connectionShun` attribute on the host block. If `connectionShun` is set to true, the ARC performs connection blocking. Any host block can contain optional parameters, such as destination IP address, source port, destination port, and protocol. For a connection block to take place, at least the source and destination IP address must be present. If the source port is present on a connection block, it is ignored and not included in the block.

Under the following conditions, the ARC forces the block to be unconditional, converting the block from connection type if necessary:

- A block of any type is active for a specified source IP address
- A new block of any type is received for that source IP address
- The new block differs in any of its optional parameters (except the source port) from the old block

When a block is updated (for example, when a new block arrives while an existing block for that source IP address or network is already in effect), the remaining minutes of the existing block are determined. If the time for the new block is less than or equal to the remaining minutes, no action is taken. Otherwise, the new block timeout replaces the existing block timeout.

**Caution**

Cisco firewalls do not support connection blocking of hosts. When a connection block is applied, the firewall treats it like an unconditional block. Cisco firewalls also do not support network blocking. ARC never tries to apply a network block to a Cisco firewall.

Blocking with Cisco Firewalls

The ARC performs blocks on firewalls using the **shun** command. The **shun** command has the following formats:

- To block an IP address:
`shun srcip [destination_ip_address source_port destination_port [port]]`
- To unblock an IP address:
`no shun ip`
- To clear all blocks:
`clear shun`
- To show active blocks or to show the global address that was actually blocked:
`show shun [ip_address]`

The ARC uses the response to the **show shun** command to determine whether the block was performed. The **shun** command does not replace existing ACLs, conduits, or outbound commands, so there is no need to cache the existing firewall configuration, nor to merge blocks into the firewall configuration.

**Caution**

Do not perform manual blocks or modify the existing firewall configuration while ARC is running.

If the **block** command specifies only the source IP address, existing active TCP connections are not broken, but all incoming packets from the blocked host are dropped.

When the ARC first starts up, the active blocks in the firewall are compared to an internal blocking list. Any blocks that do not have a corresponding internal list entry are removed.

The ARC supports authentication on a firewall using local usernames or a TACACS+ server. If you configure the firewall to authenticate using AAA but without the TACACS+ server, the ARC uses the reserved username *pix* for communications with the firewall.

If the firewall uses a TACACS+ server for authentication, you use a TACACS+ username. In some firewall configurations that use AAA logins, you are presented with three password prompts: the initial firewall password, the AAA password, and the enable password. The ARC requires that the initial firewall password and the AAA password be the same.

When you configure a firewall to use NAT or PAT and the sensor is checking packets on the firewall outside network, if you detect a host attack that originates on the firewall inside network, the sensor tries to block the translated address provided by the firewall. If you are using dynamic NAT addressing, the block can be ineffective or cause innocent hosts to be blocked. If you are using PAT addressing, the firewall could block the entire inside network. To avoid these situations, position your sensor on the inside interface or do not configure the sensor to block.

Blocking with Catalyst Switches

Catalyst switches with a PFC filter packets using VACLs. VACLs filter all packets between VLANs and within a VLAN. MSFC router ACLs are supported when WAN cards are installed and you want the sensor to control the interfaces through the MSFC2.

**Note**

An MSFC2 card is not a required part of a Catalyst switch configuration for blocking with VACLs.

**Caution**

When you configure the ARC for the Catalyst switch, do not specify a direction with the controlled interface. The interface name is a VLAN number. Preblock and postblock lists should be VACLs.

The following commands apply to the Catalyst VACLs:

- To view an existing VACL:

```
show security acl info acl_name
```
- To block an address (*address_spec* is the same as used by router ACLs):

```
set security acl ip acl_name deny address_spec
```
- To activate VACLs after building the lists:

```
commit security acl all
```
- To clear a single VACL:

```
clear security acl map acl_name
```
- To clear all VACLs:

```
clear security acl map all
```
- To map a VACL to a VLAN:

```
set sec acl acl_name vlans
```

Logger

The sensor logs all events (alert, error, status, and debug messages) in a persistent, circular buffer. The sensor also generates IP logs. The messages and IP logs are accessible through the CLI, and IDM, and ASDM.

The IPS applications use the Logger to log messages. The Logger sends log messages at any of five levels of severity: debug, timing, warning, error, and fatal. The Logger writes the log messages to `/usr/cids/idsRoot/log/main.log`, which is a circular text file. New messages overwrite older messages when the file reaches its maximum size; therefore the last message written may not appear at the end of the `main.log`. Search for the string “= END OF FILE =” to locate the last line written to the `main.log`.

The `main.log` is included in the **show tech-support** command output. If the message is logged at warning level or above (error or fatal), the Logger converts the message to an `evError` event (with the corresponding error severity) and inserts it in the Event Store.

The Logger receives all syslog messages, except cron messages, that are at the level of informational and above (`*.info;cron.none`), and inserts them in to the Event Store as `evErrors` with the error severity set to Warning. The Logger and application logging are controlled through the service logger commands.

The Logger can control what log messages are generated by each application by controlling the logging severity for different logging zones. You would only access the individual-zone-control of the logger service at the request and supervision of a TAC engineer or developer. For troubleshooting purposes, TAC might request that you turn on debug logging.

AuthenticationApp

This section describes the AuthenticationApp, and contains the following topics:

- [Understanding the AuthenticationApp, page A-20](#)
- [Authenticating Users, page A-20](#)
- [Configuring Authentication on the Sensor, page A-20](#)
- [Managing TLS and SSH Trust Relationships, page A-21](#)

Understanding the AuthenticationApp

The AuthenticationApp has the following responsibilities:

- To authenticate the identity of a user
- To administer the accounts, privileges, keys, and certificates of the user
- To configure which authentication methods are used by the AuthenticationApp and other access services on the sensor

Authenticating Users

You must configure authentication on the sensor to establish appropriate security for user access. When you install a sensor, an initial cisco account with an expired password is created. A user with administrative access to the sensor accesses the sensor through the CLI or an IPS manager, such as the IDM or the ASDM, by logging in to the sensor using the default administrative account (**cisco**). In the CLI, the administrator is prompted to change the password. IPS managers initiate a `setEnableAuthenticationTokenStatus` control transaction to change the password of an account.

Through the CLI or an IPS manager, the administrator configures which authentication method is used, such as username and password or an SSH authorized key. The application servicing the administrator initiates a `setAuthenticationConfig` control transaction to establish the authentication configuration.

The authentication configuration includes a login attempt limit value that is used to specify how account locking is handled. Account locking is invoked when the number of consecutive failed login attempts for a given account exceeds the login attempt limit value. After an account is locked, all further attempts to log in to that account are rejected. The account is unlocked by resetting the authentication token of the account using the `setEnableAuthenticationTokenStatus` control transaction. The account locking feature is disabled when the login attempt limit value is set to zero.

The administrator can add additional user accounts either through the CLI or an IPS manager.

Configuring Authentication on the Sensor

When a user tries to access the sensor through a service such as web server or the CLI, the identity of the user must be authenticated and the privileges of the user must be established. The service that is providing access to the user initiates an `execAuthenticateUser` control transaction request to the

AuthenticationApp to authenticate the identity of the user. The control transaction request typically includes the username and a password, or the identity of the user can be authenticated using an SSH authorized key.

The AuthenticationApp responds to the `execAuthenticateUser` control transaction request by attempting to authenticate the identity of the user. The AuthenticationApp returns a control transaction response that contains the authentication status and privileges of the user. If the identity of the user cannot be authenticated, the AuthenticationApp returns an unauthenticated status and anonymous user privileges in the control transaction response. The control transaction response also indicates if the account password has expired. User interface applications that authenticate users by initiating an `execAuthenticateUser` control transaction prompt the user to change the password.

The AuthenticationApp uses the underlying operating system to confirm the identity of a user. All the IPS applications send control transactions to the AuthenticationApp, which then uses the operating system to form its responses.

Remote shell services, Telnet and SSH, are not IPS applications. They call the operating system directly. If the user is authenticated, it launches the IPS CLI. In this case, the CLI sends a special form of the `execAuthenticateUser` control transaction to determine the privilege level of the logged-in user. The CLI then tailors the commands it makes available based on this privilege level.

Managing TLS and SSH Trust Relationships

Encrypted communications over IP networks provide data privacy by making it impossible for a passive attacker to discover from the packets exchanged alone the secret key needed to decrypt the data in the packets.

However, an equally dangerous attack vector is for an imposter to pretend to be the server end of the connection. All encryption protocols provide a means for clients to defend themselves from these attacks. IPS supports two encryption protocols, SSH and TLS, and the AuthenticationApp helps manage trust when the sensor plays either the client or server role in encrypted communications.

The IPS web server and SSH server are server endpoints of encrypted communications. They protect their identities with a private key and offer a public key to clients that connect to them. For TLS this public key is included inside an X.509 certificate, which includes other information. Remote systems that connect to the sensor should verify that the public key received during connection establishment is the key they expect.

Clients must maintain a list of trusted public keys to protect themselves from man-in-the-middle attacks. The exact procedure by which this trust is established varies depending on the protocol and client software. In general, the client displays a fingerprint of 16 or 20 bytes. The human operator who is configuring the client to establish trust should use an out-of-band method to learn the key fingerprints of the server before attempting to establish trust. If the fingerprints match, the trust relationship is established and henceforth the client can automatically connect with that server and be confident that the remote server is not an imposter.

You can use the **`show ssh server-key`** and **`show tls fingerprint`** to display the key fingerprints of the sensor. By recording the output of these commands when directly connected to the sensor console, you can reliably use this information to confirm the identity of the sensor over the network later when establishing trust relationships.

For example, when you initially connect to a sensor through the Microsoft Internet Explorer web browser, a security warning dialog box indicates that the certificate is not trusted. Using the user interface of Internet Explorer, you can inspect the certificate thumbprint, a value that should exactly match the SHA1 fingerprint displayed by the **`show tls fingerprint`** command. After verifying this, add this certificate to the list of trusted CAs of the browser to establish permanent trust.

Each TLS client has different procedures for establishing this trust. The sensor itself includes a TLS client that is used to send control transactions to other sensors and download upgrades and configuration files from other TLS web servers. Use the **tls trusted-host** command to establish trust of the TLS servers with which the sensor communicates.

Similarly, the sensor includes an SSH client that is used to communicate with managed network devices, download upgrades, and copy configurations and support files to remote hosts. Use the **ssh host-key** command to establish trust relationships with the SSH servers the sensor will contact.

You can manage the list of TLS trusted certificates and SSH known hosts through the commands **service trusted-certificates** and **service ssh-known-hosts**.

X.509 certificates include additional information that can increase the security of the trust relationship; however, these can lead to confusion. For example, an X.509 certificate includes a validity period during which the certificate can be trusted. Typically this period is a number of years starting at the moment the certificate is created. To ensure that an X.509 certificate is valid at the moment it is being used requires that the client system maintain an accurate clock.

X.509 certificates are also tied to a particular network address. Sensors fill this field with the IP address of the command and control interface of the sensor. Consequently, if you change the command and control IP address of the sensor, the X.509 certificate of the server is regenerated. You must reconfigure all clients on the network that trusted the old certificate to locate the sensor at its new IP address and trust the new certificate.

By using the SSH known hosts and TLS trusted certificates services in the AuthenticationApp, you can operate sensors at a high level of security.

Web Server

The web server provides SDEE support, which enables the sensor to report security events, receive IDIOM transactions, and serve IP logs. The web server supports HTTP 1.0 and 1.1. Communications with the web server often include sensitive information, such as passwords, that would severely compromise the security of the system if an attacker were able to eavesdrop. For this reason, sensors ship with TLS enabled. The TLS protocol is an encryption protocol that is compatible with SSL.

**Note**

We deprecated the RDEP event sever service in IPS 6.1, and deleted it from the IPS system architecture. The web server now uses the SDEE event server.

SensorApp

This section describes the SensorApp, and contains the following topics:

- [Understanding the SensorApp, page A-23](#)
- [Inline, Normalization, and Event Risk Rating Features, page A-24](#)
- [SensorApp New Features, page A-25](#)
- [Packet Flow, page A-25](#)
- [Signature Event Action Processor, page A-26](#)

Understanding the SensorApp

The SensorApp performs packet capture and analysis. Policy violations are detected through signatures in the SensorApp and the information about the violations is forwarded to the Event Store in the form of an alert. Packets flow through a pipeline of processors fed by a producer designed to collect packets from the network interfaces on the sensor. Event actions can be associated with an event risk rating threshold that must be surpassed for the actions to take place. Some of the processors call inspectors to perform signature analysis. All inspectors can call the alarm channel to produce alerts as needed.

The SensorApp supports the following processors:

- Time Processor—This processor processes events stored in a time-slice calendar. Its primary task is to make stale database entries expire and to calculate time-dependent statistics.
- Deny Filters Processor—This processor handles the deny attacker functions. It maintains a list of denied source IP addresses. Each entry in the list expires based on the global deny timer, which you can configure in the virtual sensor configuration.
- Signature Event Action Processor—This processor processes event actions. Event actions can be associated with an event risk rating threshold that must be surpassed for the actions to take place. It supports the following event actions:
 - Reset TCP flow
 - IP log
 - Deny packets
 - Deny flow
 - Deny attacker
 - Alert
 - Block host
 - Block connection
 - Generate SNMP trap
 - Capture trigger packet
- Statistics Processor—This processor keeps track of system statistics such as packet counts and packet arrival rates.
- Layer 2 Processor—This processor processes layer 2-related events. It also identifies malformed packets and removes them from the processing path. You can configure actionable events for detecting malformed packets such as alert, capture packet, and deny packet. The layer 2 processor updates statistics about packets that have been denied because of the policy you have configured.
- Database Processor—This processor maintains the signature state and flow databases.
- Fragment Reassembly Processor—This processor reassembles fragmented IP datagrams. It is also responsible for normalization of IP fragments when the sensor is in inline mode.
- Stream Reassembly Processor—This processor reorders TCP streams to ensure the arrival order of the packets at the various stream-based inspectors. It is also responsible for normalization of the TCP stream. The normalizer engine lets you enable or disable alert and deny actions.

The TCP Stream Reassembly Processor normalizer has a hold-down timer, which lets the stream state rebuild after a reconfiguration event. You cannot configure the timer. During the hold-down interval, the system synchronizes stream state on the first packet in a stream that passes through the system. When the hold down has expired, SensorApp enforces your configured policy. If this policy calls for a denial of streams that have not been opened with a 3-way handshake, established streams

that were quiescent during the hold-down period will not be forwarded and will be allowed to timeout. Those streams that were synchronized during the hold-down period are allowed to continue.

- **Signature Analysis Processor**—This processor dispatches packets to the inspectors that are not stream-based and that are configured for interest in the packet in process.
- **Slave Dispatch Processor**—A process found only on dual CPU systems.

The SensorApp also supports the following units:

- **Analysis Engine**—The Analysis Engine handles sensor configuration. It maps the interfaces and also the signature and alarm channel policy to the configured interfaces.
- **Alarm Channel**—The Alarm Channel processes all signature events generated by the inspectors. Its primary function is to generate alerts for each event it is passed.

Inline, Normalization, and Event Risk Rating Features

The SensorApp contains the following inline, normalization, and event risk rating features:

- **Processing packets inline**

When the sensor is processing packets in the data path, all packets are forwarded without any modifications unless explicitly denied by policy configuration. Because of TCP normalization it is possible that some packets will be delayed to ensure proper coverage. When policy violations are encountered, the SensorApp allows for the configuration of actions. Additional actions are available in inline mode, such as deny packet, deny flow, and deny attacker.

All packets that are unknown or of no interest to the IPS are forwarded to the paired interface with no analysis. All bridging and routing protocols are forwarded with no participation other than a possible deny due to policy violations. There is no IP stack associated with any interface used for inline (or promiscuous) data processing. The current support for 802.1q packets in promiscuous mode is extended to inline mode.

- **IP normalization**

Intentional or unintentional fragmentation of IP datagrams can serve to hide exploits making them difficult or impossible to detect. Fragmentation can also be used to circumvent access control policies like those found on firewalls and routers. And different operating systems use different methods to queue and dispatch fragmented datagrams. If the sensor has to check for all possible ways that the end host will reassemble the datagrams, it makes the sensor vulnerable to denial of service attacks. Reassembling all fragmented datagrams inline and only forwarding completed datagrams, refragmenting the datagram if necessary, is the solution to this problem. The IP Fragmentation Normalization unit performs this function.

- **TCP normalization**

Through intentional or natural TCP session segmentation, some classes of attacks can be hidden. To make sure policy enforcement can occur with no false positives and false negatives, the state of the two TCP endpoints must be tracked and only the data that is actually processed by the real host endpoints should be passed on. Overlaps in a TCP stream can occur, but are extremely rare except for TCP segment retransmits. Overwrites in the TCP session should not occur. If overwrites do occur, someone is intentionally trying to elude the security policy or the TCP stack implementation is broken. Maintaining full information about the state of both endpoints is not possible unless the sensor acts as a TCP proxy. Instead of the sensor acting as a TCP proxy, the segments will be ordered properly and the normalizer will look for any abnormal packets associated with evasion and attacks.

- Event risk rating

Event risk rating helps reduce false positives from the system and gives you more control over what causes an alarm. The event risk rating incorporates the following additional information beyond the detection of a potentially malicious action:

 - Severity of the attack if it were to succeed
 - Fidelity of the signature
 - Relevance of the potential attack with respect to the target host
 - Overall value of the target host

SensorApp New Features

The SensorApp contains the following new features:

- Policy table—Provides a list of risk category settings (high, medium, and low).
- Evasion protection—Lets an inline interface mode sensor switch from strict mode to asymmetric mode for the Normalizer.
- Sensor health meter—Provides sensor-wide health statistics.
- Top services—Provides the top ten instances of the TCP, UDP, ICMP, and IP protocols.
- Security meter—Profiles alerts into threat categories and reports this information in red, yellow, and green buckets. You can configure the transition points for these buckets.
- Clear Flow state—Lets you clear the database, which causes the sensor to start fresh just as in a restart.
- Restart status—Reports periodically the current start and restart stages of the sensor.

Packet Flow

Packets are received by the NIC and placed in the kernel user-mapped memory space by the IPS-shared driver. The packet is prepended by the IPS header. Each packet also has a field that indicates whether to pass or deny the packet when it reaches Signature Event Action Processor.

The producer pulls packets from the shared-kernel user-mapped packet buffer and calls the process function that implements the processor appropriate to the sensor model. The following orders occur:

- Single processor execution

Time Processor --> Layer 2 Processor --> Deny Filters Processor --> Fragment Reassembly Processor --> Statistics Processor --> Database Processor --> Signature Analysis Processor --> Stream Reassembly Processor --> Signature Event Action Processor
- Dual processor execution

Execution Thread 1 Time Processor --> Layer 2 Processor --> Deny Filters Processor --> Fragment Reassembly Processor --> Statistics Processor --> Database Processor --> Signature Analysis Processor --> Slave Dispatch Processor --> | Execution Thread 2 Database Processor --> Stream Reassembly Processor --> Signature Event Action Processor

Signature Event Action Processor

The Signature Event Action Processor coordinates the data flow from the signature event in the Alarm Channel to processing through the Signature Event Action Override, the Signature Event Action Filter, and the Signature Event Action Handler. It consists of the following components:

- Alarm Channel—The unit that represents the area to communicate signature events from the SensorApp inspection path to signature event handling.
- Signature Event Action Override—Adds actions based on the risk rating value. Signature Event Action Override applies to all signatures that fall in the range of the configured risk rating threshold. Each Signature Event Action Override is independent and has a separate configuration value for each action type.
- Signature Event Action Filter—Subtracts actions based on the signature ID, addresses, and risk rating of the signature event. The input to the Signature Event Action Filter is the signature event with actions possibly added by the Signature Event Action Override.



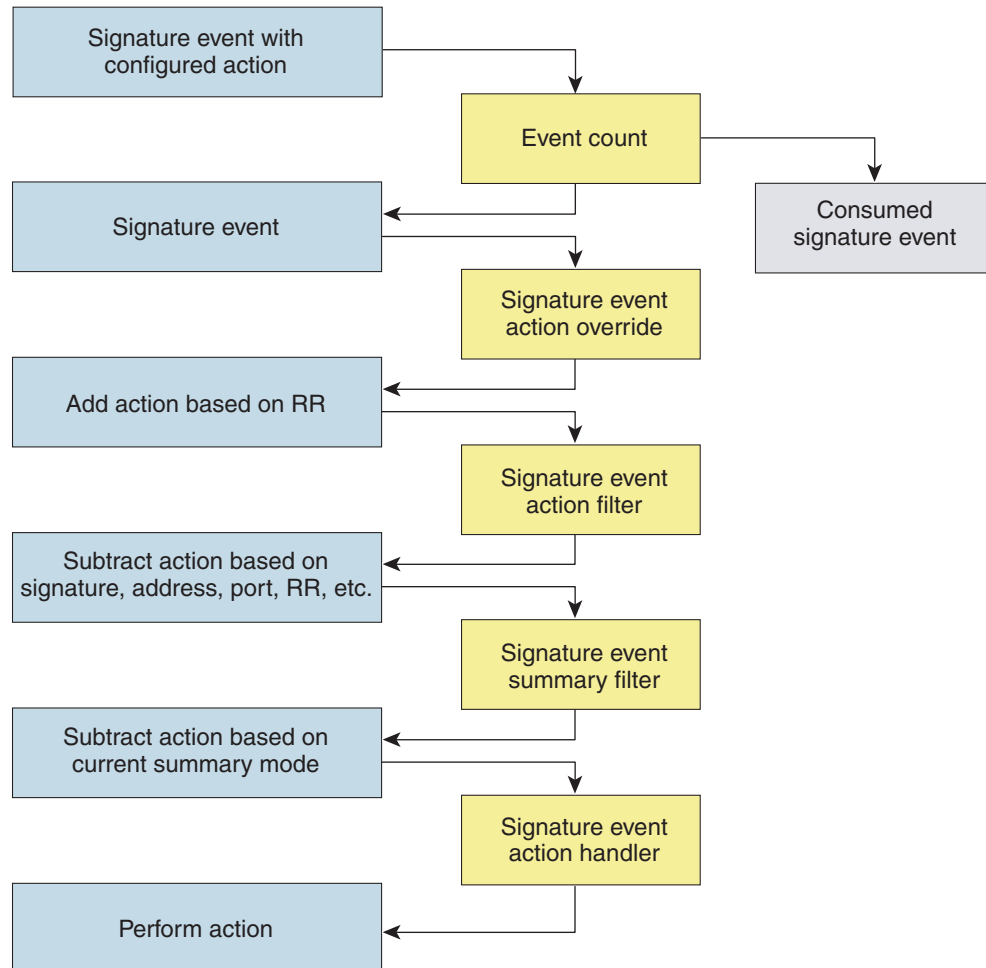
Note The Signature Event Action Filter can only subtract actions, it cannot add new actions.

The following parameters apply to the Signature Event Action Filter:

- Signature ID
 - Subsignature ID
 - Attacker address
 - Attacker port
 - Victim address
 - Victim port
 - Risk rating threshold range
 - Actions to subtract
 - Sequence identifier (optional)
 - Stop-or-continue bit
 - Enable action filter line bit
 - Victim OS relevance or OS relevance
- Signature Event Action Handler—Performs the requested actions. The output from the Signature Event Action Handler is the actions being performed and possibly an evIdsAlert written to the Event Store.

Figure A-5 illustrates the logical flow of the signature event through the Signature Event Action Processor and the operations performed on the action for this event. It starts with the signature event with configured action received in the Alarm Channel and flows top to bottom as the signature event passes through the functional components of the Signature Event Action Processor.

Figure A-5 Signature Event Through Signature Event Action Processor



132188

CollaborationApp

This section describes the CollaborationApp, and contains the following sections:

- [Understanding the CollaborationApp, page A-28](#)
- [Update Components, page A-28](#)
- [Error Events, page A-29](#)

Understanding the CollaborationApp

The CollaborationApp is a peer of the MainApp and the SensorApp. It interfaces with them using various interprocess communication technologies, such as IDAPI control transactions, semaphores, shared memory, and file exchange.

Reputation updates are exchanged between the Global Correlation server and the CollaborationApp. The CollaborationApp communicates with the sensors using four update components:

- Set of rules score weight values
- Set of IP addresses and address ranges, which together with the rules and alerts provide the information needed to calculate reputation scores
- List of IP addresses and address ranges for which traffic should always be denied
- Network participation configuration, which allows the server to control the rate at which sensors send telemetry data to the server

The sensor sends collaboration information to the Network Participation server. The sensor queries the Global Correlation server for a list of what collaboration updates are available and from which Global Correlation server to download the update files.



Note

The SensorApp starts before the CollaborationApp, but they initialize asynchronously. Therefore, it is possible that the Reputation Update server may download and attempt to apply one or more global correlation updates before the SensorApp is ready to accept the update. The update server may download and partially process the update, but it must wait until the SensorApp is ready before it can commit the update.

For More Information

For detailed information on global correlation and how to configure it, see [Chapter 10, “Configuring Global Correlation.”](#)

Update Components

The Global Correlation Update client exchanges manifests with the Global Correlation Update server. It parses the server manifest to determine what new updates are available for download and where they reside, and then builds a list of updates to be installed. If all updates are applied successfully, then the Global Correlation Update client commits the applied updates for each component, notifies SensorApp that new updates are available, and updates the client manifest to reflect the latest committed updates for each component.

The client manifest contains the UDI of the sensor, which includes the serial number of the sensor, and an encrypted shared secret that the server uses to verify the sensor is an authentic Cisco IPS sensor. The server manifest contains a list of update files available for each component. For each update file in the list, the server manifest contains data, such as the update version, type, order, location, file transfer protocol, and so forth.

There are two types of updates files: a full update file that replaces any existing data in the database of the component, and an incremental update that modifies the existing reputation data by adding, deleting, or replacing information. When all update files have been applied for all components, the temporary databases are committed by replacing the working databases.

Authentication and authorization are achieved through the secret encryption mechanism and decryption key management. The Global Correlation Update server authenticates the sensor using the shared secret encryption mechanism contained in the client manifest. The Global Correlation Update client authorizes sensors using decryption key management. Sensors that have been authenticated by the Global Correlation Update server are sent valid keys in the server manifest so that they can decrypt the update files.

**Caution**

You receive a warning message if you have enabled global correlation, but you have not configured a DNS or HTTP proxy server. This warning is a reminder to either disable global correlation or add a DNS or HTTP proxy server.

For More Information

For the procedure for adding a DNS or proxy server to support global correlation, see [Changing Network Settings, page 4-2](#).

Error Events

Whenever a global correlation update fails, an evError event is generated. The error message is included in sensor statistics. The following conditions result in a status message with the severity of Error:

- The sensor is unlicensed
- No DNS or HTTP proxy server is configured
- The manifest exchange failed
- An update file download failed
- Applying or committing the update failed

An evError event is generated with the severity level of Warning if you edit and save either the host or global correlation configurations so that global correlation is enabled, but no DNS or HTTP proxy servers are configured.

For More Information

For the procedure for displaying sensor statistics, see [Displaying Statistics, page 17-30](#).

SwitchApp

The 4500 series sensors have a built-in switch that provides the external monitoring interfaces of the sensor. The SwitchApp is part of the IPS 4500 series design that enables the InterfaceApp and sensor initialization scripts to communicate with and control the switch. Any application that needs to get or set information on the switch must communicate with the SwitchApp. Additionally the SwitchApp implements the following:

- Detects bypass—When the SensorApp is not monitoring, the SwitchApp places the switch in bypass mode and then back to inspection mode once the SensorApp is up and running normally.
- Collects port statistics—The SwitchApp monitors the switch and collects statistics on the external interfaces of the switch for reporting by InterfaceApp.

- Handles the external interface configuration—When you update the interface configuration, the configuration is sent to the InterfaceApp, which updates the interface configuration for SwitchApp, which then forwards that configuration on to the switch.

CLI

The CLI provides the sensor user interface for all direct node access such as Telnet, SSH, and serial interface. You configure the sensor applications with the CLI. Direct access to the underlying OS is allowed through the service role. This section describes the IPS CLI, and contains the following topics:

- [User Roles, page A-30](#)
- [Service Account, page A-31](#)

User Roles



Caution

You should carefully consider whether you want to create a service account. The service account provides shell access to the system, which makes the system vulnerable. However, you can use the service account to create a password if the administrator password is lost. Analyze your situation to decide if you want a service account existing on the system.

There are four user roles:

- **Viewer**—Can view configuration and events, but cannot modify any configuration data except their user passwords.
- **Operator**—Can view everything and can modify the following options:
 - Signature tuning (priority, disable or enable)
 - Virtual sensor definition
 - Managed routers
 - Their user passwords
- **Administrator**—Can view everything and can modify all options that operators can modify in addition to the following:
 - Sensor addressing configuration
 - List of hosts allowed to connect as configuration or viewing agents
 - Assignment of physical sensing interfaces
 - Enable or disable control of physical interfaces
 - Add and delete users and passwords
 - Generate new SSH host keys and server certificates
- **Service**—Only one user with service privileges can exist on a sensor. The service user cannot log in to the IDM or the IME. The service user logs in to a bash shell rather than the CLI.

The service role is a special role that allows you to bypass the CLI if needed. Only one service account is allowed. You should only create an account with the service role for troubleshooting purposes. Only a user with administrator privileges can edit the service account.

When you log in to the service account, you receive the following warning:

```
***** WARNING *****
UNAUTHORIZED ACCESS TO THIS NETWORK DEVICE IS PROHIBITED.
This account is intended to be used for support and troubleshooting purposes only.
Unauthorized modifications are not supported and will require this device to be
re-imaged to guarantee proper operation.
*****
```



Note

For IPS 5.0 and later, you can no longer remove the **cisco** account. You can disable it using the **no password cisco** command, but you cannot remove it. To use the **no password cisco** command, there must be another administrator account on the sensor. Removing the **cisco** account through the service account is not supported. If you remove the **cisco** account through the service account, the sensor most likely will not boot up, so to recover the sensor you must reinstall the sensor system image.

Service Account

The service account is a support and troubleshooting tool that enables the TAC to log in to a native operating system shell rather than the CLI shell. It does not exist on the sensor by default. You must create it so that it is available for the TAC to use for troubleshooting your sensor.

Only one service account is allowed per sensor and only one account is allowed a service role. When the password of the service account is set or reset, the password of the root account is set to the same password. This allows the service account user to su to root using the same password. When the service account is removed, the password of the root account is locked.

The service account is not intended to be used for configuration purposes. Only modifications made to the sensor through the service account under the direction of the TAC are supported. Cisco Systems does not support the addition and/or running of an additional service to the operating system through the service account, because it affects proper performance and proper functioning of the other IPS services. The TAC does not support a sensor on which additional services have been added.

You can track logins to the service account by checking the log file `/var/log/.tac`, which is updated with a record of service account logins.



Note

The Cisco IPS incorporates several troubleshooting features that are available through the CLI, IDM, or IME. The service account is not necessary for most troubleshooting situations. You may need to create the service account at the direction of TAC to troubleshoot a very unique problem. The service account lets you bypass the protections built into the CLI and allows root privilege access to the sensor, which is otherwise disabled. We recommend that you do not create a service account unless it is needed for a specific reason. You should remove the service account when it is no longer needed.

Communications

This section describes the communications protocols used by the Cisco IPS. It contains the following topics:

- [IDAPI, page A-32](#)
- [IDIOM, page A-32](#)
- [IDCONF, page A-33](#)

- [SDEE, page A-33](#)
- [CIDEE, page A-34](#)

IDAPI

IPS applications use an interprocess communication API called the IDAPI to handle internal communications. The IDAPI reads and writes event data and provides a mechanism for control transactions. The IDAPI is the interface through which all the applications communicate.

The SensorApp captures and analyzes the network traffic on its interfaces. When a signature is matched, the SensorApp generates an alert, which is stored in the Event Store. If the signature is configured to perform the blocking response action, the SensorApp generates a block event, which is also stored in the Event Store.

Figure A-6 illustrates the IDAPI interface.

Figure A-6 IDAPI



Each application registers to the IDAPI to send and receive events and control transactions. The IDAPI provides the following services:

- Control transactions
 - Initiates the control transaction.
 - Waits for the inbound control transaction.
 - Responds to the control transaction.
- IPS events
 - Subscribes to remote IPS events, which are stored in the Event Store when received.
 - Reads IPS events from the Event Store.
 - Writes IPS events to the Event Store.

The IDAPI provides the necessary synchronization mechanisms to guarantee atomic data accesses.

IDIOM

IDIOM is a data format standard that defines the event messages that are reported by the IPS as well as the operational messages that are used to configure and control intrusion detection systems. These messages consist of XML documents that conform to the IDIOM XML schema.

IDIOM supports two types of interactions: event and control transaction. Event interactions are used to exchange IPS events such as alerts. IDIOM uses two types of messages for event interactions: event and error messages. Control transactions provide a means for one host to initiate an action in, change the state of, or read the state of another host. Control transactions utilize four types of IDIOM messages: request, response, configuration, and error messages. Events and control transactions that are communicated between application instances within a host are known as local events or local control

transactions, or collectively, local IDIOM messages. Events and control transactions that are communicated between different hosts are known as remote events and remote control transactions, or collectively, remote IDIOM messages.



Note

IDIOM for the most part has been superseded by IDCONF, SDEE, and CIDEE.

IDCONF

The Cisco IPS manages its configuration using XML documents. IDCONF specifies the XML schema including the Cisco IPS control transactions. The IDCONF schema does not specify the contents of the configuration documents, but rather the framework and building blocks from which the configuration documents are developed. It provides mechanisms that let the IPS managers and CLI ignore features that are not configurable by certain platforms or functions through the use of the feature-supported attribute.

IDCONF messages are wrapped inside IDIOM request and response messages.

The following is an IDCONF example:

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<request xmlns="http://www.cisco.com/cids/idiom" schemaVersion="2.00">
  <editConfigDelta xmlns="http://www.cisco.com/cids/idconf">
    <component name="userAccount">
      <config typedefsVersion="2004-03-01" xmlns="http://www.cisco.com/cids/idconf">
        <struct>
          <map name="user-accounts" editOp="merge">
            <mapEntry>
              <key>
                <var name="name">cisco</var>
              </key>
              <struct>
                <struct name="credentials">
                  <var name="role">administrator</var>
                </struct>
              </struct>
            </mapEntry>
          </map>
        </struct>
      </config>
    </component>
  </editDefaultConfig>
</request>
```

SDEE

The Cisco IPS produces various types of events including intrusion alerts and status events. The IPS communicates events to clients such as management applications using the proprietary IPS-industry leading protocol, SDEE, which is a product-independent standard for communicating security device events. SDEE adds extensibility features that are needed for communicating events generated by various types of security devices.

Systems that use SDEE to communicate events to clients are referred to as SDEE providers. SDEE specifies that events can be transported using the HTTP or HTTP over SSL and TLS protocols. When HTTP or HTTPS is used, SDEE providers act as HTTP servers, while SDEE clients are the initiators of HTTP requests.

The IPS includes the web server, which processes HTTP or HTTPS requests. The web server uses run-time loadable servlets to process the different types of HTTP requests. Each servlet handles HTTP requests that are directed to the URL associated with the servlet. The SDEE server is implemented as a web server servlet.

The SDEE server only processes authorized requests. A request is authorized if it originates from a web server to authenticate the identity of the client and determine the privilege level of the client.

CIDEE

CIDEE specifies the extensions to SDEE that are used by the Cisco IPS. The CIDEE standard specifies all possible extensions that are supported by the Cisco IPS. Specific systems may implement a subset of CIDEE extensions. However, any extension that is designated as being required **MUST** be supported by all systems. CIDEE specifies the Cisco IPS-specific security device events and the IPS extensions to the SDEE `evIdsAlert` element.

CIDEE supports the following events:

- `evError`—Error event
Generated by the CIDEE provider when the provider detects an error or warning condition. The `evError` event contains error code and textual description of the error.
- `evStatus`—Status message event
Generated by CIDEE providers to indicate that something of potential interest occurred on the host. Different types of status messages can be reported in the status event—one message per event. Each type of status message contains a set of data elements that are specific to the type of occurrence that the status message is describing. The information in many of the status messages are useful for audit purposes. Errors and warnings are not considered status information and are reported using `evError` rather than `evStatus`.
- `evShunRqst`—Block request event
Generated to indicate that a block action is to be initiated by the service that handles network blocking.

The following is a CIDEE extended event example:

```
<sd:events xmlns:cid="http://www.cisco.com/cids/2004/04/cidee"
xmlns:sd="http://example.org/2003/08/sdee">
  <sd:evIdsAlert eventId="1042648730045587005" vendor="Cisco" severity="medium">
    <sd:originator>
      <sd:hostId>Beta4Sensor1</sd:hostId>
      <cid:appName>sensorApp</cid:appName>
      <cid:appInstanceId>8971</cid:appInstanceId>
    </sd:originator>
    <sd:time offset="0" timeZone="UTC">1043238671706378000</sd:time>
    <sd:signature description="IOS Udp Bomb" id="4600" cid:version="S37">
      <cid:subsigId>0</cid:subsigId>
    </sd:signature> ...
  </sd:evIdsAlert>
</sd:events>
```

Cisco IPS File Structure

The Cisco IPS has the following directory structure:

- `/usr/cids/idsRoot`—Main installation directory.
- `/usr/cids/idsRoot/shared`—Stores files used during system recovery.

- /usr/cids/idsRoot/var—Stores files created dynamically while the sensor is running.
- /usr/cids/idsRoot/var/updates—Stores files and logs for update installations.
- /usr/cids/idsRoot/var/virtualSensor—Stores files used by SensorApp to analyze regular expressions.
- /usr/cids/idsRoot/var/eventStore—Contains the Event Store application.
- /usr/cids/idsRoot/var/core—Stores core files that are created during system crashes.
- /usr/cids/idsRoot/var/iplogs—Stores iplog file data.
- /usr/cids/idsRoot/bin—Contains the binary executables.
- /usr/cids/idsRoot/bin/authentication—Contains the authentication application.
- /usr/cids/idsRoot/bin/cidDump—Contains the script that gathers data for tech support.
- /usr/cids/idsRoot/bin/cidwebserver—Contains the web server application.
- /usr/cids/idsRoot/bin/cidcli—Contains the CLI application.
- /usr/cids/idsRoot/bin/nac—Contains the ARC application.
- /usr/cids/idsRoot/bin/logApp—Contains the logger application.
- /usr/cids/idsRoot/bin/mainApp—Contains the main application.
- /usr/cids/idsRoot/bin/sensorApp—Contains the sensor application.
- /usr/cids/idsRoot/bin/collaborationApp—Contains the collaboration application.
- /usr/cids/idsRoot/bin/switchApp—Contains the switch application.
- /usr/cids/idsRoot/etc—Stores sensor configuration files.
- /usr/cids/idsRoot/htdocs—Contains the IDM files for the web server.
- /usr/cids/idsRoot/lib—Contains the library files for the sensor applications.
- /usr/cids/idsRoot/log—Contains the log files for debugging.
- /usr/cids/idsRoot/tmp—Stores the temporary files created during run time of the sensor.

Summary of Cisco IPS Applications

Table A-2 gives a summary of the applications that make up the IPS.

Table A-2 Summary of Applications

Application	Description
AuthenticationApp	Authorizes and authenticates users based on IP address, password, and digital certificates.
Attack Response Controller	An ARC is run on every sensor. Each ARC subscribes to network access events from its local Event Store. The ARC configuration contains a list of sensors and the network access devices that its local ARC controls. If an ARC is configured to send network access events to a master blocking sensor, it initiates a network access control transaction to the remote ARC that controls the device. These network access action control transactions are also used by IPS managers to issue occasional network access actions.

Table A-2 Summary of Applications (continued)

Application	Description
CLI	Accepts command line input and modifies the local configuration using the IDAPI.
CollaborationApp	Shares information with other devices through a global correlation database to improve the combined efficacy of all the devices.
Control Transaction Server ¹	Accepts control transactions from a remote client, initiates a local control transaction, and returns the response to the remote client.
Control Transaction Source ²	Waits for control transactions directed to remote applications, forwards the control transactions to the remote node, and returns the response to the initiator.
IDM	The Java applet that provides an HTML IPS management interface.
IME	The Java applet that provides an interface for viewing and archiving events.
InterfaceApp	Handles bypass and physical settings and defines paired interfaces. Physical settings are speed, duplex, and administrative state.
Logger	Writes all the log messages of the application to the log file and the error messages of the application to the Event Store.
MainApp	Reads the configuration and starts applications, handles starting and stopping of applications and node reboots, handles software upgrades.
NotificationApp	Sends SNMP traps when triggered by alert, status, and error events. NotificationApp uses the public domain SNMP agent. SNMP GETs provide information about the general health of the sensor.
SDEE Server ³	Accepts requests for events from remote clients.
SensorApp	Captures and analyzes traffic on the monitored network and generates intrusion and network access events. Responds to IP logging control transactions that turn logging on and off and that send and delete IP log files.
SwitchApp	Part of the IPS 4500 series design that enables the InterfaceApp and sensor initialization scripts to communicate with and control the built-in switch. Any application that needs to get or set information on the switch must communicate with the SwitchApp.
Web Server	Waits for remote HTTP client requests and calls the appropriate servlet application.

1. This is a web server servlet.
2. This is a remote control transaction proxy.
3. This is a web server servlet.



Signature Engines

This appendix describes the IPS signature engines, and contains the following sections:

- [Understanding Signature Engines, page B-1](#)
- [Master Engine, page B-4](#)
- [Regular Expression Syntax, page B-9](#)
- [AIC Engine, page B-10](#)
- [Atomic Engine, page B-14](#)
- [Fixed Engine, page B-29](#)
- [Flood Engine, page B-32](#)
- [Meta Engine, page B-33](#)
- [Multi String Engine, page B-35](#)
- [Normalizer Engine, page B-36](#)
- [Service Engines, page B-38](#)
- [State Engine, page B-59](#)
- [String Engines, page B-61](#)
- [String XL Engines, page B-64](#)
- [Sweep Engines, page B-67](#)
- [Traffic Anomaly Engine, page B-70](#)
- [Traffic ICMP Engine, page B-72](#)
- [Trojan Engines, page B-73](#)

Understanding Signature Engines

A signature engine is a component of the Cisco IPS that is designed to support many signatures in a certain category. An engine is composed of a parser and an inspector. Each engine has a set of parameters that have allowable ranges or sets of values.



Note

The Cisco IPS engines support a standardized Regex.

Cisco IPS contains the following signature engines:

- **AIC**—Provides thorough analysis of web traffic. The AIC engine provides granular control over HTTP sessions to prevent abuse of the HTTP protocol. It allows administrative control over applications, such as instant messaging and gotomypc, that try to tunnel over specified ports. You can also use AIC to inspect FTP traffic and control the commands being issued. There are two AIC engines: AIC FTP and AIC HTTP.
- **Atomic**—The Atomic engines are combined into four engines with multi-level selections. You can combine Layer 3 and Layer 4 attributes within one signature, for example IP + TCP. The Atomic engine uses the standardized Regex support. The Atomic engines consist of the following types:
 - **Atomic ARP**—Inspects Layer 2 ARP protocol. The Atomic ARP engine is different because most engines are based on Layer 3 IP protocol.
 - **Atomic IP Advanced**—Inspects IPv6 Layer 3 and ICMPv6 Layer 4 traffic.
 - **Atomic IP**—Inspects IP protocol packets and associated Layer 4 transport protocols. This engine lets you specify values to match for fields in the IP and Layer 4 headers, and lets you use Regex to inspect Layer 4 payloads.



Note All IP packets are inspected by the Atomic IP engine. This engine replaces the 4.x Atomic ICMP, Atomic IP Options, Atomic L3 IP, Atomic TCP, and Atomic UDP engines.

- **Atomic IPv6**—Detects two IOS vulnerabilities that are stimulated by malformed IPv6 traffic.
- **Fixed**—Performs parallel regular expression matches up to a fixed depth, then stops inspection using a single regular expression table. There are three Fixed engines: ICMP, TCP, and UDP.
- **Flood**—Detects ICMP and UDP floods directed at hosts and networks. There are two Flood engines: Flood Host and Flood Net.
- **Meta**—Defines events that occur in a related manner within a sliding time interval. This engine processes events rather than packets.
- **Multi String**—Inspects Layer 4 transport protocols and payloads by matching several strings for one signature. This engine inspects stream-based TCP and single UDP and ICMP packets.
- **Normalizer**—Configures how the IP and TCP Normalizer functions and provides configuration for signature events related to the IP and TCP Normalizer. Allows you to enforce RFC compliance.
- **Service**—Deals with specific protocols. The Service engines are divided in to the following protocol types:
 - **DNS**—Inspects DNS (TCP and UDP) traffic.
 - **FTP**—Inspects FTP traffic.
 - **FTP V2**—Supports IOS IPS. This signature engine provides a protocol decode engine tuned for IOS IPS. If you try to use this engine, you receive an error message.
 - **Generic**—Decodes custom service and payload, and generically analyzes network protocols.
 - **H225**—Inspects VoIP traffic. Helps the network administrator make sure the SETUP message coming in to the VoIP network is valid and within the bounds that the policies describe. Is also helps make sure the addresses and Q.931 string fields such as url-ids, email-ids, and display information adhere to specific lengths and do not contain possible attack patterns.
 - **HTTP**—Inspects HTTP traffic. The WEBPORTS variable defines inspection port for HTTP traffic.

- HTTP V2—Supports IOS IPS. This signature engine provides a protocol decode engine tuned for IOS IPS. If you try to use this engine, you receive an error message.
- IDENT—Inspects IDENT (client and server) traffic.
- MSRPC—Inspects MSRPC traffic.
- MSSQL—Inspects Microsoft SQL traffic.
- NTP—Inspects NTP traffic.
- P2P—Inspects P2P traffic.
- RPC—Inspects RPC traffic.
- SMB Advanced—Processes Microsoft SMB and Microsoft DCE/RPC (MSRPC) over SMB packets.



Note The SMB engine has been replaced by the SMB Advanced engine. Even though the SMB engine is still visible in IDM, IME, and the CLI, its signatures have been obsoleted; that is, the new signatures have the obsoletes parameter set with the IDs of their corresponding old signatures. Use the new SMB Advanced engine to rewrite any custom signature that were in the SMB engine.

- SMPT V1—Supports IOS IPS.
This signature engine provides a protocol decode engine tuned for IOS IPS. If you try to use this engine, you receive an error message.
- SNMP—Inspects SNMP traffic.
- SSH—Inspects SSH traffic.
- TNS—Inspects TNS traffic.
- State—Conducts stateful searches of strings in protocols such as SMTP. The state engine has a hidden configuration file that is used to define the state transitions so new state definitions can be delivered in a signature update.
- String—Searches on Regex strings based on ICMP, TCP, or UDP protocol. There are three String engines: String ICMP, String TCP, and String UDP.
- String XL—Searches on Regex strings based on ICMP, TCP, or UDP protocol. The String XL engines provide optimized operation for the Regex accelerator card. There are three String engines: String ICMP XL, String TCP XL, and String UDP XL.



Note The IPS 4345, IPS 4360, IPS 4510, IPS 4520, and IPS 4520-XL support the String XL engines and the Regex accelerator card.



Note The Regex accelerator card is used for both the standard String engines and the String XL engines. Most standard String engine signatures can be compiled and analyzed by the Regex accelerator card without modification. However, there are special circumstances in which the standard String engine signatures cannot be compiled for the Regex accelerator card. In these situations a new signature is written in a String XL engine using the specific parameters in the String XL engine that do compile on the Regex accelerator card. The new signature in the String XL engine obsoletes the original signature in the standard String engine.

- Sweep—Analyzes sweeps from a single host (ICMP and TCP), from destination ports (TCP and UDP), and multiple ports with RPC requests between two nodes. There are two Sweep engines: Sweep and Sweep Other TCP.
- Traffic Anomaly—Inspects TCP, UDP, and other traffic for worms.
- Traffic ICMP—Analyzes nonstandard protocols, such as TFN2K, LOKI, and DDOS. There are only two signatures with configurable parameters.
- Trojan—Analyzes traffic from nonstandard protocols, such as BO2K andTFN2K. There are three Trojan engines: Bo2k, Tfn2k, and UDP. There are no user-configurable parameters in these engines.

For More Information

For a list of the signature regular expression syntax, see [Regular Expression Syntax, page B-9](#).

Master Engine

The Master engine provides structures and methods to the other engines and handles input from configuration and alert output. This section describes the Master engine, and contains the following topics:

- [General Parameters, page B-4](#)
- [Alert Frequency, page B-7](#)
- [Event Actions, page B-8](#)

General Parameters

The following parameters are part of the Master engine and apply to all signatures (if it makes sense for that signature engine).

[Table B-1](#) lists the general master engine parameters.

Table B-1 Master Engine Parameters

Parameter	Description	Value
signature-id	Specifies the ID of this signature.	<i>number</i>
sub-signature-id	Specifies the sub ID of this signature	<i>number</i>
alert-severity	Specifies the severity of the alert: <ul style="list-style-type: none"> • Dangerous alert • Medium-level alert • Low-level alert • Informational alert 	high medium low informational (default)
sig-fidelity-rating	Specifies the rating of the fidelity of this signature.	0 to 100 (default = 100)
promisc-delta	Specifies the delta value used to determine the seriousness of the alert.	0 to 30 (default = 5)
sig-name	Specifies the name of the signature.	<i>sig-name</i>

Table B-1 Master Engine Parameters (continued)

Parameter	Description	Value
alert-notes	Provides additional information about this signature that will be included in the alert message.	<i>alert-notes</i>
user-comments	Provides comments about this signature.	<i>comments</i>
alert-traits	Specifies traits you want to document about this signature.	0 to 65335
release	Provides the release in which the signature was most recently updated.	<i>release</i>
signature-creation-date	Specifies the date the signature was created.	—
signature-type	Specifies the signature category.	anomaly component exploit other vulnerability
engine	Specifies the engine to which the signature belongs. Note The engine-specific parameters appear under the engine category.	—
event-count	Specifies the number of times an event must occur before an alert is generated.	1 to 65535 (default = 1)
event-count-key	Specifies the storage type on which to count events for this signature: <ul style="list-style-type: none"> Attacker address Attacker and victim addresses Attacker address and victim port Victim address Attacker and victim addresses and ports 	Axxx AxBx Axxb xxBx AaBb
specify-alert-interval {yes no}	Enables the alert interval: <ul style="list-style-type: none"> alert-interval—Specifies the time in seconds before the event count is reset. 	2 to 1000
status	Specifies whether the signature is enabled or disabled, active or retired.	enabled retired {yes no}
obsoletes	Indicates that a newer signature has disabled an older signature.	—

Table B-1 Master Engine Parameters (continued)

Parameter	Description	Value
vulnerable-os-list	When combined with passive OS fingerprinting, it allows the IPS to determine if it is likely a given attack is relevant to the target system.	aix bsd general-os hp-ux ios irix linus mac-os netware other solaris unix windows windows-ut windows-nt-2k-xp
mars-category {yes no}	Maps signatures to a MARS attack category. ¹	—

1. This is a static information category that you can set in the configuration and view in the alerts. Refer to the MARS documentation for more information.

Promiscuous Delta

The promiscuous delta lowers the risk rating of certain alerts in promiscuous mode. Because the sensor does not know the attributes of the target system and in promiscuous mode cannot deny packets, it is useful to lower the prioritization of promiscuous alerts (based on the lower risk rating) so the administrator can focus on investigating higher risk rating alerts. In inline mode, the sensor can deny the offending packets so that they never reach the target host, so it does not matter if the target was vulnerable. Because the attack was not allowed on the network, the IPS does not subtract from the risk rating value. Signatures that are not service, OS, or application-specific have 0 for the promiscuous delta. If the signature is specific to an OS, service, or application, it has a promiscuous delta of 5, 10, or 15 calculated from 5 points for each category.



Caution

We recommend that you do NOT change the promisc-delta setting for a signature.

Obsoletes

The Cisco signature team uses the obsoletes field to indicate obsoleted, older signatures that have been replaced by newer, better signatures, and to indicate disabled signatures in an engine when a better instance of that engine is available. For example, some String XL hardware-accelerated signatures now replace equivalent signatures that were defined in the String engine.

Vulnerable OS List

When you combine the vulnerable OS setting of a signature with passive OS fingerprinting, the IPS can determine if it is likely that a given attack is relevant to the target system. If the attack is found to be relevant, the risk rating value of the resulting alert receives a boost. If the relevancy is unknown, usually because there is no entry in the passive OS fingerprinting list, then no change is made to the risk rating.

If there is a passive OS fingerprinting entry and it does not match the vulnerable OS setting of a signature, the risk rating value is decreased. The default value by which to increase or decrease the risk rating is +/- 10 points.

For More Information

- For more information about promiscuous mode, see [Understanding Promiscuous Mode, page 5-14](#).
- For more information about passive OS fingerprinting, see [Configuring OS Identifications, page 8-26](#).

Alert Frequency

The purpose of the alert frequency parameter is to reduce the volume of the alerts written to the Event Store to counter IDS DoS tools, such as stick. There are four modes: fire-all, fire-once, summarize, and global-summarize. The summary mode is changed dynamically to adapt to the current alert volume. For example, you can configure the signature to fire-all, but after a certain threshold is reached, it starts summarizing.

[Table B-2](#) lists the alert frequency parameters.

Table B-2 Master Engine Alert Frequency Parameters

Parameter	Description	Value
summary-mode	Specifies the mode used for summarization: <ul style="list-style-type: none"> • fire-all—Fires an alert on all events. • fire-once—Fires an alert only once. • global-summarize—Summarizes an alert so that it only fires once regardless of how many attackers or victims. • summarize—Summarizes alerts. 	fire-all fire-once global-summarize summarize
specify-summary-threshold {yes no}	Enables summary threshold mode: <ul style="list-style-type: none"> • summary-threshold—Specifies the threshold number of alerts to send a signature into summary mode. • summary-interval—Specifies the time in seconds used in each summary alert. 	0 to 65535 1 to 1000

Table B-2 Master Engine Alert Frequency Parameters (continued)

Parameter	Description	Value
specify-global-summary-threshold {yes no}	Enables global summary threshold mode: <ul style="list-style-type: none"> global-summary-threshold—Specifies the threshold number of events to take alerts into global summary. 	1 to 65535
summary-key	Specifies the storage type on which to summarize this signature: <ul style="list-style-type: none"> Attacker address Attacker and victim addresses Attacker address and victim port Victim address Attacker and victim addresses and ports 	Axxx AxBx Axxb xxBx AaBb

Event Actions

The Cisco IPS supports the following event actions. Most of the event actions belong to each signature engine unless they are not appropriate for that particular engine.

Alert and Log Actions

- produce-alert—Writes an evIdsAlert to Event Store.
- produce-verbose-alert—Includes an encoded dump (possibly truncated) of the offending packet in the evIdsAlert.
- log-attacker-packets—Starts IP logging of packets containing the attacker address and sends an alert.
- log-victim-packets—Starts IP logging of packets containing the victim address and sends an alert.
- log-pair-packets (inline mode only)—Starts IP logging of packets containing the attacker/victim address pair.
- request-snmp-trap—Sends request to the NotificationApp to perform SNMP notification.

Deny Actions

- deny-packet-inline (inline mode only)—Does not transmit this packet.



Note You cannot delete the event action override for deny-packet-inline because it is protected. If you do not want to use that override, set the override-item-status to disabled for that entry.

- deny-connection-inline (inline mode only)—Does not transmit this packet and future packets on the TCP Flow.
- deny-attacker-victim-pair-inline (inline mode only)—Does not transmit this packet and future packets on the attacker/victim address pair for a specified period of time.
- deny-attacker-service-pair-inline (inline mode only)—Does not transmit this packet and future packets on the attacker address victim port pair for a specified period of time.

- deny-attacker-inline (inline mode only)—Does not transmit this packet and future packets from the attacker address for a specified period of time.



Note This is the most severe of the deny actions. It denies the current and future packets from a single attacker address. Each deny address times out for *X* seconds from the first event that caused the deny to start, where *X* is the amount of seconds that you configured. You can clear all denied attacker entries with the **clear denied-attackers** command, which permits the addresses back on the network.

- modify-packet-inline (inline mode only)—Modifies packet data to remove ambiguity about what the end point might do with the packet.



Note The event action modify-packet-inline is part of the Normalizer engine. It scrubs the packet and corrects irregular issues such as bad checksum, out of range values, and other RFC violations.

Other Actions



Note IPv6 does not support the following event actions: request-block-host, request-block-connection, or request-rate-limit.

- request-block-connection—Requests the ARC to block this connection.
- request-block-host—Requests the ARC to block this attacker host.
- request-rate-limit—Requests the ARC to perform rate limiting.
- reset-tcp-connection—Sends TCP resets to hijack and terminate the TCP flow.

Regular Expression Syntax

Regular expressions (Regex) are a powerful and flexible notational language that allow you to describe text. In the context of pattern matching, regular expressions allow a succinct description of any arbitrary pattern.

Table B-3 lists the IPS signature Regex syntax.

Table B-3 Signature Regular Expression Syntax

Metacharacter	Name	Description
?	Question mark	Repeat 0 or 1 times.
*	Star, asterisk	Repeat 0 or more times.
+	Plus	Repeat 1 or more times.
{x}	Quantifier	Repeat exactly <i>X</i> times.
{x,}	Minimum quantifier	Repeat at least <i>X</i> times.
.	Dot	Any one character except new line (0x0A).
[abc]	Character class	Any character listed.

Table B-3 Signature Regular Expression Syntax (continued)

Metacharacter	Name	Description
[^abc]	Negated character class	Any character not listed.
[a-z]	Character range class	Any character listed inclusively in the range.
()	Parenthesis	Used to limit the scope of other metacharacters.
	Alternation, or	Matches either expression it separates.
^	caret	The beginning of the line.
\char	Escaped character	When <i>char</i> is a metacharacter or not, matches the literal <i>char</i> .
<i>char</i>	Character	When <i>char</i> is not a metacharacter, matches the literal <i>char</i> .
\r	Carriage return	Matches the carriage return character (0x0D).
\n	New line	Matches the new line character (0x0A).
\t	Tab	Matches the tab character (0x09).
\f	Form feed	Matches the form feed character (0x0C).
\xNN	Escaped hexadecimal character	Matches character with the hexadecimal code 0xNN (0<=N<=F).
\NNN	Escaped octal character	Matches the character with the octal code NNN (0<=N<=8).

All repetition operators will match the shortest possible string as opposed to other operators that consume as much of the string as possible thus giving the longest string match.

Table B-4 lists examples of Regex patterns.

Table B-4 Regex Patterns

To Match	Regular Expression
Hacker	Hacker
Hacker or hacker	[Hh]acker
Variations of bananas, banananas, bananananas	ba(na)+s
foo and bar on the same line with anything except a new line between them	foo.*bar
Either foo or bar	foolbar
Either moon or soon	(m s)oon

AIC Engine

The Application Inspection and Control (AIC) engine inspects HTTP web traffic and enforces FTP commands. This section describes the AIC engine and its parameters, and contains the following topics:

- [Understanding the AIC Engine, page B-11](#)
- [AIC Engine and Sensor Performance, page B-11](#)
- [AIC Engine Parameters, page B-11](#)

Understanding the AIC Engine

AIC provides thorough analysis of web traffic. It provides granular control over HTTP sessions to prevent abuse of the HTTP protocol. It allows administrative control over applications, such as instant messaging and gotomypc, that try to tunnel over specified ports. Inspection and policy checks for P2P and instant messaging are possible if these applications are running over HTTP. AIC also provides a way to inspect FTP traffic and control the commands being issued. You can enable or disable the predefined signatures or you can create policies through custom signatures.

**Note**

The AIC engines run when HTTP traffic is received on AIC web ports. If traffic is web traffic, but not received on the AIC web ports, the Service HTTP engine is executed. AIC inspection can be on any port if it is configured as an AIC web port and the traffic to be inspected is HTTP traffic.

AIC Engine and Sensor Performance

Application policy enforcement is a unique sensor feature. Rather than being based on traditional IPS technologies that inspect for exploits, vulnerabilities, and anomalies, AIC policy enforcement is designed to enforce HTTP and FTP service policies. The inspection work required for this policy enforcement is extreme compared with traditional IPS inspection work. A large performance penalty is associated with using this feature. When AIC is enabled, the overall bandwidth capacity of the sensor is reduced.

AIC policy enforcement is disabled in the IPS default configuration. If you want to activate AIC policy enforcement, we highly recommend that you carefully choose the exact policies of interest and disable those you do not need. Also, if your sensor is near its maximum inspection load capacity, we recommend that you not use this feature since it can oversubscribe the sensor. We recommend that you use the adaptive security appliance firewall to handle this type of policy enforcement.

AIC Engine Parameters

The AIC engines define signatures for deep inspection of web traffic. They also define signatures that authorize and enforce FTP commands. There are two AIC engines: AIC HTTP and AIC FTP. The AIC engines have the following features:

- Web traffic:
 - RFC compliance enforcement
 - HTTP request method authorization and enforcement
 - Response message validation
 - MIME type enforcement
 - Transfer encoding type validation
 - Content control based on message content and type of data being transferred
 - URI length enforcement
 - Message size enforcement according to policy configured and the header
 - Tunneling, P2P and instant messaging enforcement.

This enforcement is done using regular expressions. There are predefined signature but you can expand the list.

- FTP traffic:
 - FTP command authorization and enforcement

Table B-5 lists the parameters that are specific to the AIC HTTP engine.

Table B-5 AIC HTTP Engine Parameters

Parameter	Description	
signature-type	Specifies the type of AIC signature.	<ul style="list-style-type: none"> • content-types • define-web-traffic-policy • max-outstanding-requests-overrun • max-outstanding-requests-overrun • msg-body-pattern • request-methods • transfer-encodings •
content-types	Specifies the AIC signature that deals with MIME types: <ul style="list-style-type: none"> • define-content-type—Associates actions such as denying a specific MIME type (image/gif), defining a message-size violation, and determining that the MIME-type mentioned in the header and body do not match. • define-recognized-content-types—Lists the content types recognized by the sensor. 	—
define-web-traffic-policy	Specifies the action to take when noncompliant HTTP traffic is seen. The alarm-on-non-http-traffic {true false} command enables the signature. This signature is disabled by default.	—
max-outstanding-requests-overrun	Specifies the maximum allowed HTTP requests per connection.	1 - 16
msg-body-pattern	Uses Regex to define signatures that look for specific patterns in the message body: <ul style="list-style-type: none"> • regex-list— • regex-list-in-order— 	—

Table B-5 AIC HTTP Engine Parameters (continued)

Parameter	Description	
request-methods	Specifies an AIC signature that allows actions to be associated with HTTP request methods: <ul style="list-style-type: none"> define-request-method—Specifies get, put, and so forth. recognized-request-methods—Lists methods recognized by the sensor. 	—
transfer-encodings	Specifies an AIC signature that deals with transfer encodings: <ul style="list-style-type: none"> define-transfer-encoding—Associates an action with each method, such as compress, chunked, and so forth. recognized-transfer-encodings—Lists methods recognized by the sensor. chunked-transfer-encoding—Error specifies actions to be taken when a chunked encoding error is seen. 	—

Table B-6 lists the parameters that are specific to the AIC FTP engine.

Table B-6 AIC FTP Engine Parameters

Parameter	Description	Value
signature-type	Specifies the type of AIC signature.	<ul style="list-style-type: none"> ftp-commands unrecognized-ftp-command
ftp-commands	Associates an action with an FTP command: <ul style="list-style-type: none"> ftp-command—Lets you choose the FTP command you want to inspect. 	<ul style="list-style-type: none"> help, noop, stat, syst, user, abort, acct, allo, appe, cdup, cwd, dele, list, mkd, mode, nlst, pass, pasv, port, pass, quit, rein, rest, retr, rmd, rnfr, rnto, site, smnt, stor, stou, stru, type
unrecognized-ftp-command	Inspects unrecognized FTP commands.	—

For More Information

- For the procedures for configuring AIC engine signatures, see [Configuring AIC Signatures, page 7-22](#).
- For an example of a custom AIC signature, see [Creating an AIC Signature, page 7-32](#).
- For more information on the parameters common to all signature engines, see [Master Engine, page B-4](#).

Atomic Engine

The Atomic engine contains signatures for simple, single packet conditions that cause alerts to be fired. This section describes the Atomic engine, and contains the following topics:

- [Atomic ARP Engine, page B-14](#)
- [Atomic IP Advanced Engine, page B-15](#)
- [Atomic IP Engine, page B-25](#)
- [Atomic IPv6 Engine, page B-28](#)

Atomic ARP Engine

The Atomic ARP engine defines basic Layer 2 ARP signatures and provides more advanced detection of the ARP spoof tools dsniff and ettercap.

[Table B-7](#) lists the parameters that are specific to the Atomic ARP engine.

Table B-7 Atomic ARP Engine Parameters

Parameter	Description	Value
specify-arp-operation {yes no}	(Optional) Enables ARP operation: <ul style="list-style-type: none"> • arp-operation—Specifies the type of ARP operation to inspect. 	0 to 65535
specify-mac-flip {yes no}	(Optional) Enables MAC address flip times: <ul style="list-style-type: none"> • mac-flip—Specifies how many times to flip the MAC address in the alert. 	0 to 65535
specify-request-inbalance {yes no}	(Optional) Enables request inbalance: <ul style="list-style-type: none"> • request-inbalance—Specifies the value for firing an alert when there are this many more requests than replies on the IP address. 	0 to 65535

Table B-7 Atomic ARP Engine Parameters (continued)

Parameter	Description	Value
specify-type-of-arp-sig {yes no}	(Optional) Enables the ARP signature type: <ul style="list-style-type: none"> • type-of-arp-sig—Specifies the type of ARP signatures you want to fire on: <ul style="list-style-type: none"> – Destination Broadcast—Fires an alert for this signature when it sees an ARP destination address of 255.255.255.255. – Same Source and Destination—Fires an alert for this signature when it sees an ARP destination address with the same source and destination MAC address – Source Broadcast (default)—Fires an alert for this signature when it sees an ARP source address of 255.255.255.255. – Source Multicast—Fires an alert for this signature when it sees an ARP source MAC address of 01:00:5e:(00-7f). 	dst-broadcast same-src-dst src-broadcast src-multicast
storage-key	Specifies the type of address key used to store persistent data: <ul style="list-style-type: none"> • Attacker address • Attacker and victim addresses • Victim address • Global 	Axxx AxBx xxBx xxxx

For More Information

For more information on the parameters common to all signature engines, see [Master Engine, page B-4](#).

Atomic IP Advanced Engine

The Atomic IP Advanced engine parses and interprets the IPv6 header and its extensions, the IPv4 header and its options, ICMP, ICMPv6, TCP, and UDP, and seeks out anomalies that indicate unusual activity.

Atomic IP Advanced engine signatures do the following:

- Inspect for anomalies in IP addresses, for example, spoofed addresses.
- Inspect for bad information in the length fields of the packet.
- Fire informational alerts about the packet.
- Fire higher severity alerts for the limited set of known vulnerabilities.
- Duplicate any IPv6-specific signatures in Engine Atomic IP that can also apply to IPv6.
- Provide default signatures for identifying tunneled traffic based on IP address, port, protocol, and limited information from the packet data.

Only the outermost IP tunnel is identified. When an IPv6 tunnel or IPv6 traffic inside of an IPv4 tunnel is detected, a signature fires an alert. All of the other IPv6 traffic in embedded tunnels is not inspected. The following tunneling methods are supported, but not individually detected. For example, ISATAP, 6to4, and manual IPv6 RFC 4213 tunnels all appear as IPv6 in IPv4, which is detected by signature 1007:

- ISATAP
- 6to4 (RFC 3056)
- Manually configured tunnels (RFC 4213)
- IPv6 over GRE
- Teredo (IPv6) inside UDP
- MPLS (unencrypted)
- IPv6 over IPv6

IPv6 supports the following:

- Denying by source IP address, destination IP address, or IP address pair
- Alerts
- Resetting the TCP connection
- Logging

Atomic IP Advanced Engine Restrictions

The Atomic IP Advanced engine contains the following restrictions:

- Cannot detect the Layer 4 field of the packets if the packets are fragmented so that the Layer 4 identifier does not appear in the first packet.
- Cannot detect Layer 4 attacks in flows with packets that are fragmented by IPv6 because there is no fragment reassembly.
- Cannot detect attacks with tunneled flows.
- Limited checks are provided for the fragmentation header.
- There is no support for IPv6 on the management (command and control) interface.
- If there are illegal duplicate headers, a signature fires, but the individual headers cannot be separately inspected.
- Anomaly detection does not support IPv6 traffic; only IPv4 traffic is directed to the anomaly detection processor.
- Rate limiting and blocking are not supported for IPv6 traffic. If a signature is configured with a block or rate limit event action and is triggered by IPv6 traffic, an alert is generated but the action is not carried out.



Note

The second number in the ranges must be greater than or equal to the first number.

Table B-8 lists the parameters that are specific to the Atomic IP Advanced engine.

Table B-8 Atomic IP Advanced Engine Parameters

Parameter	Description	Value
Global		
fragment-status	Specifies whether or not fragments are wanted.	any no-fragments want-fragments
specify-encapsulation {yes no}	(Optional) Enables any encapsulation before the start of Layer 3 for the packet: ¹ <ul style="list-style-type: none"> encapsulation—Specifies the type of encapsulation to inspect. 	none mpls gre ipv4-in-ipv6 ipipl any
specify-ip-version {yes no}	(Optional) Enables the IP protocol version: <ul style="list-style-type: none"> version—Specifies IPv4 or IPv6. 	ipv4 ipv6
swap-attacker-victim	Swaps the attacker and victim addresses and ports (source and destination) in the alert message and in any actions taken.	true false (default)
Regex		
specify-regex-inspection	(Optional) Enables Regex inspection.	yes no
regex-scope	Specifies the start and end points for the regular expression search.	ipv6-doh-only ipv6-doh-plus ipv6-hoh-only ipv6-hoh-plus ipv6-rh-only ipv6-rh-plus layer3-only layer3-plus layer4
regex-string	Specifies the regular expression to search for in a single TCP packet.	<i>string</i>
specify-exact-match-offset {yes no}	Enables exact match offset: <ul style="list-style-type: none"> exact-match-offset—Specifies the exact stream offset the regex-string must report for a match to be valid. 	0 to 65535
specify-min-match-length {yes no}	Enables minimum match length: <ul style="list-style-type: none"> min-match-length—Specifies the minimum number of bytes the regex-string must match. 	0 to 65535
specify-min-match-offset {yes no}	Enables minimum match offset: <ul style="list-style-type: none"> min-match-offset—Specifies the minimum stream offset the regex-string must report for a match to be valid. 	0 to 65535

Table B-8 Atomic IP Advanced Engine Parameters (continued)

Parameter	Description	Value
specify-max-match-offset {yes no}	Enables maximum match offset: <ul style="list-style-type: none"> max-match-offset—Specifies the maximum stream offset the regex-string must report for a match to be valid. 	0 to 65535
IPv6		
specify-authentication-header {yes no}	(Optional) Enables inspection of the authentication header: <ul style="list-style-type: none"> ah-present—Inspects the authentication header: <ul style="list-style-type: none"> ah-length—Specifies the length of the authentication header to inspect. ah-next-header—Specifies the value of the authentication header to inspect. 	<ul style="list-style-type: none"> have-ah no-ah 0 to 1028 0 to 255
specify-dest-options-header {yes no}	(Optional) Enables inspection of the destination options header: <ul style="list-style-type: none"> doh-present—Inspects the destination options header: <ul style="list-style-type: none"> doh-count—Specifies the number of destination options headers to inspect. doh-length—Specifies the length of destination options headers to inspect. doh-next-header—Specifies the number of next destination options headers to inspect. doh-option-type—Specifies the type of destination options headers to inspect. doh-option-length—Specifies the length of destination options headers to inspect. 	<ul style="list-style-type: none"> have-doh no-doh 0 to 2 8 to 2048 0 to 255 0 to 255 0 to 255
specify-esp-header {yes no}	(Optional) Enables inspection of the ESP header: <ul style="list-style-type: none"> esp-present —Inspects the ESP header. 	have-esp no-esp
specify-first-next-header {yes no}	(Optional) Enables inspection of the first next header: <ul style="list-style-type: none"> first-next-header—Specifies the value of the first next header to inspect. 	0 to 255

Table B-8 Atomic IP Advanced Engine Parameters (continued)

Parameter	Description	Value
specify-flow-label {yes no}	(Optional) Enables inspection of the flow label: <ul style="list-style-type: none"> flow-label—Specifies the value of the flow label to inspect. 	0 to 1048575
specify-headers-out-of-order {yes no}	(Optional) Enables inspection of out-of-order headers: <ul style="list-style-type: none"> headers-out-of-order—Inspects headers that are out of order. 	true false
specify-headers-repeated {yes no}	(Optional) Enables inspection of repeated headers: <ul style="list-style-type: none"> headers-repeated—Inspects repeated headers. 	{yes no}
specify-hop-limit {yes no}	(Optional) Enables hop limit: <ul style="list-style-type: none"> hop-limit—Specifies the value of the hop limit to inspect. 	0 to 255
specify-hop-options-header {yes no}	(Optional) Enables inspection of the hop-by-hop options header: <ul style="list-style-type: none"> hoh-present—Inspects the hop-by-hop options header. 	have-hoh no-hoh
specify-ipv6-addr-options {yes no}	(Optional) Enables the IPv6 address options: <ul style="list-style-type: none"> ipv6-addr-options—Specifies the IPv6 address options: <ul style="list-style-type: none"> address-with-localhost—IP address with ::1. documentation-address—IP address with 2001:db8::/32 prefix. ipv6-addr—IP address. link-local-address—Inspects for an IPv6 link local address. multicast-dst—Inspects for a destination multicast address. multicast-src—Inspects for a source multicast address. not-link-local-address—Inspects for an address that is not link-local. not-valid-address—Inspects for an address that is not reserved for link-local, global, or multicast. src-ip-eq-dst-ip—Source and destination addresses are the same. 	true false

Table B-8 Atomic IP Advanced Engine Parameters (continued)

Parameter	Description	Value
specify-ipv6-data-length {yes no}	(Optional) Enables inspection of IPv6 data length: <ul style="list-style-type: none"> ipv6-data-length—Specifies the IPv6 data length to inspect. 	0 to 65535
specify-ipv6-header-length {yes no}	(Optional) Enables inspection of IPv6 header length: <ul style="list-style-type: none"> ipv6-header-length—Specifies the length of the IPv6 header to inspect. 	0 to 65535
specify-ipv6-total-length {yes no}	(Optional) Enables inspection of IPv6 total length: <ul style="list-style-type: none"> ipv6-total-length—Specifies the IPv6 total length to inspect. 	0 to 65535
specify-ipv6-payload-length {yes no}	(Optional) Enables inspection of IPv6 payload length: <ul style="list-style-type: none"> ipv6-payload-length—Specifies the IPv6 payload length to inspect. 	0 to 65535
specify-routing-header {yes no}	(Optional) Enables inspection of the routing header: <ul style="list-style-type: none"> rh-present—Inspects the routing header. 	have-rh no-rh
specify-traffic-class {yes no}	(Optional) Enables inspection of the traffic class: <ul style="list-style-type: none"> traffic-class—Specifies the value of the traffic class to inspect. 	0 to 255
IPV4		
specify-ip-addr-options {yes no}	(Optional) Enables IP address options: <ul style="list-style-type: none"> ip-addr-options—Specifies the IP address options. 	address-with-localhost ip-addr ² rfc-1918-address src-ip-eq-dst-ip
specify-ip-header-length {yes no}	(Optional) Enables inspection of the IP header length: <ul style="list-style-type: none"> ip-header-length—Specifies the length of the IP header to inspect. 	0 to 16
specify-ip-id {yes no}	(Optional) Enables inspection of the IP identifier: <ul style="list-style-type: none"> ip-id—Specifies the IP ID to inspect. 	0 to 255

Table B-8 Atomic IP Advanced Engine Parameters (continued)

Parameter	Description	Value
specify-ip-option-inspection {yes no}	(Optional) Enables inspection of the IP options: <ul style="list-style-type: none"> ip-option-inspection—Specifies the value of the IP option: <ul style="list-style-type: none"> ip-option—IP OPTION code to match. ip-option-abnormal—The list of options is malformed. 	0 to 65535
specify-ip-payload-length {yes no}	(Optional) Enables inspection of the IP payload length: <ul style="list-style-type: none"> ip-payload-length—Specifies the length of the IP payload to inspect. 	0 to 65535
specify-ip-tos {yes no}	(Optional) Enables inspection of the IP type of service: <ul style="list-style-type: none"> ip-tos—Specifies the IP type of service to inspect. 	0 to 255
specify-ip-total-length {yes no}	(Optional) Enables inspection of the IP total length: <ul style="list-style-type: none"> ip-total-length—Specifies the total length of the IP packet to inspect. 	0 to 65535
specify-ip-ttl {yes no}	(Optional) Enables inspection of the IP time-to-live: <ul style="list-style-type: none"> ip-ttl—Specifies the value of the IP TTL to inspect. 	0 to 255
specify-ip-version {yes no}	(Optional) Enables inspection of the IP version: <ul style="list-style-type: none"> ip-version—Specifies which IP version to inspect. 	0 to 16
L4 Protocol		
specify-l4-protocol {yes no}	(Optional) Enables inspection of Layer 4 protocol: <ul style="list-style-type: none"> l4-protocol—Specifies which Layer 4 protocol to inspect. 	icmp icmpv6 tcp udp other
L4 Protocol Other		
specify-other-ip-protocol-id	(Optional) Enables inspection of other Layer 4 protocols: <ul style="list-style-type: none"> other-ip-protocol-id—Specifies which single IP protocol ID or single range of IP protocol IDs for which to send alerts. 	0 to 255

Table B-8 Atomic IP Advanced Engine Parameters (continued)

Parameter	Description	Value
L4 Protocol ICMP		
specify-icmp-code {yes no}	(Optional) Enables inspection of Layer 4 ICMP code: <ul style="list-style-type: none"> icmp-code—Specifies the value of the ICMP header CODE. 	0 to 65535
specify-icmp-id {yes no}	(Optional) Enables inspection of Layer 4 ICMP ID: <ul style="list-style-type: none"> icmp-id—Specifies the value of the ICMP header IDENTIFIER. 	0 to 65535
specify-icmp-seq {yes no}	(Optional) Enables inspection of Layer 4 ICMP sequence: <ul style="list-style-type: none"> icmp-seq—Specifies the ICMP sequence for which to look. 	0 to 65535
specify-icmp-type {yes no}	(Optional) Enables inspection of the Layer 4 ICMP header type: <ul style="list-style-type: none"> icmp-type—Specifies the value of the ICMP header TYPE. 	0 to 65535
L4 Protocol ICMPv6		
specify-icmpv6-code {yes no}	(Optional) Enables inspection of the Layer 4 ICMPv6 code: <ul style="list-style-type: none"> icmpv6-code—Specifies the value of the ICMPv6 header CODE. 	0 to 255
specify-icmpv6-id {yes no}	(Optional) Enables inspection of the Layer 4 ICMPv6 identifier: <ul style="list-style-type: none"> icmpv6-id—Specifies the value of the ICMPv6 header IDENTIFIER. 	0 to 65535
specify-icmpv6-length {yes no}	(Optional) Enables inspection of the Layer 4 ICMPv6 length: <ul style="list-style-type: none"> icmpv6-length—Specifies the value of the ICMPv6 header LENGTH. 	0 to 65535
specify-icmpv6-mtu-field {yes no}	(Optional) Enables inspection of the Layer 4 ICMPv6 MTU field: <ul style="list-style-type: none"> icmpv6-mtu-field—Specifies the value of the ICMPv6 header MTU field. 	4,294,967,295
specify-icmpv6-option-type {yes no}	(Optional) Enables inspection of the Layer 4 ICMPv6 type: <ul style="list-style-type: none"> icmpv6-option-type—Specifies the ICMPv6 option type to inspect. 	0 to 255

Table B-8 Atomic IP Advanced Engine Parameters (continued)

Parameter	Description	Value
specify-icmpv6-option-length {yes no}	(Optional) Enables inspection of the Layer 4 ICMPv6 option length: <ul style="list-style-type: none"> icmpv6-option-length—Specifies the ICMPv6 option length to inspect. 	0 to 255
specify-icmpv6-seq {yes no}	(Optional) Enables inspection of the Layer 4 ICMPv6 sequence: <ul style="list-style-type: none"> icmpv6-seq—Specifies the value of the ICMPv6 header SEQUENCE. 	0 to 65535
specify-icmpv6-type {yes no}	(Optional) Enables inspection of the Layer 4 ICMPv6 type: <ul style="list-style-type: none"> icmpv6-type—Specifies the value of the ICMPv6 header TYPE. 	0 to 255
L4 Protocol TCP and UDP		
specify-dst-port {yes no}	(Optional) Enables the destination port for use: <ul style="list-style-type: none"> dst-port—Specifies the destination port of interest for this signature. 	0 to 65535
specify-src-port {yes no}	(Optional) Enables source port for use: <ul style="list-style-type: none"> src-port—Specifies the source port of interest for this signature. 	0 to 65535
specify-tcp-mask {yes no}	(Optional) Enables the TCP mask for use: <ul style="list-style-type: none"> tcp-mask—Specifies the mask used in TCP flags comparison: <ul style="list-style-type: none"> URG bit ACK bit PSH bit RST bit SYN bit FIN bit 	urg ack psh rst syn fin
specify-tcp-flags {yes no}	(Optional) Enables TCP flags for use: <ul style="list-style-type: none"> tcp-flags—Specifies the TCP flags to match when masked by mask: <ul style="list-style-type: none"> URG bit ACK bit PSH bit RST bit SYN bit FIN bit 	urg ack psh rst syn fin

Table B-8 Atomic IP Advanced Engine Parameters (continued)

Parameter	Description	Value
specify-tcp-reserved {yes no}	(Optional) Enables TCP reserved for use: <ul style="list-style-type: none"> tcp-reserved—Specifies the value of TCP reserved. 	0 to 63
specify-tcp-header-length {yes no}	(Optional) Enables inspection of the Layer 4 TCP header length: <ul style="list-style-type: none"> tcp-header-length—Specifies the length of the TCP header used in inspection. 	0 to 60
specify-tcp-payload-length {yes no}	(Optional) Enables inspection of the Layer 4 TCP payload length: <ul style="list-style-type: none"> tcp-payload-length—Specifies the length of the TCP payload. 	0 to 65535
specify-tcp-urg-pointer {yes no}	(Optional) Enables inspection of the Layer 4 TCP URG pointer: <ul style="list-style-type: none"> tcp-urg-pointer—Specifies the value of the TCP URG flag inspection. 	0 to 65535
specify-tcp-window-size {yes no}	(Optional) Enables inspection of the Layer 4 TCP window size: <ul style="list-style-type: none"> tcp-window-size—Specifies the window size of the TCP packet. 	0 to 65535
specify-udp-valid-length {yes no}	(Optional) Enables inspection of the Layer 4 UDP valid length: <ul style="list-style-type: none"> udp-valid-length—Specifies the UDP packet lengths that are considered valid and should not be inspected. 	0 to 65535
specify-udp-length-mismatch {yes no}	(Optional) Enables inspection of the Layer 4 UDP length mismatch: <ul style="list-style-type: none"> udp-length-mismatch—Fires an alert when IP Data length is less than the UDP Header length. 	0 to 65535

1. When a packet is GRE, IPIP, IPv4inIPv6, or MPL the sensor skips the Layer 3 encapsulation header and the encapsulation header, and all inspection is done starting from the second Layer 3. The encapsulation enumerator allows the engine to look backward to see if there is an encapsulation header before the Layer 3 in question.
2. Use the following syntax: x.x.x.x-z.z.z.z, for example, 10.10.10.1-10.10.10.254.

For More Information

- For an example custom IPv6 signature, see [Example IPv6 Engine Signature, page 7-57](#).
- For a list of the signature regular expression syntax, see [Regular Expression Syntax, page B-9](#).
- For more information on the parameters common to all signature engines, see [Master Engine, page B-4](#).

Atomic IP Engine

The Atomic IP engine defines signatures that inspect IP protocol headers and associated Layer 4 transport protocols (TCP, UDP, and ICMP) and payloads. The Atomic engines do not store persistent data across packets. Instead they can fire an alert from the analysis of a single packet.

Table B-9 lists the parameters that are specific to the Atomic IP engine.

Table B-9 Atomic IP Engine Parameters

Parameter	Description	Value
specify-ip-addr-options {yes no}	(Optional) Enables IP address options: <ul style="list-style-type: none"> ip-addr-options—Specifies the IP address options. 	address-with-localhost ip-addr ¹ rfc-1918-address src-ip-eq-dst-ip
specify-ip-header-length {yes no}	(Optional) Enables inspection of the IP header length: <ul style="list-style-type: none"> ip-header-length—Specifies the length of the IP header to inspect. 	0 to 16
specify-ip-id {yes no}	(Optional) Enables inspection of the IP identifier: <ul style="list-style-type: none"> ip-id—Specifies the IP ID to inspect. 	0 to 255
specify-ip-option-inspection {yes no}	(Optional) Enables inspection of the IP options: <ul style="list-style-type: none"> ip-option-inspection—Specifies the value of the IP option: <ul style="list-style-type: none"> ip-option—Specifies the IP OPTION code to match. ip-option-abnormal—Specifies the list of options is malformed. 	0 to 65535
specify-ip-payload-length {yes no}	(Optional) Enables inspection of the IP payload length: <ul style="list-style-type: none"> ip-payload-length—Specifies the length of IP payload to inspect. 	0 to 65535
specify-ip-tos {yes no}	(Optional) Specifies the IP type of service: <ul style="list-style-type: none"> ip-tos—Specifies the IP type of service to inspect. 	0 to 6 255
specify-ip-total-length {yes no}	(Optional) Enables inspection of the IP total length: <ul style="list-style-type: none"> ip-total-length—Specifies the total length of IP packet to inspect. 	0 to 65535

Table B-9 Atomic IP Engine Parameters (continued)

Parameter	Description	Value
specify-ip-ttl {yes no}	(Optional) Enables inspection of IP time-to-live: <ul style="list-style-type: none"> ip-ttl—Specifies the value of the IP TTL to inspect. 	0 to 255
specify-ip-version {yes no}	(Optional) Enables inspection of the IP version: <ul style="list-style-type: none"> ip-version—Specifies which IP version to inspect. 	0 to 16
specify-l4-protocol {yes no}	(Optional) Enables inspection of the Layer 4 protocol: <ul style="list-style-type: none"> l4-protocol—Specifies which Layer 4 protocol to inspect. 	icmp tcp udp other-protocol
specify-icmp-code {yes no}	(Optional) Enables inspection of the Layer 4 ICMP code: <ul style="list-style-type: none"> icmp-code—Specifies the value of the ICMP header CODE. 	0 to 65535
specify-icmp-id {yes no}	(Optional) Enables inspection of the Layer 4 ICMP ID: <ul style="list-style-type: none"> icmp-id—Specifies the value of the ICMP header IDENTIFIER. 	0 to 65535
specify-icmp-seq {yes no}	(Optional) Enables inspection of the Layer 4 ICMP sequence: <ul style="list-style-type: none"> icmp-seq—Specifies the ICMP sequence to inspect. 	0 to 65535
specify-icmp-type {yes no}	(Optional) Enables inspection of the ICMP header type: <ul style="list-style-type: none"> icmp-type—Specifies the value of the ICMP header TYPE. 	0 to 65535
specify-icmp-total-length {yes no}	(Optional) Enables inspection of the Layer 4 ICMP total header length: <ul style="list-style-type: none"> icmp-total-length—Specifies the value of the ICMP total length to inspect. 	0 to 65535
specify-other-ip-protocol-id {yes no}	(Optional) Enables inspection of the other Layer 4 protocols: <ul style="list-style-type: none"> other-ip-protocol-id—Specifies which single IP protocol ID or single range of IP protocol IDs for which to send alerts. 	0 to 255

Table B-9 Atomic IP Engine Parameters (continued)

Parameter	Description	Value
specify-dst-port {yes no}	(Optional) Enables the destination port for use: <ul style="list-style-type: none"> dst-port—Specifies the destination port of interest for this signature. 	0 to 65535
specify-src-port {yes no}	(Optional) Enables source port for use: <ul style="list-style-type: none"> src-port—Specifies the source port of interest for this signature. 	0 to 65535
specify-tcp-mask {yes no}	(Optional) Enables the TCP mask for use: <ul style="list-style-type: none"> tcp-mask—Specifies the mask used in TCP flags comparison: <ul style="list-style-type: none"> URG bit ACK bit PSH bit RST bit SYN bit FIN bit 	urg ack psh rst syn fin
specify-tcp-flags {yes no}	(Optional) Enables TCP flags for use: <ul style="list-style-type: none"> tcp-flags—Specifies the TCP flags to match when masked by mask: <ul style="list-style-type: none"> URG bit ACK bit PSH bit RST bit SYN bit FIN bit 	urg ack psh rst syn fin
specify-tcp-reserved {yes no}	(Optional) Enables TCP reserved for use: <ul style="list-style-type: none"> tcp-reserved—Specifies the value of TCP reserved. 	0 to 63
specify-tcp-header-length {yes no}	(Optional) Enables inspection of the Layer 4 TCP header length: <ul style="list-style-type: none"> tcp-header-length—Specifies the length of the TCP header used in inspection. 	0 to 60
specify-tcp-payload-length {yes no}	(Optional) Enables inspection of the Layer 4 TCP payload length: <ul style="list-style-type: none"> tcp-payload-length—Specifies the length of the TCP payload. 	0 to 65535

Table B-9 Atomic IP Engine Parameters (continued)

Parameter	Description	Value
specify-tcp-urg-pointer {yes no}	(Optional) Enables inspection of the L4 TCP URG pointer: <ul style="list-style-type: none"> tcp-urg-pointer—Specifies the value of the TCP URG flag to inspect. 	0 to 65535
specify-tcp-window-size {yes no}	(Optional) Enables inspection of the Layer 4 TCP window size: <ul style="list-style-type: none"> tcp-window-size—Specifies the window size of the TCP packet. 	0 to 65535
specify-udp-length {yes no}	(Optional) Enables inspection of the Layer 4 UDP length: <ul style="list-style-type: none"> udp-length—Fires an alert when the IP Data length is less than the UDP Header length. 	0 to 65535
specify-udp-valid-length {yes no}	(Optional) Enables inspection of the Layer 4 UDP valid length: <ul style="list-style-type: none"> udp-valid-length—Specifies UDP packet lengths that are considered valid and should not be inspected. 	0 to 65535
specify-udp-length-mismatch {yes no}	(Optional) Enables inspection of the Layer 4 UDP length mismatch: <ul style="list-style-type: none"> udp-length-mismatch—Fires an alert when the IP Data length is less than the UDP Header length. 	0 to 65535

1. Use the following syntax: x.x.x.x-z.z.z.z, for example, 10.10.10.1-10.10.10.254.

For More Information

For more information on the parameters common to all signature engines, see [Master Engine, page B-4](#).

Atomic IPv6 Engine

The Atomic IPv6 engine detects two IOS vulnerabilities that are stimulated by malformed IPv6 traffic. These vulnerabilities can lead to router crashes and other security issues. One IOS vulnerability deals with multiple first fragments, which cause a buffer overflow. The other one deals with malformed ICMPv6 Neighborhood Discovery options, which also cause a buffer overflow.

**Note**

IPv6 increases the IP address size from 32 bits to 128 bits, which supports more levels of addressing hierarchy, a much greater number of addressable nodes, and autoconfiguration of addresses.

Atomic IPv6 Signatures

There are eight Atomic IPv6 signatures. The Atomic IPv6 inspects Neighborhood Discovery protocol of the following types:

- Type 133—Router Solicitation
- Type 134—Router Advertisement
- Type 135—Neighbor Solicitation
- Type 136—Neighbor Advertisement
- Type 137—Redirect

**Note**

Hosts and routers use Neighborhood Discovery to determine the link-layer addresses for neighbors known to reside on attached links and to quickly purge cached values that become invalid. Hosts also use Neighborhood Discovery to find neighboring routers that will forward packets on their behalf.

Each Neighborhood Discovery type can have one or more Neighborhood Discovery options. The Atomic IPv6 engine inspects the length of each option for compliance with the legal values stated in RFC 2461. Violations of the length of an option results in an alert corresponding to the option type where the malformed length was encountered (signatures 1601 to 1605).

**Note**

The Atomic IPv6 signatures do not have any specific parameters to configure.

For More Information

For more information on the parameters common to all signature engines, see [Master Engine, page B-4](#).

Fixed Engine

The Fixed engines combine multiple regular expression patterns in to a single pattern matching table that allows a single search through the data. It supports ICMP, TCP, and UDP protocols. After a minimum inspection depth is reached (1 to 100 bytes), inspection stops. There are three Fixed engines: Fixed ICMP, Fixed TCP, and Fixed UDP.

**Note**

The Fixed TCP and Fixed UDP engines use the service-ports parameter as exclusion ports. The Fixed ICMP engine uses the service-ports parameter as excluded ICMP types.

Table B-10 lists the parameters specific to the Fixed ICMP engine.

Table B-10 Fixed ICMP Engine Parameters

Parameter	Description	Value
direction	Specifies the direction of traffic: <ul style="list-style-type: none"> Traffic from service port destined to client port. Traffic from client port destined to service port. 	from-service to-service
max-payload-inspect-length	Specifies the maximum inspection depth for the signature.	1 to 250
regex-string	Specifies the regular expression to search for in a single packet.	<i>string</i>
specify-exact-match-offset {yes no}	(Optional) Enables exact match offset: <ul style="list-style-type: none"> exact-match-offset—Specifies the exact stream offset the regex-string must report for a match to be valid. 	0 to 65535
specify-min-match-length {yes no}	(Optional) Enables minimum match length: <ul style="list-style-type: none"> min-match-length—Specifies the minimum number of bytes the regex-string must match. 	0 to 65535
specify-icmp-type {yes no}	(Optional) Enables inspection of the Layer 4 ICMP header type: <ul style="list-style-type: none"> icmp-type—Specifies the value of the ICMP header TYPE. 	0 to 65535
swap-attacker-victim	Swaps the attacker and victim addresses and ports (source and destination) in the alert message and in any actions taken.	false true (default)

Table B-11 lists the parameters specific to the Fixed TCP engine.

Table B-11 Fixed TCP Engine Parameters

Parameter	Description	Value
direction	Specifies the direction of traffic: <ul style="list-style-type: none"> Traffic from service port destined to client port. Traffic from client port destined to service port. 	from-service to-service
max-payload-inspect-length	Specifies the maximum inspection depth for the signature.	1 to 250
regex-string	Specifies the regular expression to search for in a single packet.	<i>string</i>

Table B-11 Fixed TCP Engine Parameters (continued)

Parameter	Description	Value
specify-exact-match-offset {yes no}	(Optional) Enables exact match offset: <ul style="list-style-type: none"> exact-match-offset—Specifies the exact stream offset the regex-string must report for a match to be valid. 	0 to 65535
specify-min-match-length {yes no}	(Optional) Enables minimum match length: <ul style="list-style-type: none"> min-match-length—Specifies the minimum number of bytes the regex-string must match. 	0 to 65535
exclude-service-ports {yes no}	Enables service ports for use: <ul style="list-style-type: none"> excluded-service-ports—Specifies a comma-separated list of ports or port ranges to exclude. 	0 to 65535 ¹ a-b[,c-d]
swap-attacker-victim	Swaps the attacker and victim addresses and ports (source and destination) in the alert message and in any actions taken.	false true (default)

1. The second number in the range must be greater than or equal to the first number.

Table B-12 lists the parameters specific to the Fixed UDP engine.

Table B-12 Fixed UDP Engine Parameters

Parameter	Description	Value
direction	Specifies the direction of traffic: <ul style="list-style-type: none"> Traffic from service port destined to client port. Traffic from client port destined to service port 	from-service to-service
max-payload-inspect-length	Specifies the maximum inspection depth for the signature.	1 to 250
regex-string	Specifies the regular expression to search for in a single packet.	<i>string</i>
specify-exact-match-offset {yes no}	(Optional) Enables exact match offset: <ul style="list-style-type: none"> exact-match-offset—Specifies the exact stream offset the regex-string must report for a match to be valid. 	0 to 65535
specify-min-match-length {yes no}	(Optional) Enables minimum match length: <ul style="list-style-type: none"> min-match-length—Specifies the minimum number of bytes the regex-string must match. 	0 to 65535

Table B-12 Fixed UDP Engine Parameters (continued)

Parameter	Description	Value
exclude-service-ports {yes no}	Enables service ports for use: <ul style="list-style-type: none"> excluded-service-ports—Specifies a comma-separated list of ports or port ranges to exclude. 	0 to 65535 ¹ a-b[,c-d]
swap-attacker-victim	Swaps the attacker and victim addresses and ports (source and destination) in the alert message and in any actions taken.	false true (default)

1. The second number in the range must be greater than or equal to the first number.

For More Information

- For more information on the parameters common to all signature engines, see [Master Engine, page B-4](#).
- For a list of the signature regular expression syntax, see [Regular Expression Syntax, page B-9](#).

Flood Engine

The Flood engines define signatures that watch for any host or network sending multiple packets to a single host or network. For example, you can create a signature that fires when 150 or more packets per second (of the specific type) are found going to the victim host. There are two Flood engines: Flood Host and Flood Net.

[Table B-13](#) lists the parameters specific to the Flood Host engine.

Table B-13 Flood Host Engine Parameters

Parameter	Description	Value
protocol	Specifies which kind of traffic to inspect.	ICMP UDP
rate	Specifies the threshold number of packets per second.	0 to 65535 ¹
icmp-type	Specifies the value for the ICMP header type.	0 to 65535
dst-ports	Specifies the destination ports when you choose UDP protocol.	0 to 65535 ² a-b[,c-d]
src-ports	Specifies the source ports when you choose UDP protocol.	0 to 65535 ² a-b[,c-d]

- An alert fires when the rate is greater than the packets per second.
- The second number in the range must be greater than or equal to the first number.

Table B-14 lists the parameters specific to the Flood Net engine.

Table B-14 Flood Net Engine Parameters

Parameter	Description	Value
gap	Specifies the gap of time allowed (in seconds) for a flood signature.	0 to 65535
peaks	Specifies the number of allowed peaks of flood traffic.	0 to 65535
protocol	Specifies which kind of traffic to inspect.	ICMP TCP UDP
rate	Specifies the threshold number of packets per second.	0 to 65535 ¹
sampling-interval	Specifies the interval used for sampling traffic.	1 to 3600
icmp-type	Specifies the value for the ICMP header type.	0 to 65535

1. An alert fires when the rate is greater than the packets per second.

For More Information

For more information on the parameters common to all signature engines, see [Master Engine, page B-4](#).

Meta Engine



Caution

A large number of Meta engine signatures could adversely affect overall sensor performance.

The Meta engine defines events that occur in a related manner within a sliding time interval. This engine processes events rather than packets. As signature events are generated, the Meta engine inspects them to determine if they match any or several Meta definitions. The Meta engine generates a signature event after all requirements for the event are met.

All signature events are handed off to the Meta engine by the Signature Event Action Processor. The Signature Event Action Processor hands off the event after processing the minimum hits option. Summarization and event action are processed after the Meta engine has processed the component events.

Table B-15 lists the parameters specific to the Meta engine.

Table B-15 Meta Engine Parameters

Parameter	Description	Value
component-list	<p>Specifies the Meta engine component:</p> <ul style="list-style-type: none"> • edit—Edits an existing entry. • insert—Inserts a new entry into the list: <ul style="list-style-type: none"> – begin—Places the entry at the beginning of the active list. – end—Places the entry at the end of the active list. – inactive—Places the entry into the inactive list. – before—Places the entry before the specified entry. – after—Places the entry after the specified entry. • move—Moves an entry in the list. • component-count—Specifies the number of times component must fire before this component is satisfied • component-sig-id—Specifies the signature ID of the signature to match this component on. • component-subsig-id—Specifies the subsignature ID of the signature to match this component on. • is-not-component—Specifies that the component is a NOT component. 	<p><i>name1</i></p> <p>true false</p>
component-list-in-order	Specifies to fire the component list in order.	true false
all-components-required	Specifies to use all components.	true false
all-not-components-required	Specifies to use all of the NOT components.	true false
swap-attacker-victim	Swaps the attacker and victim addresses and ports (source and destination) in the alert message and in any actions taken.	true false (default)
meta-reset-interval	Specifies the time in seconds to reset the Meta signature.	0 to 3600
meta-key	<p>Specifies the storage type for the Meta signature:</p> <ul style="list-style-type: none"> • Attacker address • Attacker and victim addresses • Attacker and victim addresses and ports • Victim address 	<ul style="list-style-type: none"> • Axxx • AxBx • AaBb • xxBx
unique-victim-ports	Specifies the number of unique victims ports required per Meta signature.	1 to 256

For More Information

- For an example of a custom Meta engine signature, see [Example Meta Engine Signature, page 7-52](#).
- For more information on the parameters common to all signature engines, see [Master Engine, page B-4](#).

Multi String Engine

**Caution**

The Multi String engine can have a significant impact on memory usage.

The Multi String engine lets you define signatures that inspect Layer 4 transport protocol (ICMP, TCP, and UDP) payloads using multiple string matches for one signature. You can specify a series of regular expression patterns that must be matched to fire the signature. For example, you can define a signature that looks for regex 1 followed by regex 2 on a UDP service. For UDP and TCP you can specify port numbers and direction. You can specify a single source port, a single destination port, or both ports. The string matching takes place in both directions.

Use the Multi String engine when you need to specify more than one Regex pattern. Otherwise, you can use the String ICMP, String TCP, or String UDP engine to specify a single Regex pattern for one of those protocols.

[Table B-16](#) lists the parameters specific to the Multi String Engine.

Table B-16 Multi String Engine Parameters

Parameter	Description	Value
inspect-length	Specifies the length of the stream or packet that must contain all offending strings for the signature to fire.	0 to 4294967295
protocol	Specifies the Layer 4 protocol selection.	icmp tcp udp
regex-component	Specifies the list of Regex components: <ul style="list-style-type: none"> • regex-string—Specifies the string to search for. • spacing-type—Specifies the type of spacing required from the match before or from the beginning of the stream/packet if it is the first entry in the list. 	list (1 to 16 items) exact minimum
port-selection	Specifies the type of TCP or UDP port to inspect: <ul style="list-style-type: none"> • both-ports—Specifies both source and destination port. • dest-ports—Specifies a range of destination ports. • source-ports—Specifies a range of source ports.¹ 	0 to 65535 ²

Table B-16 Multi String Engine Parameters (continued)

Parameter	Description	Value
exact-spacing	Specifies the exact number of bytes that must be between this Regex string and the one before, or from the beginning of the stream/packet if it is the first entry in the list.	0 to 4294967296
min-spacing	Specifies the minimum number of bytes that must be between this Regex string and the one before, or from the beginning of the stream/packet if it is the first entry in the list.	0 to 4294967296
swap-attacker-victim	Swaps the attacker and victim addresses and ports (source and destination) in the alert message and in any actions taken.	true false (default)

1. Port matching is performed bidirectionally for both the client-to-server and server-to-client traffic flow directions. For example, if the source-ports value is 80, in a client-to-server traffic flow direction, inspection occurs if the client port is 80. In a server-to-client traffic flow direction, inspection occurs if the server port is port 80.
2. A valid value is a comma-separated list of integer ranges a-b[,c-d] within 0 to 65535. The second number in the range must be greater than or equal to the first number.

For More Information

- For more information on the parameters common to all signature engines, see [Master Engine, page B-4](#).
- For a list of the signature regular expression syntax, see [Regular Expression Syntax, page B-9](#).

Normalizer Engine



Note

You cannot add custom signatures to the Normalizer engine. You can tune the existing ones.

The Normalizer engine deals with IP fragment reassembly and TCP stream reassembly. With the Normalizer engine you can set limits on system resource usage, for example, the maximum number of fragments the sensor tries to track at the same time. Sensors in promiscuous mode report alerts on violations. Sensors in inline mode perform the action specified in the event action parameter, such as produce-alert, deny-packet-inline, and modify-packet-inline.



Caution

For signature 3050 Half Open SYN Attack, if you choose modify-packet-inline as the action, you can see as much as 20 to 30% performance degradation while the protection is active. The protection is only active during an actual SYN flood.

IP Fragmentation Normalization

Intentional or unintentional fragmentation of IP datagrams can hide exploits making them difficult or impossible to detect. Fragmentation can also be used to circumvent access control policies like those found on firewalls and routers. And different operating systems use different methods to queue and dispatch fragmented datagrams. If the sensor has to check for all possible ways that the end host can

reassemble the datagrams, the sensor becomes vulnerable to DoS attacks. Reassembling all fragmented datagrams inline and only forwarding completed datagrams and refragmenting the datagram if necessary, prevents this. The IP Fragmentation Normalization unit performs this function.

TCP Normalization

Through intentional or natural TCP session segmentation, some classes of attacks can be hidden. To make sure policy enforcement can occur with no false positives and false negatives, the state of the two TCP endpoints must be tracked and only the data that is actually processed by the real host endpoints should be passed on. Overlaps in a TCP stream can occur, but are extremely rare except for TCP segment retransmits. Overwrites in the TCP session should not occur. If overwrites do occur, someone is intentionally trying to elude the security policy or the TCP stack implementation is broken. Maintaining full information about the state of both endpoints is not possible unless the sensor acts as a TCP proxy. Instead of the sensor acting as a TCP proxy, the segments are ordered properly and the Normalizer engine looks for any abnormal packets associated with evasion and attacks.

IPv6 Fragments

The Normalizer engine can reassemble IPv6 fragments and forward the reassembled buffer for inspection and actions by other engines and processors. The following differences exist between IPv4 and IPv6:

- modify-packet-inline for Normalizer engine signatures has no effect on IPv6 datagrams.
- Signature 1206 (IP Fragment Too Small) does not fire for IPv6 datagrams. Signature 1741 in the Atomic IP Advanced engine fires for IPv6 fragments that are too small.
- Signature 1202 allows 48 additional bytes beyond the max-datagram-size for IPv6 because of the longer IPv6 header fields.

TCP Normalizer Signature Warning

You receive the following warning if you disable a default-enabled TCP Normalizer signature or remove a default-enabled modify-packet-inline, deny-packet-inline, or deny-connection-inline action:

Use caution when disabling, retiring, or changing the event action settings of a <Sig ID> TCP Normalizer signature for a sensor operating in IPS mode. The TCP Normalizer signature default values are essential for proper operation of the sensor.

If the sensor is seeing duplicate packets, consider assigning the traffic to multiple virtual sensors. If you are having problems with asymmetric or out-of-order TCP packets, consider changing the normalizer mode from strict evasion protection to asymmetric mode protection. Contact Cisco TAC if you require further assistance.

Table B-17 lists the parameters that are specific to the Normalizer engine.

Table B-17 Normalizer Engine Parameters

Parameter	Description
edit-default-sigs-only	Editable signatures.
specify-fragment-reassembly-timeout	(Optional) Enables fragment reassembly timeout.
specify-hijack-max-old-ack	(Optional) Enables hijack-max-old-ack.
specify-max-dgram-size	(Optional) Enables maximum datagram size.
specify-max-fragments	(Optional) Enables maximum fragments: <ul style="list-style-type: none"> • max-fragments—Lets you specify the number of maximum fragments.
specify-max-fragments-per-dgram	(Optional) Enables maximum fragments per datagram.

Table B-17 Normalizer Engine Parameters (continued)

Parameter	Description
specify-max-last-fragments	(Optional) Enables maximum last fragments.
specify-max-partial-dgrams	(Optional) Enables maximum partial datagrams.
specify-max-small-fragss	(Optional) Enables maximum small fragments.
specify-min-fragment-size	(Optional) Enables minimum fragment size.
specify-service-ports	(Optional) Enables service ports.
specify-syn-flood-max-embryonic	(Optional) Enables SYN flood maximum embryonic.
specify-tcp-closed-timeout	(Optional) Enables TCP closed timeout.
specify-tcp-embryonic-timeout	(Optional) Enables TCP embryonic timeout.
specify-tcp-idle-timeout	(Optional) Enables TCP idle timeout: <ul style="list-style-type: none"> tcp-idle-timeout—Lets you specify the TCP idle timeout time.
specify-tcp-max-mss	(Optional) Enables TCP maximum mss.
specify-tcp-max-queue	(Optional) Enables TCP maximum queue.
specify-tcp-min-mss	(Optional) Enables TCP minimum mss.
specify-tcp-option-number	(Optional) Enables TCP option number.

For More Information

- For the procedure for configuring IP fragment reassembly signatures in the Normalizer engine, see [Configuring IP Fragment Reassembly, page 7-34](#).
- For the procedure for configuring TCP stream reassembly signatures in the Normalizer engine, see [Configuring TCP Stream Reassembly, page 7-37](#).
- For more information on the parameters common to all signature engines, see [Master Engine, page B-4](#).

Service Engines

This section describes the Service engines, and contains the following topics:

- [Understanding the Service Engines, page B-39](#)
- [Service DNS Engine, page B-39](#)
- [Service FTP Engine, page B-40](#)
- [Service Generic Engine, page B-41](#)
- [Service H225 Engine, page B-43](#)
- [Service HTTP Engine, page B-45](#)
- [Service IDENT Engine, page B-47](#)
- [Service MSRPC Engine, page B-48](#)
- [Service MSSQL Engine, page B-50](#)
- [Service NTP Engine, page B-51](#)

- [Service P2P Engine, page B-52](#)
- [Service RPC Engine, page B-52](#)
- [Service SMB Advanced Engine, page B-54](#)
- [Service SNMP Engine, page B-56](#)
- [Service SSH Engine, page B-57](#)
- [Service TNS Engine, page B-58](#)

Understanding the Service Engines

The Service engines analyze Layer 5+ traffic between two hosts. These are one-to-one signatures that track persistent data. The engines analyze the Layer 5+ payload in a manner similar to the live service.

The Service engines have common characteristics but each engine has specific knowledge of the service that it is inspecting. The Service engines supplement the capabilities of the generic string engine specializing in algorithms where using the string engine is inadequate or undesirable.

Service DNS Engine

The Service DNS engine specializes in advanced DNS decode, which includes anti-evasive techniques, such as following multiple jumps. It has many parameters, such as lengths, opcodes, strings, and so forth. The Service DNS engine is a biprotocol inspector operating on both TCP and UDP port 53. It uses the stream for TCP and the quad for UDP.

[Table B-18](#) lists the parameters specific to the Service DNS engine.

Table B-18 Service DNS Engine Parameters

Parameter	Description	Value
protocol	Specifies the protocol of interest for this inspector.	tcp udp
specify-query-chaos-string {yes no}	(Optional) Enables the DNS Query Class Chaos String: <ul style="list-style-type: none"> • query-chaos-string—Specifies the query chaos string to search on. 	<i>query-chaos-string</i>
specify-query-class {yes no}	(Optional) Enables the query class: <ul style="list-style-type: none"> • query-class—Specifies the DNS Query Class 2 Byte Value. 	0 to 65535
specify-query-invalid-domain-name {yes no}	(Optional) Enables the query invalid domain name: <ul style="list-style-type: none"> • query-invalid-domain-name—Specifies the DNS Query Length greater than 255. 	no yes
specify-query-jump-count-exceeded {yes no}	(Optional) Enables query jump count exceeded: <ul style="list-style-type: none"> • query-jump-count-exceeded—DNS compression counter. 	no yes

Table B-18 Service DNS Engine Parameters (continued)

Parameter	Description	Value
specify-query-opcode {yes no}	(Optional) Enables query opcode: <ul style="list-style-type: none"> query-opcode—Specifies the DNS Query Opcode 1 byte Value. 	0 to 65535
specify-query-record-data-invalid {yes no}	(Optional) Enables query record data invalid: <ul style="list-style-type: none"> query-record-data-invalid—Specifies the DNS Record Data incomplete. 	no yes
specify-query-record-data-len {yes no}	(Optional) Enables the query record data length: <ul style="list-style-type: none"> query-record-data-len—Specifies the DNS Response Record Data Length. 	0 to 65535
specify-query-src-port-53 {yes no}	(Optional) Enables the query source port 53: <ul style="list-style-type: none"> query-src-port-53—Specifies the DNS packet source port 53. 	no yes
specify-query-stream-len {yes no}	(Optional) Enables the query stream length: <ul style="list-style-type: none"> query-stream-len—Specifies the DNS Packet Length. 	0 to 65535
specify-query-type {yes no}	(Optional) Enables the query type: <ul style="list-style-type: none"> query-type—Specifies the DNS Query Type 2 Byte Value. 	0 to 65535
specify-query-value {yes no}	(Optional) Enables the query value: <ul style="list-style-type: none"> query-value—Specifies the Query 0 Response 1. 	no yes

For More Information

For more information on the parameters common to all signature engines, see [Master Engine, page B-4](#).

Service FTP Engine

The Service FTP engine specializes in FTP port command decode, trapping invalid **port** commands and the PASV port spoof. It fills in the gaps when the String engine is not appropriate for detection. The parameters are Boolean and map to the various error trap conditions in the **port** command decode. The Service FTP engine runs on TCP ports 20 and 21. Port 20 is for data and the Service FTP engine does not do any inspection on this. It inspects the control transactions on port 21.

Table B-19 lists the parameters that are specific to the Service FTP engine.

Table B-19 Service FTP Engine Parameters

Parameter	Description	Value
direction	Specifies the direction of traffic: <ul style="list-style-type: none"> Traffic from service port destined to client port. Traffic from client port destined to service port. 	from-service to-service
ftp-inspection-type	Specifies the type of inspection to perform: <ul style="list-style-type: none"> Looks for an invalid address in the FTP port command. Looks for an invalid port in the FTP port command. Looks for the PASV port spoof. 	bad-port-cmd-address bad-port-cmd-port pasv
service-ports	Specifies a comma-separated list of ports or port ranges where the target service resides.	0 to 65535 ¹ a-b[,c-d]
swap-attacker-victim	Swaps the attacker and victim addresses and ports (source and destination) in the alert message and in any actions taken.	true false (default)

1. The second number in the range must be greater than or equal to the first number.

For More Information

For more information on the parameters common to all signature engines, see [Master Engine, page B-4](#).

Service Generic Engine

The Service Generic engine allows programmatic signatures to be issued in a config-file-only signature update. It has a simple machine and assembly language that is defined in the configuration file. It runs the machine code (distilled from the assembly language) through its virtual machine, which processes the instructions and pulls the important pieces of information out of the packet and runs them through the comparisons and operations specified in the machine code. It is intended as a rapid signature response engine to supplement the String and State engines.

New functionality adds the Regex parameter to the Service Generic engine and enhanced instructions. The Service Generic engine can analyze traffic based on the mini-programs that are written to parse the packets. These mini-programs are composed of commands, which dissect the packet and look for certain conditions.



Note

You cannot use the Service Generic engine to create custom signatures.



Caution

Due to the proprietary nature of this complex language, we do not recommend that you edit the Service Generic engine signature parameters other than severity and event action.

Table B-20 lists the parameters specific to the Service Generic engine.

Table B-20 Service Generic Engine Parameters

Parameter	Description	Value
specify-dst-port {yes no}	(Optional) Enables the destination port: <ul style="list-style-type: none"> dst-port—Specifies the destination port of interest for this signature. 	0 to 65535
specify-ip-protocol {yes no}	(Optional) Enables IP protocol: <ul style="list-style-type: none"> ip-protocol—Specifies the IP protocol this inspector should examine. 	0 to 255
specify-payload-source {yes no}	(Optional) Enables payload source inspection: <ul style="list-style-type: none"> payload-source—Specifies the payload source inspection for the following types: <ul style="list-style-type: none"> Inspects ICMP data Inspects Layer 2 headers Inspects Layer 3 headers Inspects Layer 4 headers Inspects TCP data Inspects UDP data 	icmp-data l2-header l3-header l4-header tcp-data udp-data
specify-src-port {yes no}	(Optional) Enables the source port: <ul style="list-style-type: none"> src-port—Specifies the source port of interest for this signature. 	0 to 65535
specify-regex-string {yes no}	Specifies the regular expression to look for when the policy type is Regex: <ul style="list-style-type: none"> regex-string—Specifies a regular expression to search for in a single TCP packet. (Optional) specify-min-match-length—Enables minimum match length for use: <ul style="list-style-type: none"> min-match-length—Specifies the minimum length of the Regex match required to constitute a match. 	<i>string</i> 0 to 65535
swap-attacker-victim	Swaps the attacker and victim addresses and ports (source and destination) in the alert message and in any actions taken.	true false (default)

For More Information

- For more information on the parameters common to all signature engines, see [Master Engine, page B-4](#).
- For a list of the signature regular expression syntax, see [Regular Expression Syntax, page B-9](#).

Service H225 Engine

The Service H225 engine analyzes H225.0 protocol, which consists of many subprotocols and is part of the H.323 suite. H.323 is a collection of protocols and other standards that together enable conferencing over packet-based networks.

H.225.0 call signaling and status messages are part of the H.323 call setup. Various H.323 entities in a network, such as the gatekeeper and endpoint terminals, run implementations of the H.225.0 protocol stack. The Service H225 engine analyzes H225.0 protocol for attacks on multiple H.323 gatekeepers, VoIP gateways, and endpoint terminals. It provides deep packet inspection for call signaling messages that are exchanged over TCP PDUs. The Service H225 engine analyzes the H.225.0 protocol for invalid H.255.0 messages, and misuse and overflow attacks on various protocol fields in these messages.

H.225.0 call signaling messages are based on Q.931 protocol. The calling endpoint sends a Q.931 setup message to the endpoint that it wants to call, the address of which it procures from the admissions procedure or some lookup means. The called endpoint either accepts the connection by transmitting a Q.931 connect message or rejects the connection. When the H.225.0 connection is established, either the caller or the called endpoint provides an H.245 address, which is used to establish the control protocol (H.245) channel.

Especially important is the SETUP call signaling message because this is the first message exchanged between H.323 entities as part of the call setup. The SETUP message uses many of the commonly found fields in the call signaling messages, and implementations that are exposed to probable attacks will mostly also fail the security checks for the SETUP messages. Therefore, it is highly important to check the H.225.0 SETUP message for validity and enforce checks on the perimeter of the network.

The Service H225 engine has built-in signatures for TPKT validation, Q.931 protocol validation, and ASN.1PER validations for the H225 SETUP message. ASN.1 is a notation for describing data structures. PER uses a different style of encoding. It specializes the encoding based on the data type to generate much more compact representations.

You can tune the Q.931 and TPKT length signatures and you can add and apply granular signatures on specific H.225 protocol fields and apply multiple pattern search signatures of a single field in Q.931 or H.225 protocol.

The Service H225 engine supports the following features:

- TPKT validation and length check
- Q.931 information element validation
- Regular expression signatures on text fields in Q.931 information elements
- Length checking on Q.931 information elements
- SETUP message validation
- ASN.1 PER encode error checks
- Configuration signatures for fields like ULR-ID, E-mail-ID, h323-id, and so forth for both regular expression and length.

There is a fixed number of TPKT and ASN.1 signatures. You cannot create custom signatures for these types. For TPKT signatures, you should only change the value-range for length signatures. You should not change any parameters for ASN.1. For Q.931 signatures, you can add new regular expression signatures for text fields. For SETUP signatures, you can add signatures for length and regular expression checks on various SETUP message fields.

Table B-21 lists parameters specific to the Service H225 engine.

Table B-21 Service H.225 Engine Parameters

Parameter	Description	Value
message-type	Specifies the type of H225 message to which the signature applies: <ul style="list-style-type: none"> • SETUP • ASN.1-PER • Q.931 • TPKT 	asn.1-per q.931 setup tpkt
policy-type	Specifies the type of H225 policy to which the signature applies: <ul style="list-style-type: none"> • Inspects field length. • Inspects presence. If certain fields are present in the message, an alert is sent. • Inspects regular expressions. • Inspects field validations. • Inspects values. <p>Note Regex and presence are not valid for TPKT signatures.</p>	length presence regex validate value
specify-field-name {yes no}	(Optional) Enables field name for use. Gives a dotted representation of the field name to which this signature applies. <ul style="list-style-type: none"> • field-name—Specifies the field name to inspect. <p>Note Only valid for SETUP and Q.931 message types.</p>	1 to 512
specify-invalid-packet-index {yes no}	(Optional) Enables invalid packet index for use for specific errors in ASN, TPKT, and other errors that have fixed mapping. <ul style="list-style-type: none"> • invalid-packet-index—Specifies the inspection for invalid packet index. 	0 to 255

Table B-21 Service H.225 Engine Parameters (continued)

Parameter	Description	Value
specify-regex-string {yes no}	<p>Specifies the regular expression to look for when the policy type is Regex:</p> <ul style="list-style-type: none"> regex-string—Specifies a regular expression to search for in a single TCP packet. (Optional) specify-min-match-length—Enables minimum match length for use: <ul style="list-style-type: none"> min-match-length—Specifies the minimum length of the Regex match required to constitute a match. <p>Note This is never set for TPKT signatures.</p>	<p>string</p> <p>0 to 65535</p>
specify-value-range {yes no}	<p>Enables value range for use:</p> <ul style="list-style-type: none"> value-range—Specifies the range of values. <p>Note Valid for the length or value policy types (0x00 to 6535). Not valid for other policy types.</p>	<p>0 to 65535¹</p> <p>a-b</p>
swap-attacker-victim	Swaps the attacker and victim addresses and ports (source and destination) in the alert message and in any actions taken.	true false (default)

1. The second number in the range must be greater than or equal to the first number.

For More Information

- For more information on the parameters common to all signature engines, see [Master Engine, page B-4](#).
- For a list of the signature regular expression syntax, see [Regular Expression Syntax, page B-9](#).

Service HTTP Engine

The Service HTTP engine is a service-specific string-based pattern-matching inspection engine. The HTTP protocol is one of the most commonly used in networks of today. In addition, it requires the most amount of preprocessing time and has the most number of signatures requiring inspection making it critical to the overall performance of the system.

The Service HTTP engine uses a Regex library that can combine multiple patterns into a single pattern-matching table allowing a single search through the data. This engine searches traffic directed only to web services, or HTTP requests. You cannot inspect return traffic with this engine. You can specify separate web ports of interest in each signature in this engine.

HTTP deobfuscation is the process of decoding an HTTP message by normalizing encoded characters to ASCII equivalent characters. It is also known as ASCII normalization.

Before an HTTP packet can be inspected, the data must be deobfuscated or normalized to the same representation that the target system sees when it processes the data. It is ideal to have a customized decoding technique for each host target type, which involves knowing what operating system and web server version is running on the target. The Service HTTP engine has default deobfuscation behavior for the Microsoft IIS web server.

Table B-22 lists the parameters specific the Service HTTP engine.

Table B-22 Service HTTP Engine Parameters

Parameter	Description	Value
de-obfuscate	Applies anti-evasive deobfuscation before searching.	true false
max-field-sizes	Enables maximum field sizes grouping.	—
specify-max-arg-field-length {yes no}	(Optional) Enables maximum argument field length: <ul style="list-style-type: none"> max-arg-field-length—Specifies the maximum length of the arguments field. 	0 to 65535
specify-max-header-field-length {yes no}	(Optional) Enables maximum header field length: <ul style="list-style-type: none"> max-header-field-length—Specifies the maximum length of the header field. 	0 to 65535
specify-max-request-length {yes no}	(Optional) Enables maximum request field length: <ul style="list-style-type: none"> max-request-length—Specifies the maximum length of the request field. 	0 to 65535
specify-max-uri-field-length {yes no}	(Optional) Enables the maximum URI field length: <ul style="list-style-type: none"> max-uri-field-length—Specifies the maximum length of the URI field. 	0 to 65535
regex	Enables regular expression grouping.	—
specify-arg-name-regex {yes no}	(Optional) Enables searching the Arguments field for a specific regular expression: <ul style="list-style-type: none"> arg-name-regex—Specifies the regular expression to search for in the HTTP Arguments field (after the ? and in the Entity body as defined by Content-Length). 	—
specify-header-regex {yes no}	(Optional) Enables searching the Header field for a specific regular expression: <ul style="list-style-type: none"> header-regex—Specifies the regular expression to search in the HTTP Header field. <p>Note The Header is defined after the first CRLF and continues until CRLF CRLF.</p>	—

Table B-22 Service HTTP Engine Parameters (continued)

Parameter	Description	Value
specify-request-regex {yes no}	(Optional) Enables searching the Request field for a specific regular expression: <ul style="list-style-type: none"> request-regex—Specifies the regular expression to search in both HTTP URI and HTTP Argument fields. specify-min-request-match-length—Enables setting a minimum request match length: <ul style="list-style-type: none"> min-request-match-length—Specifies the minimum request match length. 	0 to 65535
specify-uri-regex {yes no}	(Optional) Specifies the regular expression to search in HTTP URI field. <p>Note The URI field is defined to be after the HTTP method (GET, for example) and before the first CRLF.</p> <p>Note The regular expression is protected, which means you cannot change the value.</p>	[/\][a-zA-Z][a-zA-Z][a-zA-Z][a-zA-Z][a-zA-Z][a-zA-Z][.].jpeg
service-ports	Specifies a comma-separated list of ports or port ranges where the target service resides.	0 to 65535 ¹ a-b[,c-d]
swap-attacker-victim	Swaps the attacker and victim addresses and ports (source and destination) in the alert message and in any actions taken.	true false (default)

1. The second number in the range must be greater than or equal to the first number.

For More Information

- For an example Service HTTP custom signature, see [Example Service HTTP Engine Signature, page 7-50](#).
- For more information on the parameters common to all signature engines, see [Master Engine, page B-4](#).
- For a list of the signature regular expression syntax, see [Regular Expression Syntax, page B-9](#).

Service IDENT Engine

The Service IDENT engine inspects TCP port 113 traffic. It has basic decode and provides parameters to specify length overflows. For example, when a user or program at computer A makes an IDENT request of computer B, it may only ask for the identity of users of connections between A and B. The IDENT server on B listens for connections on TCP port 113. The client at A establishes a connection, then specifies which connection it wants identification for by sending the numbers of the ports on A and B that the connection is using. The server at B determines what user is using that connection, and replies to A with a string that names that user. The Service IDENT engine inspects the TCP port 113 for IDENT abuse.

Table B-23 lists the parameters specific to the Service IDENT engine.

Table B-23 Service IDENT Engine Parameters

Parameter	Description	Value
inspection-type	Specifies the type of inspection to perform.	has-newline has-bad-port size
has-newline	Inspects payload for a nonterminating new line character.	—
has-bad-port	Inspects payload for a bad port.	—
size	Inspects for payload length longer than this: <ul style="list-style-type: none"> max-bytes—Specifies the maximum bytes for the payload length. 	0 to 65535
service-ports	Specifies a comma-separated list of ports or port ranges where the target service resides.	0 to 65535 ¹ a-b[,c-d]
direction	Specifies the direction of the traffic: <ul style="list-style-type: none"> Traffic from service port destined to client port. Traffic from client port destined to service port. 	from-service to-service

1. The second number in the range must be greater than or equal to the first number.

For More Information

For more information on the parameters common to all signature engines, see [Master Engine, page B-4](#).

Service MSRPC Engine

The Service MSRPC engine processes MSRPC packets. MSRPC allows for cooperative processing between multiple computers and their application software in a networked environment. It is a transaction-based protocol, implying that there is a sequence of communications that establishes the channel and passes processing requests and replies.

MSRPC is an ISO Layer 5-6 protocol and is layered on top of other transport protocols such as UDP, TCP, and SMB. The MSRPC engine contains facilities to allow for fragmentation and reassembly of the MSRPC PDUs.

This communication channel is the source of recent Windows NT, Windows 2000, and Window XP security vulnerabilities. The Service MSRPC engine only decodes the DCE and RPC protocol for the most common transaction types.

Table B-24 lists the parameters specific to the Service MSRPC engine.

Table B-24 Service MSRPC Engine Parameters

Parameter	Description	Value
protocol	Enables the protocol of interest for this inspector: <ul style="list-style-type: none"> type—Specifies UDP or TCP. 	tcp udp
specify-flags {yes no}	Enables the flags to set: <ul style="list-style-type: none"> msrpc-flags—Specifies MSRPC TCP flags. msrpc-tcp-flags-mask—Specifies the MSRPC TCP flags mask. 	concurrent-execution did-not-execute first-fragment last-fragment maybe-semantics object-uuid reserved
specify-operation {yes no}	(Optional) Enables using MSRPC operation: <ul style="list-style-type: none"> operation—Specifies the MSRPC operation requested. <p>Note Required for SMB_COM_TRANSACTION commands. Exact match.</p>	0 to 65535
swap-attacker-victim	Swaps the attacker and victim addresses and ports (source and destination) in the alert message and in any actions taken.	true false (default)

Table B-24 Service MSRPC Engine Parameters (continued)

Parameter	Description	Value
specify-regex-string {yes no}	(Optional) Enables using a regular expression string: <ul style="list-style-type: none"> • specify-exact-match-offset—Enables the exact match offset: <ul style="list-style-type: none"> – exact-match-offset—Specifies the exact stream offset the regular expression string must report for a match to be valid. • specify-min-match-length—Enables the minimum match length: <ul style="list-style-type: none"> – min-match-length—Specifies the minimum number of bytes the regular expression string must match. • specify-min-match-offset—Enables the minimum match length: <ul style="list-style-type: none"> – min-match-offset—Specifies the minimum stream offset the regular expression string must report for a match to be valid. • specify-max-match-offset—Enables the maximum match offset: <ul style="list-style-type: none"> – max-match-offset—Specifies the maximum stream offset the regular expression string must report for a match to be valid. 	0 to 65535
specify-uuid {yes no}	(Optional) Enables UUID: <ul style="list-style-type: none"> • uuid—Specifies the MSRPC UUID field. 	000001a000000000c0000000000046

For More Information

- For more information on the parameters common to all signature engines, see [Master Engine, page B-4](#).
- For a list of the signature regular expression syntax, see [Regular Expression Syntax, page B-9](#).

Service MSSQL Engine

The Service MSSQL engine inspects the protocol used by the Microsoft SQL server. There is one MSSQL signature. It fires an alert when it detects an attempt to log in to an MSSQL server with the default sa account. You can add custom signatures based on MSSQL protocol values, such as login username and whether a password was used.

Table B-25 lists the parameters specific to the Service MSSQL engine.

Table B-25 Service MSSQL Engine Parameters

Parameter	Description	Value
password-present	Specifies whether or not a password was used in an MS SQL login.	true false
specify-sql-username	(Optional) Enables using an SQL username: <ul style="list-style-type: none"> sql-username—Specifies the username (exact match) of user logging in to MS SQL service. 	sa

For More Information

For more information on the parameters common to all signature engines, see [Master Engine, page B-4](#).

Service NTP Engine

The Service NTP engine inspects NTP protocol. There is one NTP signature, the NTP readvar overflow signature, which fires an alert if a readvar command is seen with NTP data that is too large for the NTP service to capture. You can tune this signature and create custom signatures based on NTP protocol values, such as mode and size of control packets.

Table B-26 lists the parameters specific to the Service NTP engine.

Table B-26 Service NTP Engine Parameters

Parameter	Description	Value
inspection-type	Specifies the type of inspection to perform.	inspect-ntp-packets is-invalid-data-packet is-non-ntp-traffic
inspect-ntp-packets	Enables inspection of NTP packets: <ul style="list-style-type: none"> control-opcode—Specifies the opcode number of an NTP control packet according to RFC1305, Appendix B. max-control-data-size—Specifies the maximum allowed amount of data sent in a control packet. mode—Specifies the mode of operation of the NTP packet per RFC 1305. 	0 to 65535
is-invalid-data-packet	Enables inspection of invalid NTP data packets and checks the structure of the NTP data packet to make sure it is the correct size.	—
is-non-ntp-traffic	Enables the inspection of nonNTP packets on an NTP port.	—

For More Information

For more information on the parameters common to all signature engines, see [Master Engine, page B-4](#).

Service P2P Engine

P2P networks use nodes that can simultaneously function as both client and server for the purpose of file sharing. P2P networks often contain copyrighted material and their use on a corporate network can violate company policy. The Service P2P engine monitors such networks and provides optimized TCP and UDP P2P protocol identification. The Service P2P engine has the following characteristics:

- Listens on all TCP and UDP ports.
- Increased performance through the use of hard-coded signatures rather than regular expressions.
- Ignores traffic once P2P protocol is identified or after seeing 10 packets without a P2P protocol being identified.

**Note**

Because the P2P signatures are hard coded, the only parameters that you can edit are the Master engine parameters.

For More Information

For more information on the parameters common to all signature engines, see [Master Engine, page B-4](#).

Service RPC Engine

The Service RPC engine specializes in RPC protocol and has full decode as an anti-evasive strategy. It can handle fragmented messages (one message in several packets) and batch messages (several messages in a single packet).

The RPC portmapper operates on port 111. Regular RPC messages can be on any port greater than 550. RPC sweeps are like TCP port sweeps, except that they only count unique ports when a valid RPC message is sent. RPC also runs on UDP.

[Table B-27](#) lists the parameters specific to the Service RPC engine.

Table B-27 Service RPC Engine Parameters

Parameter	Description	Value
direction	Specifies the direction of traffic: <ul style="list-style-type: none"> • Traffic from service port destined to client port. • Traffic from client port destined to service port. 	from-service to-service
protocol	Specifies the protocol of interest.	tcp udp
service-ports	Specifies a comma-separated list of ports or port ranges where the target service resides.	0 to 65535 ¹ a-b[,c-d]

Table B-27 Service RPC Engine Parameters (continued)

Parameter	Description	Value
specify-regex-string {yes no}	(Optional) Enables using a regular expression string: <ul style="list-style-type: none"> specify-exact-match-offset—Enables the exact match offset: <ul style="list-style-type: none"> exact-match-offset—Specifies the exact stream offset the regular expression string must report for a match to be valid. specify-min-match-length—Enables the minimum match length: <ul style="list-style-type: none"> min-match-length—Specifies the minimum number of bytes the regular expression string must match. 	0 to 65535
specify-is-spoof-src {yes no}	(Optional) Enables the spoof source address: <ul style="list-style-type: none"> is-spoof-src—Fires an alert when the source address is 127.0.0.1. 	true false
specify-port-map-program {yes no}	(Optional) Enables the portmapper program: <ul style="list-style-type: none"> port-map-program—Specifies the program number sent to the portmapper for this signature. 	0 to 999999999
specify-rpc-max-length {yes no}	(Optional) Enables RPC maximum length: <ul style="list-style-type: none"> rpc-max-length—Specifies the maximum allowed length of the entire RPC message. <p>Note Lengths longer than what you specify fire an alert.</p>	0 to 65535
specify-rpc-procedure {yes no}	(Optional) Enables RPC procedure: <ul style="list-style-type: none"> rpc-procedure—Specifies the RPC procedure number for this signature. 	0 to 1000000
specify-rpc-program {yes no}	(Optional) Enables RPC program: <ul style="list-style-type: none"> rpc-program—Specifies the RPC program number for this signature. 	0 to 1000000
swap-attacker-victim	Swaps the attacker and victim addresses and ports (source and destination) in the alert message and in any actions taken.	true false (default)

1. The second number in the range must be greater than or equal to the first number.

For More Information

- For more information on the parameters common to all signature engines, see [Master Engine, page B-4](#).
- For a list of the signature regular expression syntax, see [Regular Expression Syntax, page B-9](#).

Service SMB Advanced Engine



Note

The SMB engine has been replaced by the SMB Advanced engine. Even though the SMB engine is still visible in IDM, IME, and the CLI, its signatures have been obsoleted; that is, the new signatures have the obsoletes parameter set with the IDs of their corresponding old signatures. Use the new SMB Advanced engine to rewrite any custom signature that were in the SMB engine.

The Service SMB Advanced engine processes Microsoft SMB and Microsoft RPC over SMB packets. The Service SMB Advanced engine uses the same decoding method for connection-oriented MSRPC as the MSRPC engine with the requirement that the MSRPC packet must be over the SMB protocol. The Service SMB Advanced engine supports MSRPC over SMB on TCP ports 139 and 445. It uses a copy of the connection-oriented DCS/RPC code from the MSRPC engine.

Table B-28 lists the parameters specific to the Service SMB Advanced engine.

Table B-28 Service SMB Advanced Engine Parameters

Parameter	Description	Value
service-ports	Specifies a comma-separated list of ports or port ranges where the target service resides.	0 to 65535 a-b[,c-d] ¹
specify-smb-command {yes no}	(Optional) Enables SMB commands: <ul style="list-style-type: none"> smb-command—Specifies the SMB command value. Note Exact match required; defines the SMB packet type. ²	0 to 255
specify-direction {yes no}	(Optional) Enables traffic direction: <ul style="list-style-type: none"> direction—Specifies the direction of traffic: <ul style="list-style-type: none"> from service—Traffic from service port destined to client port. to service—Traffic from client port destined to service port. 	from service to service
specify-msrpc-over-smb-operation {yes no}	(Optional) Enables MSRPC over SMB: <ul style="list-style-type: none"> msrpc-over-smb-operation—Specifies MSRPC over SMB. Note Required for SMB_COM_TRANSACTION commands, exact match required.	0 to 65535
specify-regex-string {yes no}	(Optional) Enables searching for Regex strings: <ul style="list-style-type: none"> regex-string—Specifies a regular expression to search for in a single TCP packet. 	string

Table B-28 Service SMB Advanced Engine Parameters (continued)

Parameter	Description	Value
specify-exact-match-offset {yes no}	(Optional) Enables exact match offset: <ul style="list-style-type: none"> exact-match-offset—Specifies the exact stream offset the Regex string must report for a match to be valid. 	0 to 65535
specify-min-match-length {yes no}	(Optional) Enables minimum match length: <ul style="list-style-type: none"> min-match-length—Specifies the minimum number of bytes the Regex string must match. 	0 to 65535
specify-regex-payload-source {yes no}	(Optional) Enables payload source inspection: <ul style="list-style-type: none"> payload-source—Specifies the kind of payload source inspection.³ 	resource smb-data tcp-data
specify-scan-interval {yes no}	(Optional) Enables scan interval: <ul style="list-style-type: none"> scan-interval—Specifies the interval in seconds used to calculate alert rates. 	1 to 131071
specify-tcp-flags {yes no}	(Optional) Enables TCP flags: <ul style="list-style-type: none"> msrpc-tcp-flags—Specifies the MSRPC TCP flags. msrpc-tcp-flags-mask—Specifies the MSRPC flags mask. 	concurrent-executio n did-not-execute first-fragment last fragment maybe-semantics object-uuid pending-cancel reserved
specify-msrpc-over-smb-pdu-type	(Optional) Enables MSRPC PDU type over the SMB packet: <ul style="list-style-type: none"> msrpc-over-smb-pdu-type—Specifies the PDU type of MSRPC over the SMB packet. 	0 = Request 2 = Response 11 = Bind 12 = Bind Ack
specify-msrpc-over-smb-uuid {yes no}	(Optional) Enables MSRPC over UUID: <ul style="list-style-type: none"> msrpc-over-smb-uuid—Specifies the MSRPC UUID. 	32-character string composed of hexadecimal characters 0-9, a-f, A-F.
swap-attacker-victim	Swaps the attacker and victim addresses and ports (source and destination) in the alert message and in any actions taken.	true false (default)

1. The second number in the range must be greater than or equal to the first number.

2. Currently supporting 37 (0x25) SMB_COM_TRANSACTION command & 162 (0xA2) SMB_COM_NT_CREATE_ANDX command.
3. TCP_Data performs Regex over entire packet, SMB_Data performs Regex on SMB payload only, Resource_DATA performs Regex on SMB_Resource.

For More Information

- For more information on the parameters common to all signature engines, see [Master Engine, page B-4](#).
- For a list of the signature regular expression syntax, see [Regular Expression Syntax, page B-9](#).

Service SNMP Engine

The Service SNMP engine inspects all SNMP packets destined for port 161. You can tune SNMP signatures and create custom SNMP signatures based on specific community names and object identifiers.

Instead of using string comparison or regular expression operations to match the community name and object identifier, all comparisons are made using the integers to speed up the protocol decode and reduce storage requirements.

[Table B-29](#) lists the parameters specific to the Service SNMP engine.

Table B-29 Service SNMP Engine Parameters

Parameter	Description	Value
inspection-type	Enables the SNMP inspection type.	brute-force-inspection (default) invalid-packet-inspection non-snmp-traffic-inspection snmp-inspection
brute-force-inspection	Enables brute force inspection: <ul style="list-style-type: none"> • brute-force-count—Specifies the number of unique SNMP community names that constitute a brute force attempt. 	0 to 65535
invalid-packet-inspection	Inspects for SNMP protocol violations.	—

Table B-29 Service SNMP Engine Parameters (continued)

Parameter	Description	Value
non-snmp-traffic-inspection	Inspects for non-SNMP traffic destined for UDP port 161.	—
snmp-inspection {yes no}	<p>Enables inspection of SNMP traffic:</p> <ul style="list-style-type: none"> • specify-object-id—Enables inspection of the SNMP Object identifier: <ul style="list-style-type: none"> – object-id—Specifies to search for the SNMP object identifier. • specify-community-name—Enables inspection of the SNMP community name: <ul style="list-style-type: none"> – community-name—Specifies to search for the SNMP community name (SNMP password). 	<p><i>object-id</i></p> <p><i>community-name</i></p>

For More Information

For more information on the parameters common to all signature engines, see [Master Engine, page B-4](#).

Service SSH Engine

The Service SSH engine specializes in port 22 SSH traffic. Because all but the setup of an SSH session is encrypted, the Service SSH engine only looks at the fields in the setup. There are two default signatures for SSH. You can tune these signatures, but you cannot create custom signatures.

[Table B-30](#) lists the parameters specific to the Service SSH engine.

Table B-30 Service SSH Engine Parameters

Parameter	Description	Value
length-type	<p>Inspects for one of the following SSH length types:</p> <ul style="list-style-type: none"> • key-length—Enables inspection of the length of the SSH key: <ul style="list-style-type: none"> – length—Specifies that keys larger than this fire the RSAREF overflow. • user-length—Enables user length SSH inspection: <ul style="list-style-type: none"> – length—Specifies that keys larger than this fire the RSAREF overflow. 	0 to 65535

Table B-30 Service SSH Engine Parameters (continued)

Parameter	Description	Value
service-ports	Specifies a comma-separated list of ports or port ranges where the target service resides.	0 to 65535 ¹ a-b[,c-d]
specify-packet-depth {yes no}	(Optional) Enables packet depth: <ul style="list-style-type: none"> packet-depth—Specifies the number of packets to watch before determining the session key was missed. 	0 to 65535

1. The second number in the range must be greater than or equal to the first number.

For More Information

For more information on the parameters common to all signature engines, see [Master Engine, page B-4](#).

Service TNS Engine

The Service TNS engine inspects TNS protocol. TNS provides database applications with a single common interface to all industry-standard network protocols. With TNS, applications can connect to other database applications across networks with different protocols. The default TNS listener port is TCP 1521. TNS also supports REDIRECT frames that redirect the client to another host and/or another TCP port. To support REDIRECT packets, the TNS engine listens on all TCP ports and has a quick TNS frame header validation routine to ignore non-TNS streams.

[Table B-31](#) lists the parameters specific to the Service TNS engine

Table B-31 Service TNS Engine Parameters

Parameter	Description	Value
direction	Specifies the direction of traffic: <ul style="list-style-type: none"> Traffic from service port destined to client port. Traffic from client port destined to service port. 	from-service to-service
specify-regex-string {yes no}	(Optional) Enables using a regular expression string: <ul style="list-style-type: none"> regex-string—Specifies the regular expression to search for. 	string
specify-exact-match-offset {yes no}	Enables the exact match offset: <ul style="list-style-type: none"> exact-match-offset—Specifies the exact stream offset the regex-string must report for a match to be valid. 	0 to 65535
specify-max-match-offset {yes no}	Enables maximum match offset: <ul style="list-style-type: none"> max-match-offset—Specifies the maximum stream offset the regex-string must report for a match to be valid. 	0 to 65535
specify-min-match-offset {yes no}	Enables minimum match offset: <ul style="list-style-type: none"> min-match-offset—Specifies the minimum stream offset the regex-string must report for a match to be valid. 	0 to 65535

Table B-31 Service TNS Engine Parameters (continued)

Parameter	Description	Value
specify-min-match-length {yes no}	Enables the minimum match length: <ul style="list-style-type: none"> – min-match-length—Specifies the minimum number of bytes the regex-string must match. 	0 to 65535
specify-regex-payload-src {yes no}	Enables the inspection of TCP or TNS protocol: <ul style="list-style-type: none"> • payload-src—Specifies which protocol to inspect: <ul style="list-style-type: none"> – tcp-data—Performs Regex over the data portion of the TCP packet. – tns-data—Performs Regex only over the TNS data (with all white space removed). 	tcp data tns data
type	Specifies the TNS frame value type: <ul style="list-style-type: none"> • 1—Connect • 2—Accept • 4—Refuse • 5—Redirect • 6—Data • 11—Resend • 12—Marker 	1 2 4 5 6 11 12

For More Information

- For more information on the parameters common to all signature engines, see [Master Engine, page B-4](#).
- For a list of the signature regular expression syntax, see [Regular Expression Syntax, page B-9](#).

State Engine

The State engine provides state-based regular expression-based pattern inspection of TCP streams. A state engine is a device that stores the state of an event and at a given time can operate on input to transition from one state to another and/or cause an action or output to take place. State machines are used to describe a specific event that causes an output or alarm. There are three state machines in the State engine: SMTP, Cisco Login, and LPR Format String.

Table B-32 lists the parameters specific to the State engine.

Table B-32 State Engine Parameters

Parameter	Description	Value
state-machine	Specifies the state machine grouping.	cisco-login lpr-format-string smtp
cisco-login	Specifies the state machine for Cisco login: <ul style="list-style-type: none"> state-name—Name of the state required before the signature fires an alert: <ul style="list-style-type: none"> Cisco device state Control-C state Password prompt state Start state 	cisco-device control-c pass-prompt start
lpr-format-string	Specifies the state machine to inspect for the LPR format string vulnerability: <ul style="list-style-type: none"> state-name—Name of the state required before the signature fires an alert: <ul style="list-style-type: none"> Abort state to end LPR Format String inspection Format character state State state 	abort format-char start
smtp	Specifies the state machine for the SMTP protocol: <ul style="list-style-type: none"> state-name—Name of the state required before the signature fires an alert: <ul style="list-style-type: none"> Abort state to end LPR Format String inspection Mail body state Mail header state SMTP commands state Start state 	abort mail-body mail-header smtp-commands start
specify-min-match-length {yes no}	(Optional) Enables minimum match length: <ul style="list-style-type: none"> min-match-length—Specifies the minimum number of bytes the regular expression string must match. 	0 to 65535
regex-string	Specifies the regular expression to search for. Note This parameter is protected; you cannot edit it.	<i>string</i>

Table B-32 State Engine Parameters (continued)

Parameter	Description	Value
direction	Specifies the direction of the traffic: <ul style="list-style-type: none"> Traffic from service port destined to client port. Traffic from client port destined to service port. 	from-service to-service
service-ports	Specifies a comma-separated list of ports or port ranges where the target service resides.	0 to 65535 ¹ a-b[,c-d]
swap-attacker-victim	Swaps the attacker and victim addresses and ports (source and destination) in the alert message and in any actions taken.	true false (default)
specify-exact-match-offset {yes no}	(Optional) Enables exact match offset: <ul style="list-style-type: none"> exact-match-offset—Specifies the exact stream offset the regular expression string must report for a match to be valid. 	0 to 65535
specify-max-match-offset {yes no}	(Optional) Enables maximum match offset: <ul style="list-style-type: none"> max-match-offset—Specifies the maximum stream offset the regular expression string must report for a match to be valid. 	0 to 65535
specify-min-match-offset {yes no}	(Optional) Enables minimum match offset: <ul style="list-style-type: none"> min-match-offset—Specifies the minimum stream offset the regular expression string must report for a match to be valid. 	0 to 65535

1. The second number in the range must be greater than or equal to the first number.

For More Information

For more information on the parameters common to all signature engines, see [Master Engine, page B-4](#).

String Engines

The String engine is a generic-based pattern-matching inspection engine for ICMP, TCP, and UDP protocols. The String engine uses a regular expression engine that can combine multiple patterns into a single pattern-matching table allowing for a single search through the data. There are three String engines: String ICMP, String TCP, and String UDP.

Table B-33 lists the parameters specific to the String ICMP engine.

Table B-33 String ICMP Engine Parameters

Parameter	Description	Value
direction	Specifies the direction of the traffic: <ul style="list-style-type: none"> Traffic from service port destined to client port. Traffic from client port destined to service port. 	from-service to-service
icmp-type	Specifies the value of the ICMP header TYPE.	0 to 18 ¹ a-b[,c-d]
regex-string	The Regex pattern to use in the search.	string
specify-exact-match-offset {yes no}	(Optional) Enables exact match offset: <ul style="list-style-type: none"> exact-match-offset—Specifies the exact stream offset the regular expression string must report for a match to be valid. 	0 to 65535
specify-min-match-length {yes no}	(Optional) Enables minimum match length: <ul style="list-style-type: none"> min-match-length—Specifies the minimum number of bytes the regular expression string must match. 	0 to 65535
swap-attacker-victim	Swaps the attacker and victim addresses and ports (source and destination) in the alert message and in any actions taken.	true false (default)

1. The second number in the range must be greater than or equal to the first number.

Table B-34 lists the parameters specific to the String TCP engine.

Table B-34 String TCP Engine

Parameter	Description	Value
direction	Specifies the direction of the traffic: <ul style="list-style-type: none"> Traffic from service port destined to client port. Traffic from client port destined to service port. 	from-service to-service
regex-string	The Regex pattern to use in the search.	string
service-ports	Specifies a comma-separated list of ports or port ranges where the target service resides.	0 to 65535 ¹ a-b[,c-d]
specify-exact-match-offset {yes no}	(Optional) Enables exact match offset: <ul style="list-style-type: none"> exact-match-offset—Specifies the exact stream offset the regular expression string must report for a match to be valid. 	0 to 65535

Table B-34 String TCP Engine (continued)

Parameter	Description	Value
specify-min-match-length {yes no}	(Optional) Enables minimum match length: <ul style="list-style-type: none"> min-match-length—Specifies the minimum number of bytes the regular expression string must match. 	0 to 65535
strip-telnet-options	Strips the Telnet option characters from the data before the pattern is searched. ²	true false
swap-attacker-victim	Swaps the attacker and victim addresses and ports (source and destination) in the alert message and in any actions taken.	true false (default)

1. The second number in the range must be greater than or equal to the first number.
2. This parameter is primarily used as an IPS anti-evasion tool.

Table B-35 lists the parameters specific to the String UDP engine.

Table B-35 String UDP Engine

Parameter	Description	Value
direction	Specifies the direction of the traffic: <ul style="list-style-type: none"> Traffic from service port destined to client port. Traffic from client port destined to service port. 	<ul style="list-style-type: none"> from-service to-service
regex-string	The Regex pattern to use in the search.	string
service-ports	Specifies a comma-separated list of ports or port ranges where the target service resides.	0 to 65535 ¹ a-b[,c-d]
specify-exact-match-offset {yes no}	(Optional) Enables exact match offset: <ul style="list-style-type: none"> exact-match-offset—Specifies the exact stream offset the regular expression string must report for a match to be valid. 	0 to 65535
specify-min-match-length {yes no}	(Optional) Enables minimum match length: <ul style="list-style-type: none"> min-match-length—Specifies the minimum number of bytes the regular expression string must match. 	0 to 65535
swap-attacker-victim	Swaps the attacker and victim addresses and ports (source and destination) in the alert message and in any actions taken.	true false

1. The second number in the range must be greater than or equal to the first number.

For More Information

- For an example custom String engine signature, see [Example String TCP Engine Signature, page 7-47](#).
- For more information on the parameters common to all signature engines, see [Master Engine, page B-4](#).

String XL Engines

**Note**

The IPS 4345, IPS 4360, IPS 4510, IPS 4520, and IPS 4520-XL support the String XL engines and the Regex accelerator card.

The String XL engines do the same thing as the other String engines—provide a matching capability of one string per signature—but they use a different Regex syntax. The String TCP XL engine is stream-based and uses cross-packet inspection (XPI). The packets must be in order. UDP and ICMP are both stateless, thus the String UDP XL and String ICMP XL signature engines require no session state to be allocated and so each packet is a separate search.

The Regex accelerator card is used for both the standard String engines and the String XL engines. Most standard String engine signatures can be compiled and analyzed by the Regex accelerator card without modification. However, there are special circumstances in which the standard String engine signatures cannot be compiled for the Regex accelerator card. In these situations a new signature is written in a String XL engine using the specific parameters in the String XL engine that do compile on the Regex accelerator card. The new signature in the String XL engine obsoletes the original signature in the standard String engine.

Although you can use regular expression syntax or raw expression syntax, raw expression syntax is for expert users only. When configuring String XL signatures, the `regex-string` parameter is required unless you are using raw expression syntax.

**Note**

Raw Regex is regular expression syntax used for raw mode processing. It is expert mode only and targeted for use by the Cisco IPS signature development team or only those who are under supervision by the Cisco IPS signature development team. You can configure a String XL signature in either regular Regex or raw Regex.

[Table B-36](#) lists the parameters specific to the String XL engines (TCP, ICMP, and UDP).

Table B-36 *String XL Engine Parameters*

Parameter	Description	Value
direction	(Required) Direction of the traffic to inspect: <ul style="list-style-type: none"> • Traffic from service port destined to client port. • Traffic from client port destined to service port. 	from-service to-service
dot-all	If set to true, matches <code>[\x00-\xFF]</code> including <code>\n</code> ; if set to false, matches anything in the range <code>[\x00-\xFF]</code> except <code>\n</code> .	true false (default)

Table B-36 String XL Engine Parameters (continued) (continued)

Parameter	Description	Value
end-optional	Specifies that at the end of a packet, if all other conditions are satisfied but the end is not seen, a match is reported if the minimum is exceeded.	true false (default)
icmp-type	Specifies the ICMP message type. Required if the signature engine is string-icmp.	0 to 18 ¹ a-b[,c-d]
no-case	Specifies to treat all alphabetic characters in the expression as case insensitive.	true false (default)
raw-regex	If set to true, min-match-length, max-match-length, min-whole-length, max-whole-length, dot-all, utf8, no-case, stingy, and end-optional are not used to reformat the regular expression string. Note raw-regex lets you enter a regular expression string in Raw syntax without being translated.	true false (default)
regex-string	(Required) Specifies the Regex pattern to use in the search. Note This parameter is required unless max-stream-length is set. Do not set the regex-string if max-stream-length is set.	string
service-ports	(Required) Specifies a comma-separated list of ports or port ranges where the target service resides. Note This parameter is required for the String XL TCP and String XL UDP signature engines. It cannot be used for the String XL ICMP signature engine.	0 to 65535 ¹ a-b[,c-d]
specify-exact-match-offset {yes No}	Enables exact match offset: <ul style="list-style-type: none">exact-match-offset—Specifies the exact stream offset in bytes the regular expression string must report for a match to be valid.	0 to 65535
specify-max-match-offset {yes No}	Enables maximum match offset: <ul style="list-style-type: none">maximum-match-offset—Specifies the maximum stream offset in bytes the regular expression string must report for a match to be valid.	0 to 65535
specify-min-match-offset {yes No}	Enables minimum match offset: <ul style="list-style-type: none">min-match-offset—Specifies the minimum stream offset in bytes the regular expression string must report for a match to be valid.	0 to 65535

Table B-36 String XL Engine Parameters (continued) (continued)

Parameter	Description	Value
specify-max-match-length {yes No}	Enables maximum match length: <ul style="list-style-type: none"> max-match-length—Specifies the maximum number of bytes the regular expression string must match for the pattern to be considered a hit. 	0 to 65535
specify-min-match-length {yes No}	Enables minimum match length: <ul style="list-style-type: none"> min-match-length—Specifies the minimum number of bytes the regular expression string must match for the pattern to be considered a hit. 	0 to 65535
specify-max-stream-length {yes No}	Enables maximum stream length: <ul style="list-style-type: none"> max-stream-length—Limits the search to the first configured number of bytes. The length of the stream is checked again this value. If the stream contains more bytes than this value, an alert is triggered. <p>Note When you specify this parameter, you cannot configure raw-regex or regex-string.</p>	yes no 0 to 65535
specify-max-whole-length {yes No}	Enables maximum whole length: <ul style="list-style-type: none"> max-whole-length—Specifies the maximum length for the pattern that will not be fragmented. 	yes no 0 to 65535
specify-min-whole-length {yes No}	Enables minimum whole length: <ul style="list-style-type: none"> min-whole-length—Specifies the minimum length for the pattern that will not be fragmented. 	yes no 0 to 65535
stingy	Specifies to stop looking for larger matches after the first completed match. <p>Note stingy can only be used with min-match-length; otherwise, it is ignored.</p>	true false (default)
strip-telnet-options	Strips the Telnet option characters from the data before the pattern is searched. ²	true false (default)
swap-attacker-victim	True if address (and ports) source and destination are swapped in the alert message. False for no swap (default).	true false (default)
utf8	Treats all legal UTF-8 byte sequences in the expression as a single character.	true false (default)

1. The second number in the range must be greater than or equal to the first number.

2. This parameter is primarily used as an IPS anti-evasion tool.

Unsupported String XL Parameters

Although you see the end-optional and specify-max-stream-length parameters in the String XL engine, they are disabled. You receive an error message if you try to configure them. For example, here is the error message you receive after you create a signature using specify-max-stream-length and then try to save it:

```
Apply Changes?[yes]: yes
Error: string-xl-tcp 60003.0 : Maximum Stream Length is currently not supported.
Please don't use this option.
```

```
The configuration changes failed validation, no changes were applied.
Would you like to return to edit mode to correct the errors? [yes]:
```

For More Information

- For more information on the parameters common to all signature engines, see [Master Engine, page B-4](#)
- For example String XL engine signatures, see [Example String XL TCP Engine Match Offset Signature, page 7-58](#) and [Example String XL TCP Engine Minimum Match Length Signature, page 7-61](#).

Sweep Engines

This section describes the Sweep engines, and contains the following topics:

- [Sweep Engine, page B-67](#)
- [Sweep Other TCP Engine, page B-69](#)

Sweep Engine

The Sweep engine analyzes traffic between two hosts or from one host to many hosts. You can tune the existing signatures or create custom signatures. The Sweep engine has protocol-specific parameters for ICMP, UDP, and TCP.

The alert conditions of the Sweep engine ultimately depend on the count of the unique parameter. The unique parameter is the threshold number of distinct hosts or ports depending on the type of sweep. The unique parameter triggers the alert when more than the unique number of ports or hosts is seen on the address set within the time period. The processing of unique port and host tracking is called counting.



Caution

Event action filters based on source and destination IP addresses do not function for the Sweep engine, because they do not filter as regular signatures. To filter source and destination IP addresses in sweep alerts, use the source and destination IP address filter parameters in the Sweep engine signatures.

A unique parameter must be specified for all signatures in the Sweep engine. A limit of 2 through 40 (inclusive) is enforced on the sweeps. 2 is the absolute minimum for a sweep, otherwise, it is not a sweep (of one host or port). 40 is a practical maximum that must be enforced so that the sweep does not consume excess memory. More realistic values for unique range between 5 and 15.

TCP sweeps must have a TCP flag and mask specified to determine which sweep inspector slot in which to count the distinct connections. ICMP sweeps must have an ICMP type specified to discriminate among the various types of ICMP packets.

Data Nodes

When an activity related to Sweep engine signatures is seen, the IPS uses a data node to determine when it should stop monitoring for a particular host. The data node contains various persistent counters and variables needed for cross-packet reassembly of streams and for tracking the inspection state on a per-stream/per-source/per-destination basis. The data node containing the sweep determines when the sweep should expire. The data node stops a sweep when the data node has not seen any traffic for x number of seconds (depending on the protocol).

There are several adaptive timeouts for the data nodes. The data node expires after 30 seconds of idle time on the address set after all of the contained objects have been removed. Each contained object has various timeouts, for example, TCP Stream has a one-hour timeout for established connections. Most other objects have a much shorter expiration time, such as 5 or 60 seconds.

Table B-37 lists the parameters specific to the Sweep engine.

Table B-37 Sweep Engine Parameters

Parameter	Description	Value
dst-addr-filter	Specifies the destination IP address to exclude from the sweep counting algorithm.	<A.B.C.D>- <A.B.C.D> [,<A.B.C.D>- <A.B.C.D>]
src-addr-filter	Specifies the source IP address to exclude from the sweep counting algorithm.	<A.B.C.D>- <A.B.C.D> [,<A.B.C.D>- <A.B.C.D>]
protocol	Specifies the protocol of interest for this inspector.	icmp udp tcp
specify-icmp-type {yes no}	(Optional) Enables the ICMP header type: <ul style="list-style-type: none"> icmp-type—Specifies the value of the ICMP header TYPE. 	0 to 255
specify-port-range {yes no}	(Optional) Enables using a port range for inspection: <ul style="list-style-type: none"> port-range—Specifies the UDP port range used in inspection. 	0 to 65535 a-b[,c-d]
fragment-status	Specifies whether fragments are wanted or not: <ul style="list-style-type: none"> Any fragment status Do not inspect fragments Inspect fragments 	any no-fragments want-fragments
inverted-sweep	Uses source port instead of destination port for unique counting.	true false

Table B-37 Sweep Engine Parameters (continued)

Parameter	Description	Value
mask	Specifies the mask used in TCP flags comparison: <ul style="list-style-type: none"> • URG bit • ACK bit • PSH bit • RST bit • SYN bit • FIN bit 	urg ack psh rst syn fin
storage-key	Specifies the type of address key used to store persistent data: <ul style="list-style-type: none"> • Attacker address • Attacker and victim addresses • Attacker address and victim port 	Axxx AxBx Axxb
suppress-reverse	Does not fire when a sweep has fired in the reverse direction on this address set.	true false
swap-attacker-victim	Swaps the attacker and victim addresses and ports (source and destination) in the alert message and in any actions taken.	true false (default)
tcp-flags	Specifies the TCP flags to match when masked by mask: <ul style="list-style-type: none"> • URG bit • ACK bit • PSH bit • RST bit • SYN bit • FIN bit 	urg ack psh rst syn fin
unique	Specifies the threshold number of unique port connections between the two hosts.	0 to 65535

For More Information

For more information on the parameters common to all signature engines, see [Master Engine, page B-4](#).

Sweep Other TCP Engine

The Sweep Other TCP engine analyzes traffic between two hosts looking for abnormal packets typically used to fingerprint a victim. You can tune the existing signatures or create custom signatures.

TCP sweeps must have a TCP flag and mask specified. You can specify multiple entries in the set of TCP flags. And you can specify an optional port range to filter out certain packets.

Table B-38 lists the parameters specific to the Sweep Other TCP engine.

Table B-38 Sweep Other TCP Engine Parameters

Parameter	Description	Value
specify-port-range {yes no}	(Optional) Enables using a port range for inspection: <ul style="list-style-type: none"> port-range—Specifies the UDP port range used in inspection. 	0 to 65535 a-b[,c-d]
set-tcp-flags	Lets you set TCP flags to match. <ul style="list-style-type: none"> tcp-flags—Specifies the TCP flags used in this inspection: <ul style="list-style-type: none"> – URG bit – ACK bit – PSH bit – RST bit – SYN bit – FIN bit 	urg ack psh rst syn fin

For More Information

For more information on the parameters common to all signature engines, see [Master Engine, page B-4](#).

Traffic Anomaly Engine



Note

You can edit or tune anomaly detection signatures but you cannot create custom anomaly detection signatures.

The Traffic Anomaly engine contains nine anomaly detection signatures covering the three protocols (TCP, UDP, and other). Each signature has two subsignatures, one for the scanner and the other for the worm-infected host (or a scanner under worm attack). When anomaly detection discovers an anomaly, it triggers an alert for these signatures. All anomaly detection signatures are enabled by default and the alert severity for each one is set to high.

When a scanner is detected but no histogram anomaly occurred, the scanner signature fires for that attacker (scanner) IP address. If the histogram signature is triggered, the attacker addresses that are doing the scanning each trigger the worm signature (instead of the scanner signature). The alert details state which threshold is being used for the worm detection now that the histogram has been triggered. From that point on, all scanners are detected as worm-infected hosts.

The following anomaly detection event actions are possible:

- produce-alert—Writes the event to the Event Store.
- deny-attacker-inline—Does not transmit this packet and future packets originating from the attacker address for a specified period of time.
- log-attacker-packets—Starts IP logging for packets that contain the attacker address.
- log-pair-packets—Starts IP logging for packets that contain the attacker and victim address pair.

- deny-attacker-service-pair-inline—Blocks the source IP address and the destination port.
- request-snmp-trap—Sends a request to NotificationApp to perform SNMP notification.
- request-block-host—Sends a request to ARC to block this host (the attacker).

Table B-39 lists the anomaly detection worm signatures.

Table B-39 Anomaly Detection Worm Signatures

Signature ID	Subsignature ID	Name	Description
13000	0	Internal TCP Scanner	Identified a single scanner over a TCP protocol in the internal zone.
13000	1	Internal TCP Scanner	Identified a worm attack over a TCP protocol in the internal zone; the TCP histogram threshold was crossed and a scanner over a TCP protocol was identified.
13001	0	Internal UDP Scanner	Identified a single scanner over a UDP protocol in the internal zone.
13001	1	Internal UDP Scanner	Identified a worm attack over a UDP protocol in the internal zone; the UDP histogram threshold was crossed and a scanner over a UDP protocol was identified.
13002	0	Internal Other Scanner	Identified a single scanner over an Other protocol in the internal zone.
13002	1	Internal Other Scanner	Identified a worm attack over an Other protocol in the internal zone; the Other histogram threshold was crossed and a scanner over an Other protocol was identified.
13003	0	External TCP Scanner	Identified a single scanner over a TCP protocol in the external zone.
13003	1	External TCP Scanner	Identified a worm attack over a TCP protocol in the external zone; the TCP histogram threshold was crossed and a scanner over a TCP protocol was identified.
13004	0	External UDP Scanner	Identified a single scanner over a UDP protocol in the external zone.
13004	1	External UDP Scanner	Identified a worm attack over a UDP protocol in the external zone; the UDP histogram threshold was crossed and a scanner over a UDP protocol was identified.
13005	0	External Other Scanner	Identified a single scanner over an Other protocol in the external zone.
13005	1	External Other Scanner	Identified a worm attack over an Other protocol in the external zone; the Other histogram threshold was crossed and a scanner over an Other protocol was identified.
13006	0	Illegal TCP Scanner	Identified a single scanner over a TCP protocol in the illegal zone.

Table B-39 Anomaly Detection Worm Signatures (continued)

Signature ID	Subsignature ID	Name	Description
13006	1	Illegal TCP Scanner	Identified a worm attack over a TCP protocol in the illegal zone; the TCP histogram threshold was crossed and a scanner over a TCP protocol was identified.
13007	0	Illegal UDP Scanner	Identified a single scanner over a UDP protocol in the illegal zone.
13007	1	Illegal UDP Scanner	Identified a worm attack over a UDP protocol in the illegal zone; the UDP histogram threshold was crossed and a scanner over a UDP protocol was identified.
13008	0	Illegal Other Scanner	Identified a single scanner over an Other protocol in the illegal zone.
13008	1	Illegal Other Scanner	Identified a worm attack over an Other protocol in the illegal zone; the Other histogram threshold was crossed and a scanner over an Other protocol was identified.

For More Information

For more information on the parameters common to all signature engines, see [Master Engine, page B-4](#).

Traffic ICMP Engine

The Traffic ICMP engine analyzes nonstandard protocols, such as TFN2K, LOKI, and DDoS. There are only two signatures (based on the LOKI protocol) with user-configurable parameters.

TFN2K is the newer version of the TFN. It is a DDoS agent that is used to control coordinated attacks by infected computers (zombies) to target a single computer (or domain) with bogus traffic floods from hundreds or thousands of unknown attacking hosts. TFN2K sends randomized packet header information, but it has two discriminators that can be used to define signatures. One is whether the L3 checksum is incorrect and the other is whether the character 64 'A' is found at the end of the payload. TFN2K can run on any port and can communicate with ICMP, TCP, UDP, or a combination of these protocols.

LOKI is a type of back door Trojan. When the computer is infected, the malicious code creates an ICMP Tunnel that can be used to send small payload in ICMP replies (which may go straight through a firewall if it is not configured to block ICMP.) The LOKI signatures look for an imbalance of ICMP echo requests to replies and simple ICMP code and payload discriminators.

The DDoS category (excluding TFN2K) targets ICMP-based DDoS agents. The main tools used here are TFN and Stacheldraht. They are similar in operation to TFN2K, but rely on ICMP only and have fixed commands: integers and strings.

Table B-40 lists the parameters specific to the Traffic ICMP engine.

Table B-40 Traffic ICMP Engine Parameters

Parameter	Description	Value
parameter-tunable-sig	Specifies the whether this signature has configurable parameters.	yes no
inspection-type	Specifies the type of inspection to perform: <ul style="list-style-type: none"> Inspects for original LOKI traffic Inspects for modified LOKI traffic 	is-loki is-mod-loki
reply-ratio	Specifies the imbalance of replies to requests. The alert fires when there are this many more replies than requests.	0 to 65535
want-request	Requires an ECHO REQUEST be seen before firing the alert.	true false

For More Information

For more information on the parameters common to all signature engines, see [Master Engine, page B-4](#).

Trojan Engines

The Trojan engines analyze nonstandard protocols, such as BO2K and TFN2K. There are three Trojan engines: Trojan BO2K, TrojanTFN2K, and Trojan UDP.

BO was the original Windows back door Trojan that ran over UDP only. It was soon superseded by BO2K. BO2K supported UDP and TCP both with basic XOR encryption. They have plain BO headers that have certain cross-packet characteristics.

BO2K also has a stealthy TCP module that was designed to encrypt the BO header and make the cross-packet patterns nearly unrecognizable. The UDP modes of BO and BO2K are handled by the Trojan UDP engine. The TCP modes are handled by the Trojan BO2K engine.



Note

There are no specific parameters to the Trojan engines, except for swap-attacker-victim in the Trojan UDP engine.

For More Information

For more information on the parameters common to all signature engines, see [Master Engine, page B-4](#).



Troubleshooting

This appendix contains troubleshooting tips and procedures for sensors and software. It contains the following sections:

- [Cisco Bug Search Tool, page C-1](#)
- [Preventive Maintenance, page C-2](#)
- [Disaster Recovery, page C-6](#)
- [Password Recovery, page C-7](#)
- [Time Sources and the Sensor, page C-15](#)
- [Advantages and Restrictions of Virtualization, page C-17](#)
- [Supported MIBs, page C-18](#)
- [When to Disable Anomaly Detection, page C-18](#)
- [The Analysis Engine Not Responding, page C-19](#)
- [Troubleshooting RADIUS Authentication, page C-20](#)
- [Troubleshooting Global Correlation, page C-20](#)
- [Troubleshooting External Product Interfaces, page C-21](#)
- [Troubleshooting the Appliance, page C-22](#)
- [Troubleshooting the IDM, page C-53](#)
- [Troubleshooting the IME, page C-55](#)
- [Troubleshooting the ASA 5500-X IPS SSP, page C-56](#)
- [Troubleshooting the ASA 5585-X IPS SSP, page C-67](#)
- [Gathering Information, page C-73](#)

Cisco Bug Search Tool

The Cisco Bug Search Tool (BST), the online successor to Bug Toolkit, is designed to improve our customers' effectiveness in network risk management and device troubleshooting.

BST allows partners and customers to search for software bugs based on product, release, and keyword, and aggregates key data such as bug details, product, and version. The service has provision to filter bugs based on credentials to provide external and internal bug views for the search input.

Check out Bug Search Tools & Resources on Cisco.com. For more details on the tool overview and FAC lists, check out the help page, located at this URL:
<http://www.cisco.com/web/applicat/cbsshelp/help.html>.

Preventive Maintenance

This section describes how to perform preventive maintenance for your sensor, and contains the following topics:

- [Understanding Preventive Maintenance, page C-2](#)
- [Creating and Using a Backup Configuration File, page C-2](#)
- [Backing Up and Restoring the Configuration File Using a Remote Server, page C-3](#)
- [Creating the Service Account, page C-5](#)

Understanding Preventive Maintenance

The following actions will help you maintain your sensor:

- Back up a good configuration. If your current configuration becomes unusable, you can replace it with the backup version.
- Save your backup configuration to a remote system.
- Always back up your configuration before you do a manual upgrade. If you have auto upgrades configured, make sure you do periodic backups.
- Create a service account. A service account is needed for special debug situations directed by TAC.



Caution

You should carefully consider whether you want to create a service account. The service account provides shell access to the system, which makes the system vulnerable. Analyze your situation to decide if you want a service account existing on the system.

For More Information

- For the procedure for backing up a configuration file, see [Creating and Using a Backup Configuration File, page C-2](#).
- For the procedure for using a remote server to copy and restore the a configuration file, see [Backing Up and Restoring the Configuration File Using a Remote Server, page C-3](#).
- For more information about the service account, see [Creating the Service Account, page C-5](#).

Creating and Using a Backup Configuration File

To protect your configuration, you can back up the current configuration and then display it to confirm that is the configuration you want to save. If you need to restore this configuration, you can merge the backup configuration file with the current configuration or overwrite the current configuration file with the backup configuration file.

To back up your current configuration, follow these steps:

-
- Step 1** Log in to the CLI using an account with administrator privileges.
- Step 2** Save the current configuration. The current configuration is saved in a backup file.
- ```
sensor# copy current-config backup-config
```
- Step 3** Display the backup configuration file. The backup configuration file is displayed.
- ```
sensor# more backup-config
```
- Step 4** You can either merge the backup configuration with the current configuration, or you can overwrite the current configuration:
- Merge the backup configuration into the current configuration.
- ```
sensor# copy backup-config current-config
```
- Overwrite the current configuration with the backup configuration.
- ```
sensor# copy /erase backup-config current-config
```
-

Backing Up and Restoring the Configuration File Using a Remote Server



Note

We recommend copying the current configuration file to a remote server before upgrading.

Use the **copy [/erase] source_url destination_url keyword** command to copy the configuration file to a remote server. You can then restore the current configuration from the remote server. You are prompted to back up the current configuration first.

The following commands apply:

- **/erase**—Erases the destination file before copying.

This keyword only applies to the current-config; the backup-config is always overwritten. If this keyword is specified for destination current-config, the source configuration is applied to the system default configuration. If it is not specified for the destination current-config, the source configuration is merged with the current-config.
- **source_url**—The location of the source file to be copied. It can be a URL or keyword.
- **destination_url**—The location of the destination file to be copied. It can be a URL or a keyword.
- **current-config**—The current running configuration. The configuration becomes persistent as the commands are entered.
- **backup-config**—The storage location for the configuration backup.

The exact format of the source and destination URLs varies according to the file. Here are the valid types:

- **ftp:**—Source or destination URL for an FTP network server. The syntax for this prefix is:


```
ftp://[username@]location[/relativeDirectory]/filename
```

```
ftp://[username@]location[/absoluteDirectory]/filename
```



Note You are prompted for a password.

- scp:—Source or destination URL for the SCP network server. The syntax for this prefix is:
 scp://[[username@]location][relativeDirectory]/filename
 scp://[[username@]location][absoluteDirectory]/filename



Note You are prompted for a password. You must add the remote host to the SSH known hosts list.

- http:—Source URL for the web server. The syntax for this prefix is:
 http://[[username@]location][directory]/filename



Note The directory specification should be an absolute path to the desired file.

- https:—Source URL for the web server. The syntax for this prefix is:
 https://[[username@]location][directory]/filename



Note The directory specification should be an absolute path to the desired file. The remote host must be a TLS trusted host.

**Caution**

Copying a configuration file from another sensor may result in errors if the sensing interfaces and virtual sensors are not configured the same.

Backing Up the Current Configuration to a Remote Server

To back up your current configuration to a remote server, follow these steps:

Step 1 Log in to the CLI using an account with administrator privileges.

Step 2 Back up the current configuration to the remote server.

```
sensor# copy current-config scp://user@192.0.2.0//configuration/cfg current-config
Password: *****
Warning: Copying over the current configuration may leave the box in an unstable state.
Would you like to copy current-config to backup-config before proceeding? [yes]:
```

Step 3 Enter **yes** to copy the current configuration to a backup configuration.

```
cfg          100% |*****| 36124          00:00
```

Restoring the Current Configuration From a Backup File

To restore your current configuration from a backup file, follow these steps:

Step 1 Log in to the CLI using an account with administrator privileges.

Step 2 Back up the current configuration to the remote server.

```
sensor# copy scp://user@192.0.2.0//configuration/cfg current-config
Password: *****
Warning: Copying over the current configuration may leave the box in an unstable state.
Would you like to copy current-config to backup-config before proceeding? [yes]:
```

Step 3 Enter **yes** to copy the current configuration to a backup configuration.

```
cfg          100% |*****| 36124          00:00

Warning: Replacing existing network-settings may leave the box in an unstable state.
Would you like to replace existing network settings
(host-ipaddress/netmask/gateway/access-list) on sensor before proceeding? [no]:
sensor#
```

Step 4 Enter **no** to retain the currently configured hostname, IP address, subnet mask, management interface, and access list. We recommend you retain this information to preserve access to your sensor after the rest of the configuration has been restored.

For More Information

For a list of supported HTTP/HTTPS servers, see [Supported FTP and HTTP/HTTPS Servers, page 21-3](#).

Creating the Service Account

You can create a service account for TAC to use during troubleshooting. Although more than one user can have access to the sensor, only one user can have service privileges on a sensor. The service account is for support purposes only.

The root user password is synchronized to the service account password when the service account is created. To gain root access you must log in with the service account and switch to user root with the **su - root** command.



Caution

Do not make modifications to the sensor through the service account except under the direction of TAC. If you use the service account to configure the sensor, your configuration is not supported by TAC. Adding services to the operating system through the service account affects proper performance and functioning of the other IPS services. TAC does not support a sensor on which additional services have been added.



Caution

You should carefully consider whether you want to create a service account. The service account provides shell access to the system, which makes the system vulnerable. However, you can use the service account to create a password if the administrator password is lost. Analyze your situation to decide if you want a service account existing on the system.

**Note**

For IPS 5.0 and later, you can no longer remove the **cisco** account. You can disable it using the **no password cisco** command, but you cannot remove it. To use the **no password cisco** command, there must be another administrator account on the sensor. Removing the **cisco** account through the service account is not supported. If you remove the **cisco** account through the service account, the sensor most likely will not boot up, so to recover the sensor you must reinstall the sensor system image.

To create the service account, follow these steps:

Step 1 Log in to the CLI using an account with administrator privileges.

Step 2 Enter configuration mode.

```
sensor# configure terminal
```

Step 3 Specify the parameters for the service account. The username follows the pattern `^[A-Za-z0-9()+:./-]+`, which means the username must start with a letter or number, and can include any letter A to Z (capital or small), any number 0 to 9, - and _, and can contain 1 to 64 characters.

```
sensor(config)# user username privilege service
```

Step 4 Specify a password when prompted. A valid password is 8 to 32 characters long. All characters except space are allowed. If a service account already exists for this sensor, the following error is displayed and no service account is created.

```
Error: Only one service account may exist
```

Step 5 Exit configuration mode.

```
sensor(config)# exit  
sensor#
```

When you use the service account to log in to the CLI, you receive this warning.

```
***** WARNING *****  
UNAUTHORIZED ACCESS TO THIS NETWORK DEVICE IS PROHIBITED. This account is intended to be  
used for support and troubleshooting purposes only. Unauthorized modifications are not  
supported and will require this device to be reimaged to guarantee proper operation.  
*****
```

Disaster Recovery

Follow these recommendations so that you are ready in case of a disaster:

- If you are using the CLI, IDM, or IME for configuration, copy the current configuration from the sensor to an FTP or SCP server any time a change has been made.
- You should note the specific software version for that configuration. You can apply the copied configuration only to a sensor of the same version.
- You also need the list of user IDs that have been used on that sensor. The list of user IDs and passwords are not saved in the configuration.

When a disaster happens and you need to recover the sensor, try the following:

1. Reimage the sensor.

2. Log in to the sensor with the default user ID and password—**cisco**.



Note You are prompted to change the **cisco** password.

3. Initialize the sensor.
4. Upgrade the sensor to the IPS software version it had when the configuration was last saved and copied.



Warning

Trying to copy the saved configuration without getting the sensor back to the same IPS software version it had before the disaster can cause configuration errors.

5. Copy the last saved configuration to the sensor.
6. Update clients to use the new key and certificate of the sensor. Reimaging changes the sensor SSH keys and HTTPS certificate, so you must add the hosts back to the SSN known hosts list.
7. Create previous users.

For More Information

- For the procedure for backing up a configuration file, see [Creating and Using a Backup Configuration File, page C-2](#).
- For the procedure for obtaining a list of the current users on the sensor, see [Showing User Status, page 4-32](#).
- For the procedures for reimage a sensor, see [Chapter 21, “Upgrading, Downgrading, and Installing System Images.”](#)
- For the procedure for using the **setup** command to initialize the sensor, see [Chapter 3, “Initializing the Sensor.”](#)
- For more information on obtaining IPS software and how to install it, see [Obtaining Cisco IPS Software, page 20-1](#).
- For the procedure for using a remote server to copy and restore the a configuration file, see [Backing Up and Restoring the Configuration File Using a Remote Server, page C-3](#).
- For the procedure for adding hosts to the SSH known hosts list, see [Adding Hosts to the SSH Known Hosts List, page 4-47](#).
- For the procedure for adding users, see [Adding and Removing Users, page 4-18](#).

Password Recovery

For most IPS platforms, you can now recover the password on the sensor rather than using the service account or reimaging the sensor. This section describes how to recover the password for the various IPS platforms. It contains the following topics:

- [Understanding Password Recovery, page C-8](#)
- [Recovering the Password for the Appliance, page C-8](#)
- [Recovering the Password for the ASA 5500-X IPS SSP, page C-10](#)
- [Recovering the Password for the ASA 5585-X IPS SSP, page C-12](#)

- [Disabling Password Recovery, page C-13](#)
- [Verifying the State of Password Recovery, page C-14](#)
- [Troubleshooting Password Recovery, page C-15](#)

Understanding Password Recovery



Note

Administrators may need to disable the password recovery feature for security reasons.

Password recovery implementations vary according to IPS platform requirements. Password recovery is implemented only for the cisco administrative account and is enabled by default. The IPS administrator can then recover user passwords for other accounts using the CLI. The cisco user password reverts to **cisco** and must be changed after the next login.

[Table C-1](#) lists the password recovery methods according to platform.

Table C-1 Password Recovery Methods According to Platform

Platform	Description	Recovery Method
4300 series sensors 4500 series sensors	Standalone IPS appliances	GRUB prompt or ROMMON
ASA 5500-X IPS SSP ASA 5585-X IPS SSP	ASA 5500 series adaptive security appliance IPS modules	Adaptive security appliance CLI command

Recovering the Password for the Appliance

This section describes the two ways to recover the password for appliances. It contains the following topics:

- [Using the GRUB Menu, page C-8](#)
- [Using ROMMON, page C-9](#)

Using the GRUB Menu



Note

You must have a terminal server or direct serial connection to the appliance to use the GRUB menu to recover the password.

For the IPS 4355, IPS 4360, IPS 4510, IPS 4520, and IPS 4520-XL appliances, the password recovery is found in the GRUB menu, which appears during bootup. When the GRUB menu appears, press any key to pause the boot process.

To recover the password on appliances, follow these steps:

Step 1 Reboot the appliance to see the GRUB menu.

```
GNU GRUB version 0.94 (632K lower / 523264K upper memory)
-----
0: Cisco IPS
```

```

1: Cisco IPS Recovery
2: Cisco IPS Clear Password (cisco)
-----

```

Use the ^ and v keys to select which entry is highlighted.
Press enter to boot the selected OS, 'e' to edit the
Commands before booting, or 'c' for a command-line.

Highlighted entry is 0:

Step 2 Press any key to pause the boot process.

Step 3 Choose **2: Cisco IPS Clear Password (cisco)**. The password is reset to **cisco**. Log in to the CLI with username **cisco** and password **cisco**. You can then change the password.

Using ROMMON

For the IPS 4345, IPS 4360, IPS 4510, IPS 4520 and IPS 4520-XL, you can use the ROMMON to recover the password. To access the ROMMON CLI, reboot the sensor from a terminal server or direct connection and interrupt the boot process.



Note

After recovering the password, you must reset the confreg to **0**, otherwise, when you try to upgrade the sensor, the upgrade fails because when the sensor reboots, it goes to password recovery (**confreg 0x7**) rather than to the upgrade option.

To recover the password using the ROMMON CLI, follow these steps:

Step 1 Reboot the appliance.

Step 2 To interrupt the boot process, press **ESC** or **Control-R** (terminal server) or send a **BREAK** command (direct connection). The boot code either pauses for 10 seconds or displays something similar to one of the following:

- Evaluating boot options
- Use **BREAK** or **ESC** to interrupt boot

Step 3 Enter the following commands to reset the password:

```

confreg 0x7
boot

```

Sample ROMMON session:

```

Booting system, please wait...
CISCO SYSTEMS
Embedded BIOS Version 1.0(11)2 01/25/06 13:21:26.17
...
Evaluating BIOS Options...
Launch BIOS Extension to setup ROMMON
Cisco Systems ROMMON Version (1.0(11)2) #0: Thu Jan 26 10:43:08 PST 2006
Platform IPS-4360-K9
Use BREAK or ESC to interrupt boot.
Use SPACE to begin boot immediately.
Boot interrupted.
Management0/0
Link is UP
MAC Address:000b.fcfa.d155

```

```
Use ? for help.
rommon #0> confreg 0x7
Update Config Register (0x7) in NVRAM...
rommon #1> boot
```

Step 4 Enter the following command to reset the confreg value to 0:

```
confreg 0
```

Recovering the Password for the ASA 5500-X IPS SSP

You can reset the password to the default (**cisco**) for the ASA 5500-X IPS SSP using the CLI or the ASDM. Resetting the password causes it to reboot. IPS services are not available during a reboot.



Note

To reset the password, you must have ASA 8.6.1 or later.

Use the **sw-module module ips password-reset** command to reset the password to the default **cisco**. If the module in the specified slot has an IPS version that does not support password recovery, the following error message is displayed:

```
ERROR: the module in slot <n> does not support password recovery.
```

To reset the password on the ASA 5500-X IPS SSP, follow these steps:

Step 1 Log into the adaptive security appliance and enter the following command:

```
asa# sw-module module ips password-reset
Reset the password on module ips? [confirm]
```

Step 2 Press **Enter** to confirm.

```
Password-Reset issued for module ips.
```

Step 3 Verify the status of the module. Once the status reads **Up**, you can session to the ASA 5500-X IPS SSP.

```
asa# show module ips
Mod Card Type                               Model                               Serial No.
-----
ips ASA 5555-X IPS Security Services Processor ASA5555-IPS FCH151070GR

Mod MAC Address Range                       Hw Version   Fw Version   Sw Version
-----
ips 503d.e59c.7c4c to 503d.e59c.7c4c N/A          N/A          7.1(4)E4

Mod SSM Application Name                    Status        SSM Application Version
-----
ips IPS                                     Up            7.1(4)E4

Mod Status      Data Plane Status   Compatibility
-----
ips Up          Up

Mod License Name   License Status   Time Remaining
-----
ips IPS Module    Enabled          210 days
```

Step 4 Session to the ASA 5500-X IPS SSP.

```
asa# session ips
Opening command session with module ips.
Connected to module ips. Escape character sequence is 'CTRL-^X'.
```

Step 5 Enter the default username (**cisco**) and password (**cisco**) at the login prompt.

```
login: cisco
Password: cisco
```

```
You are required to change your password immediately (password aged)
Changing password for cisco.
(current) password: cisco
```

Step 6 Enter your new password twice.

```
New password: new password
Retype new password: new password
```

NOTICE

This product contains cryptographic features and is subject to United States and local country laws governing import, export, transfer and use. Delivery of Cisco cryptographic products does not imply third-party authority to import, export, distribute or use encryption. Importers, exporters, distributors and users are responsible for compliance with U.S. and local country laws. By using this product you agree to comply with applicable laws and regulations. If you are unable to comply with U.S. and local laws, return this product immediately.

A summary of U.S. laws governing Cisco cryptographic products may be found at:
<http://www.cisco.com/wwl/export/crypto/tool/stqrg.html>

If you require further assistance please contact us by sending email to export@cisco.com.

LICENSE NOTICE

There is no license key installed on this IPS platform. The system will continue to operate with the currently installed signature set. A valid license must be obtained in order to apply signature updates. Please go to <http://www.cisco.com/go/license> to obtain a new license or install a license.

```
asa-ssp#
```

Using the ASDM

To reset the password in the ASDM, follow these steps:

Step 1 From the ASDM menu bar, choose **Tools > IPS Password Reset**.



Note This option does not appear in the menu if there is no IPS present.

Step 2 In the IPS Password Reset confirmation dialog box, click **OK** to reset the password to the default (**cisco**). A dialog box displays the success or failure of the password reset. If the reset fails, make sure you have the correct ASA and IPS software versions.

Step 3 Click **Close** to close the dialog box. The sensor reboots.

Recovering the Password for the ASA 5585-X IPS SSP



Note

To reset the password, you must have ASA 8.2.(4.4) or later or ASA 8.4.2 or later. The ASA 5585-X IPS SSP is not supported in ASA 8.3(x).

You can reset the password to the default (**cisco**) for the ASA 5585-X IPS SSP using the CLI or the ASDM. Resetting the password causes it to reboot. IPS services are not available during a reboot.

Use the **hw-module module slot_number password-reset** command to reset the password to the default **cisco**. If the module in the specified slot has an IPS version that does not support password recovery, the following error message is displayed:

```
ERROR: the module in slot <n> does not support password recovery.
```

To reset the password on the ASA 5585-X IPS SSP, follow these steps:

Step 1 Log into the adaptive security appliance and enter the following command:

```
asa# hw-module module 1 password-reset
Reset the password on module in slot 1? [confirm]
```

Step 2 Press **Enter** to confirm.

```
Password-Reset issued for slot 1.
```

Step 3 Verify the status of the module. Once the status reads **Up**, you can session to the ASA 5585-X IPS SSP.

```
asa# show module 1
Mod Card Type                               Model                               Serial No.
-----
  1 ASA 5585-X IPS Security Services Processor-4 ASA5585-SSP-IPS40 JAF1436ABSG

Mod MAC Address Range                       Hw Version   Fw Version   Sw Version
-----
  1 5475.d029.8c74 to 5475.d029.8c7f 0.1          2.0(12)3     7.1(4)E4

Mod SSM Application Name                     Status       SSM Application Version
-----
  1 IPS                                       Up          7.1(4)E4

Mod Status           Data Plane Status   Compatibility
-----
  1 Up                Up
```

Step 4 Session to the ASA 5585-X IPS SSP.

```
asa# session 1
Opening command session with slot 1.
Connected to slot 1. Escape character sequence is 'CTRL-^X'.
```

Step 5 Enter the default username (**cisco**) and password (**cisco**) at the login prompt.

```
login: cisco
Password: cisco
```

```
You are required to change your password immediately (password aged)
Changing password for cisco.
(current) password: cisco
```

Step 6 Enter your new password twice.

New password: **new password**
 Retype new password: **new password**

NOTICE

This product contains cryptographic features and is subject to United States and local country laws governing import, export, transfer and use. Delivery of Cisco cryptographic products does not imply third-party authority to import, export, distribute or use encryption. Importers, exporters, distributors and users are responsible for compliance with U.S. and local country laws. By using this product you agree to comply with applicable laws and regulations. If you are unable to comply with U.S. and local laws, return this product immediately.

A summary of U.S. laws governing Cisco cryptographic products may be found at:
<http://www.cisco.com/wwl/export/crypto/tool/stqrg.html>

If you require further assistance please contact us by sending email to export@cisco.com.

LICENSE NOTICE

There is no license key installed on this IPS platform. The system will continue to operate with the currently installed signature set. A valid license must be obtained in order to apply signature updates. Please go to <http://www.cisco.com/go/license> to obtain a new license or install a license.

ips_ssp#

Using the ASDM

To reset the password in the ASDM, follow these steps:

Step 1 From the ASDM menu bar, choose **Tools > IPS Password Reset**.



Note This option does not appear in the menu if there is no IPS present.

Step 2 In the IPS Password Reset confirmation dialog box, click **OK** to reset the password to the default (**cisco**). A dialog box displays the success or failure of the password reset. If the reset fails, make sure you have the correct ASA and IPS software versions.

Step 3 Click **Close** to close the dialog box. The sensor reboots.

Disabling Password Recovery



Caution

If you try to recover the password on a sensor on which password recovery is disabled, the process proceeds with no errors or warnings; however, the password is not reset. If you cannot log in to the sensor because you have forgotten the password, and password recovery is set to disabled, you must reimaged your sensor.

Password recovery is enabled by default. You can disable password recovery through the CLI, IDM, or IME.

Disabling Password Recovery Using the CLI

To disable password recovery in the CLI, follow these steps:

Step 1 Log in to the CLI using an account with administrator privileges.

Step 2 Enter global configuration mode.

```
sensor# configure terminal
```

Step 3 Enter host mode.

```
sensor(config)# service host
```

Step 4 Disable password recovery.

```
sensor(config-hos)# password-recovery disallowed
```

Disabling Password Recovery Using the IDM or IME

To disable password recovery in the IDM or IME, follow these steps:

Step 1 Log in to the IDM or IME using an account with administrator privileges.

Step 2 Choose **Configuration > sensor_name > Sensor Setup > Network**.

Step 3 To disable password recovery, uncheck the **Allow Password Recovery** check box.

Verifying the State of Password Recovery

Use the **show settings | include password** command to verify whether password recovery is enabled.

To verify whether password recovery is enabled, follow these steps:

Step 1 Log in to the CLI.

Step 2 Enter service host submode.

```
sensor# configure terminal
sensor (config)# service host
sensor (config-hos)#
```

Step 3 Verify the state of password recovery by using the **include** keyword to show settings in a filtered output.

```
sensor(config-hos)# show settings | include password
password-recovery: allowed <defaulted>
sensor(config-hos)#
```

Troubleshooting Password Recovery

When you troubleshoot password recovery, pay attention to the following:

- You cannot determine whether password recovery has been disabled in the sensor configuration from the ROMMON prompt, GRUB menu, switch CLI, or router CLI. If you attempt password recovery, it always appears to succeed. If it has been disabled, the password is not reset to **cisco**. The only option is to reimage the sensor.
- You can disable password recovery in the host configuration. For the platforms that use external mechanisms or ROMMON, although you can run commands to clear the password, if password recovery is disabled in the IPS, the IPS detects that password recovery is not allowed and rejects the external request.
- To check the state of password recovery, use the **show settings | include password** command.

Time Sources and the Sensor

This section describes how to maintain accurate time on the sensor, and contains the following topics:

- [Time Sources and the Sensor, page C-15](#)
- [Synchronizing IPS Clocks with Parent Device Clocks, page C-16](#)
- [Verifying the Sensor is Synchronized with the NTP Server, page C-16](#)
- [Correcting Time on the Sensor, page C-17](#)

Time Sources and the Sensor



Note

We recommend that you use an NTP server to regulate time on your sensor. You can use authenticated or unauthenticated NTP. For authenticated NTP, you must obtain the NTP server IP address, NTP server key ID, and the key value from the NTP server. You can set up NTP during initialization or you can configure NTP through the CLI, IDM, or IME, or ASDM.

The sensor requires a reliable time source. All events (alerts) must have the correct UTC and local time stamp, otherwise, you cannot correctly analyze the logs after an attack. When you initialize the sensor, you set up the time zones and summertime settings. This section provides a summary of the various ways to set the time on sensors.

The IPS Standalone Appliances

- Use the **clock set** command to set the time. This is the default.
- Configure the appliance to get its time from an NTP time synchronization source.



Note

The currently supported Cisco IPS appliances are the IPS 4345, IPS 4360, IPS 4510, IPS 4520, and IPS 4520-XL

The ASA IPS Modules

- The ASA 5500-X IPS SSP and ASA 5585-X IPS SSP automatically synchronize their clocks with the clock in the adaptive security appliance in which they are installed. This is the default.

- Configure them to get their time from an NTP time synchronization source, such as a Cisco router other than the parent router.

For More Information

For the procedure for configuring NTP, see [Configuring NTP, page 4-43](#).

Synchronizing IPS Clocks with Parent Device Clocks

The ASA IPS modules (ASA 5500-X IPS SSP and ASA 5585-X IPS SSP) synchronize their clocks to the parent chassis clock (adaptive security appliance) each time the IPS boots up and any time the parent chassis clock is set. The IPS clock and parent chassis clock tend to drift apart over time. The difference can be as much as several seconds per day. To avoid this problem, make sure that both the IPS clock and the parent clock are synchronized to an external NTP server. If only the IPS clock or only the parent chassis clock is synchronized to an NTP server, the time drift occurs.

Verifying the Sensor is Synchronized with the NTP Server

In IPS, you cannot apply an incorrect NTP configuration, such as an invalid NTP key value or ID, to the sensor. If you try to apply an incorrect configuration, you receive an error message. To verify the NTP configuration, use the **show statistics host** command to gather sensor statistics. The NTP statistics section provides NTP statistics including feedback on sensor synchronization with the NTP server.

To verify the NTP configuration, follow these steps:

Step 1 Log in to the sensor.

Step 2 Generate the host statistics.

```
sensor# show statistics host
...
NTP Statistics
  remote          refid          st t when poll reach  delay  offset  jitter
11.22.33.44      CHU_AUDIO(1)   8 u  36  64   1   0.536  0.069  0.001
LOCAL(0)         73.78.73.84    5 l  35  64   1   0.000  0.000  0.001
ind assID status  conf reach auth  condition  last_event cnt
  1 10372 f014  yes  yes  ok    reject    reachable  1
  2 10373 9014  yes  yes  none  reject    reachable  1
status = Not Synchronized
...
```

Step 3 Generate the hosts statistics again after a few minutes.

```
sensor# show statistics host
...
NTP Statistics
  remote          refid          st t when poll reach  delay  offset  jitter
*11.22.33.44     CHU_AUDIO(1)   8 u  22  64  377  0.518  37.975  33.465
LOCAL(0)         73.78.73.84    5 l  22  64  377  0.000  0.000  0.001
ind assID status  conf reach auth  condition  last_event cnt
  1 10372 f624  yes  yes  ok    sys.peer  reachable  2
  2 10373 9024  yes  yes  none  reject    reachable  2
status = Synchronized
```

Step 4 If the status continues to read `Not Synchronized`, check with the NTP server administrator to make sure the NTP server is configured correctly.

Correcting Time on the Sensor

If you set the time incorrectly, your stored events will have the incorrect time because they are stamped with the time the event was created. The Event Store time stamp is always based on UTC time. If during the original sensor setup, you set the time incorrectly by specifying 8:00 p.m. rather than 8:00 a.m., when you do correct the error, the corrected time will be set backwards. New events might have times older than old events.

For example, if during the initial setup, you configure the sensor as central time with daylight saving time enabled and the local time is 8:04 p.m., the time is displayed as 20:04:37 CDT and has an offset from UTC of -5 hours (01:04:37 UTC, the next day). A week later at 9:00 a.m., you discover the error: the clock shows 21:00:23 CDT. You then change the time to 9:00 a.m. and now the clock shows 09:01:33 CDT. Because the offset from UTC has not changed, it requires that the UTC time now be 14:01:33 UTC, which creates the time stamp problem.

To ensure the integrity of the time stamp on the event records, you must clear the event archive of the older events by using the **clear events** command.

**Note**

You cannot remove individual events.

For More Information

For the procedure for clearing events, see [Clearing Events, page C-102](#).

Advantages and Restrictions of Virtualization

To avoid configuration problems on your sensor, make sure you understand the advantages and restrictions of virtualization on your sensor.

Virtualization has the following advantages:

- You can apply different configurations to different sets of traffic.
- You can monitor two networks with overlapping IP spaces with one sensor.
- You can monitor both inside and outside of a firewall or NAT device.

Virtualization has the following restrictions:

- You must assign both sides of asymmetric traffic to the same virtual sensor.
- Using VACL capture or SPAN (promiscuous monitoring) is inconsistent with regard to VLAN tagging, which causes problems with VLAN groups.
 - When using Cisco IOS software, a VACL capture port or a SPAN target does not always receive tagged packets even if it is configured for trunking.
 - When using the MSFC, fast path switching of learned routes changes the behavior of VACL captures and SPAN.
- Persistent store is limited.

Virtualization has the following traffic capture requirements:

- The virtual sensor must receive traffic that has 802.1q headers (other than traffic on the native VLAN of the capture port).
- The sensor must see both directions of traffic in the same VLAN group in the same virtual sensor for any given sensor.

The following sensors support virtualization:

- ASA 5500-X IPS SSP
- ASA 5585-X IPS SSP
- IPS 4345
- IPS 4345-DC
- IPS 4360
- IPS 4510
- IPS 4520
- IPS 4520-XL

Supported MIBs

To avoid problems with configuring SNMP, be aware of the MIBs that are supported on the sensor.

The following private MIBs are supported on the sensor:

- CISCO-CIDS-MIB
The CISCO-CIDS-MIB has been updated to include SNMP health data.
- CISCO-ENHANCED-MEMPOOL-MIB
- CISCO-ENTITY-ALARM-MIB

You can obtain these private Cisco MIBs under the heading SNMP v2 MIBs at this URL:

<http://www.cisco.com/public/sw-center/netmgmt/cmtk/mibs.shtml>

**Note**

MIB II is available on the sensor, but we do not support it. We know that some elements are not correct (for example, the packet counts from the IF MIB on the sensing interfaces). While you can use elements from MIB II, we do not guarantee that they all provide correct information. We fully support the other listed MIBs and their output is correct.

**Note**

CISCO-PROCESS-MIB is available on the sensor, but we do not support it. We know that some elements are not available. While you can use elements from CISCO-PROCESS-MIB, we do not guarantee that they all provide correct information. We fully support the other listed MIBs and their output is correct.

When to Disable Anomaly Detection

If you have anomaly detection enabled and you have your sensor configured to see only one direction of traffic, you should disable anomaly detection. Otherwise, you will receive many alerts, because anomaly detection sees asymmetric traffic as having incomplete connections, that is, like worm scanners, and fires alerts.

To disable anomaly detection, follow these steps:

-
- Step 1** Log in to the CLI using an account with administrator privileges.
- Step 2** Enter analysis engine submode.
- ```
sensor# configure terminal
sensor(config)# service analysis-engine
sensor(config-ana)#
```
- Step 3** Enter the virtual sensor name that contains the anomaly detection policy you want to disable.
- ```
sensor(config-ana)# virtual-sensor vs0
sensor(config-ana-vir)#
```
- Step 4** Disable anomaly detection operational mode.
- ```
sensor(config-ana-vir)# anomaly-detection
sensor(config-ana-vir-ano)# operational-mode inactive
sensor(config-ana-vir-ano)#
```
- Step 5** Exit analysis engine submode.
- ```
sensor(config-ana-vir-ano)# exit
sensor(config-ana-vir)# exit
sensor(config-ana-)# exit
Apply Changes:[yes]:
```
- Step 6** Press **Enter** to apply your changes or enter **no** to discard them.
-

For More Information

To learn more about Worms, see [Understanding Worms, page 9-2](#).

The Analysis Engine Not Responding

Error Message Output from `show statistics analysis-engine`
Error: getAnalysisEngineStatistics : ct-sensorApp.424 not responding, please check system processes - The connect to the specified Io::ClientPipe failed.

Error Message Output from `show statistics anomaly-detection`
Error: getAnomalyDetectionStatistics : ct-sensorApp.424 not responding, please check system processes - The connect to the specified Io::ClientPipe failed.

Error Message Output from `show statistics denied-attackers`
Error: getDeniedAttackersStatistics : ct-sensorApp.424 not responding, please check system processes - The connect to the specified Io::ClientPipe failed.

Possible Cause These error messages appear when you run the **show tech support** command and the Analysis Engine is not running.

Recommended Action Verify the Analysis Engine is running and monitor it to see if the issue is resolved.

To verify the Analysis Engine is running and to monitor the issue, follow these steps:

-
- Step 1** Log in to the sensor.
- Step 2** Verify that the Analysis Engine is not running, Check to see if the Analysis Engine reads `Not Running`.
- ```

sensor# show version

MainApp C-2013_12_16_14_00_7_3_0_143 (Release) 2013-12-16T14:06:20-0600
Running
AnalysisEngine C-2013_12_16_14_00_7_3_0_143 (Release) 2013-12-16T14:06:20-0600
Not Running
CollaborationApp C-2013_12_16_14_00_7_3_0_143 (Release) 2013-12-16T14:06:20-0600
Running
CLI C-2013_12_16_14_00_7_3_0_143 (Release) 2013-12-16T14:06:20-0600

```
- Step 3** Enter `show tech-support` and save the output.
- Step 4** Reboot the sensor.
- Step 5** Enter `show version` after the sensor has stabilized to see if the issue is resolved.
- Step 6** If the Analysis Engine still reads `Not Running`, contact TAC with the original `show tech support` command output.
- 

## Troubleshooting RADIUS Authentication

**Symptom** Attempt limit configured on the IPS sensor may not be enforced for a RADIUS user.

**Conditions** Applicable for RADIUS users only. The RADIUS user must have logged in to the sensor at least once after RADIUS authentication is enabled or after the sensor is reset or rebooted.

**Workaround** Log in to the sensor with the correct credentials and from that time on the attempt limit is enforced for that RADIUS user.

### For More Information

For detailed information about RADIUS authentication, see [Configuring Authentication and User Parameters](#), page 4-18.

## Troubleshooting Global Correlation

Make sure you observe the following when configuring global correlation:

- Because global correlation updates occur through the sensor management interface, firewalls must allow port 443/80 traffic.
- You must have an HTTP proxy server or a DNS server configured to allow global correlation features to function.



- If you have an HTTP proxy server configured, the proxy must allow port 443/80 traffic from IPS systems.
- You must have a valid IPS license to allow global correlation features to function.
- Global correlation features only contain external IP addresses, so if you position a sensor in an internal lab, you may never receive global correlation information.
- Make sure your sensor supports the global correlation features.
- Make sure your IPS version supports the global correlation features.

**For More Information**

For detailed information about global correlation, see [Chapter 10, “Configuring Global Correlation.”](#)

## Troubleshooting External Product Interfaces

This section lists issues that can occur with external product interfaces and provides troubleshooting tips. For more information on external product interfaces, see [Chapter 11, “Configuring External Product Interfaces.”](#) This section contains the following topics:

- [External Product Interfaces Issues, page C-21](#)
- [External Product Interfaces Troubleshooting Tips, page C-22](#)

### External Product Interfaces Issues

When the external product interface receives host posture and quarantine events, the following issues can arise:

- The sensor can store only a certain number of host records:
  - If the number of records exceeds 10,000, subsequent records are dropped.
  - If the 10,000 limit is reached and then it drops to below 9900, new records are no longer dropped.
- Hosts can change an IP address or appear to use another host IP address, for example, because of DHCP lease expiration or movement in a wireless network. In the case of an IP address conflict, the sensor presumes the most recent host posture event to be the most accurate.
- A network can include overlapping IP address ranges in different VLANs, but host postures do not include VLAN ID information. You can configure the sensor to ignore specified address ranges.
- A host can be unreachable from the CSA MC because it is behind a firewall. You can exclude unreachable hosts.
- The CSA MC event server allows up to ten open subscriptions by default. You can change this value. You must have an administrative account and password to open subscriptions.
- CSA data is not virtualized; it is treated globally by the sensor.
- Host posture OS and IP addresses are integrated into passive OS fingerprinting storage. You can view them as imported OS profiles.
- You cannot see the quarantined hosts.
- The sensor must recognize each CSA MC host X.509 certificate. You must add them as a trusted host.

- You can configure a maximum of two external product devices.

**For More Information**

- For more information on working with OS maps and identifications, see [Adding, Editing, Deleting, and Moving Configured OS Maps, page 8-28](#) and [Displaying and Clearing OS Identifications, page 8-32](#).
- For the procedure for adding trusted hosts, see [Adding TLS Trusted Hosts, page 4-53](#).

## External Product Interfaces Troubleshooting Tips

To troubleshoot external product interfaces, check the following:

- Make sure the interface is active by checking the output from the **show statistics external-product-interface** command in the CLI, or choose **Monitoring > Sensor Monitoring > Support Information > Statistics** in the IDM and check the Interface state line in the response, or choose **Configuration > sensor\_name > Sensor Monitoring > Support Information > Statistics** in the IME, and check the Interface state line in the response.
- Make sure you have added the CSA MC IP address to the trusted hosts. If you forgot to add it, add it, wait a few minutes and then check again.
- Confirm subscription login information by opening and closing a subscription on the CSA MC using the browser.
- Check the Event Store for the CSA MC subscription errors.

**For More Information**

- For the procedure for adding trusted hosts, see [Adding TLS Trusted Hosts, page 4-53](#).
- For the procedure for displaying events, see [Displaying Events, page C-99](#).

## Troubleshooting the Appliance

This section contains information to troubleshoot the appliance. It contains the following topics:

- [Troubleshooting Loose Connections, page C-23](#)
- [The Analysis Engine is Busy, page C-23](#)
- [Communication Problems, page C-24](#)
- [Communication Problems, page C-24](#)
- [The SensorApp and Alerting, page C-28](#)
- [Blocking, page C-35](#)
- [Logging, page C-44](#)
- [TCP Reset Not Occurring for a Signature, page C-49](#)
- [Software Upgrades, page C-51](#)



**Tip**

---

Before troubleshooting the appliance, check the Caveats section of the Readme for the software version you have installed on your sensor to see if you are dealing with a known issue.

---

## Troubleshooting Loose Connections

Perform the following actions to troubleshoot loose connections on sensors:

- Make sure all power cords are securely connected.
- Make sure all cables are properly aligned and securely connected for all external and internal components.
- Remove and check all data and power cables for damage. Make sure no cables have bent pins or damaged connectors.
- Make sure each device is properly seated.
- If a device has latches, make sure they are completely closed and locked.
- Check any interlock or interconnect indicators that indicate a component is not connected properly.
- If problems continue, remove and reinstall each device, checking the connectors and sockets for bent pins or other damage.

## The Analysis Engine is Busy

After you reimage a sensor, the Analysis Engine is busy rebuilding Regex tables and does not respond to new configurations. You can check whether the Analysis Engine is busy by using the **show statistics virtual-sensor** command. You receive the following error message if the Analysis Engine is busy:

```
sensor# show statistics virtual-sensor
Error: getVirtualSensorStatistics : Analysis Engine is busy rebuilding regex tables. This
may take a while.
sensor#
```

When the Analysis Engine is busy rebuilding Regex tables, you receive an error message if you try to update a configuration, for example, enabling or retiring a signature:

```
sensor# configure terminal
sensor(config)# service sig sig0
sensor(config-sig)# sig 2000 0
sensor(config-sig-sig)# status enabled
sensor(config-sig-sig)# status
sensor(config-sig-sig-sta)# enabled true
sensor(config-sig-sig-sta)# retired false
sensor(config-sig-sig-sta)# exit
sensor(config-sig-sig)# exit
sensor(config-sig)# exit
Apply Changes?[yes]:
Error: editConfigDeltaSignatureDefinition : Analysis Engine is busy rebuilding regex
tables. This may take a while.
The configuration changes failed validation, no changes were applied.
Would you like to return to edit mode to correct the errors? [yes]: no
No changes were made to the configuration.
sensor(config)#
```

If you try to get the virtual sensor statistics immediately after you boot a sensor, you receive an error message. Although the sensor has rebuilt the cache files, the virtual sensor is not finished initializing.

```
sensor# show statistics virtual-sensor
Error: getVirtualSensorStatistics : Analysis Engine is busy.
sensor#
```

When you receive the errors that the Analysis Engine is busy, wait a while before trying to make configuration changes. Use the **show statistics virtual-sensor** command to find out when the Analysis Engine is available again.

## Communication Problems

This section helps you troubleshoot communication problems with the 4200 series sensor. It contains the following topics:

- [Cannot Access the Sensor CLI Through Telnet or SSH, page C-24](#)
- [Correcting a Misconfigured Access List, page C-26](#)
- [Duplicate IP Address Shuts Interface Down, page C-27](#)

### Cannot Access the Sensor CLI Through Telnet or SSH

If you cannot access the sensor CLI through Telnet (if you already have it enabled) or SSH, follow these steps:

- 
- Step 1** Log in to the sensor CLI through a console, terminal, or module session.
  - Step 2** Make sure that the sensor management interface is enabled. The management interface is the interface in the list with the status line `Media Type = TX`. If the Link Status is `Down`, go to Step 3. If the Link Status is `Up`, go to Step 5.

```

sensor# show interfaces
Interface Statistics
 Total Packets Received = 0
 Total Bytes Received = 0
 Missed Packet Percentage = 0
 Current Bypass Mode = Auto_off
MAC statistics from interface GigabitEthernet0/1
 Media Type = backplane
 Missed Packet Percentage = 0
 Inline Mode = Unpaired
 Pair Status = N/A
 Link Status = Up
 Link Speed = Auto_1000
 Link Duplex = Auto_Full
 Total Packets Received = 0
 Total Bytes Received = 0
 Total Multicast Packets Received = 0
 Total Broadcast Packets Received = 0
 Total Jumbo Packets Received = 0
 Total Undersize Packets Received = 0
 Total Receive Errors = 0
 Total Receive FIFO Overruns = 0
 Total Packets Transmitted = 0
 Total Bytes Transmitted = 0
 Total Multicast Packets Transmitted = 0
 Total Broadcast Packets Transmitted = 0
 Total Jumbo Packets Transmitted = 0
 Total Undersize Packets Transmitted = 0
 Total Transmit Errors = 0
 Total Transmit FIFO Overruns = 0
MAC statistics from interface GigabitEthernet0/0
 Media Type = TX
 Link Status = Up

```

```

Link Speed = Auto_100
Link Duplex = Auto_Full
Total Packets Received = 944333
Total Bytes Received = 83118358
Total Multicast Packets Received = 0
Total Receive Errors = 0
Total Receive FIFO Overruns = 0
Total Packets Transmitted = 397633
Total Bytes Transmitted = 435730956
Total Transmit Errors = 0
Total Transmit FIFO Overruns = 0
sensor#

```

- Step 3** Make sure the sensor IP address is unique. If the management interface detects that another device on the network has the same IP address, it does not come up.

```

sensor# setup
--- System Configuration Dialog ---

```

At any point you may enter a question mark '?' for help.  
 User ctrl-c to abort configuration dialog at any prompt.  
 Default settings are in square brackets '[]'.

Current Configuration:

```

service host
network-settings
host-ip 192.168.1.2/24,192.168.1.1
host-name sensor
telnet-option enabled
access-list 0.0.0.0/0
ftp-timeout 300
no login-banner-text
exit
--MORE--

```

- Step 4** Make sure the management port is connected to an active network connection. If the management port is not connected to an active network connection, the management interface does not come up.

- Step 5** Make sure the IP address of the workstation that is trying to connect to the sensor is permitted in the sensor access list. If the workstation network address is permitted in the sensor access list, go to Step 6.

```

sensor# setup
--- System Configuration Dialog ---

```

At any point you may enter a question mark '?' for help.  
 User ctrl-c to abort configuration dialog at any prompt.  
 Default settings are in square brackets '[]'.

Current Configuration:

```

service host
network-settings
host-ip 192.168.1.2/24,192.168.1.1
host-name sensor
telnet-option enabled
access-list 0.0.0.0/0
ftp-timeout 300
no login-banner-text
exit

```

--MORE--

- Step 6** Add a permit entry for the workstation network address, save the configuration, and try to connect again.
- Step 7** Make sure the network configuration allows the workstation to connect to the sensor. If the sensor is protected behind a firewall and the workstation is in front of the firewall, make sure the firewall is configured to allow the workstation to access the sensor. Or if the workstation is behind a firewall that is performing network address translation on the workstation IP address, and the sensor is in front of the firewall, make sure that the sensor access list contains a permit entry for the workstation translated address.
- 

#### For More Information

- For the procedure for enabling and disabling Telnet on the sensor, see [Enabling and Disabling Telnet, page 4-5](#).
- For the various ways to open a CLI session directly on the sensor, see [Chapter 2, “Logging In to the Sensor.”](#)
- For the procedure for changing the IP address, see [Changing the IP Address, Netmask, and Gateway, page 4-4](#).
- For the procedure for changing the access list, see [Correcting a Misconfigured Access List, page C-26](#).

## Correcting a Misconfigured Access List

To correct a misconfigured access list, follow these steps:

---

- Step 1** Log in to the CLI.

- Step 2** View your configuration to see the access list.

```
sensor# show configuration | include access-list
access-list 10.0.0.0/8
access-list 64.0.0.0/8
sensor#
```

- Step 3** Verify that the client IP address is listed in the allowed networks. If it is not, add it.

```
sensor# configure terminal
sensor(config)# service host
sensor(config-hos)# network-settings
sensor(config-hos-net)# access-list 171.69.70.0/24
```

- Step 4** Verify the settings.

```
sensor(config-hos-net)# show settings
network-settings

host-ip: 192.168.1.2/24,192.168.1.1 default: 10.1.9.201/24,10.1.9.1
host-name: sensor-238 default: sensor
telnet-option: enabled default: disabled
access-list (min: 0, max: 512, current: 3)

network-address: 10.0.0.0/8

network-address: 64.0.0.0/8

network-address: 171.69.70.0/24
```

```

ftp-timeout: 300 seconds <defaulted>
login-banner-text: <defaulted>

sensor(config-hos-net)#

```

## Duplicate IP Address Shuts Interface Down

If you have two newly imaged sensors with the same IP address that come up on the same network at the same time, the interface shuts down. Linux prevents the command and control interface from activating if it detects an address conflict with another host.

To verify that the sensor in question does not have an IP address conflict with another host on the network, follow these steps:

- 
- Step 1** Log in to the CLI.
  - Step 2** Determine whether the interface is up. If the output says the command and control interface link status is down, there is a hardware issue or an IP address conflict.

```

sensor# show interfaces
Interface Statistics
 Total Packets Received = 0
 Total Bytes Received = 0
 Missed Packet Percentage = 0
 Current Bypass Mode = Auto_off
MAC statistics from interface GigabitEthernet0/1
 Media Type = backplane
 Missed Packet Percentage = 0
 Inline Mode = Unpaired
 Pair Status = N/A
 Link Status = Up
 Link Speed = Auto_1000
 Link Duplex = Auto_Full
 Total Packets Received = 0
 Total Bytes Received = 0
 Total Multicast Packets Received = 0
 Total Broadcast Packets Received = 0
 Total Jumbo Packets Received = 0
 Total Undersize Packets Received = 0
 Total Receive Errors = 0
 Total Receive FIFO Overruns = 0
 Total Packets Transmitted = 0
 Total Bytes Transmitted = 0
 Total Multicast Packets Transmitted = 0
 Total Broadcast Packets Transmitted = 0
 Total Jumbo Packets Transmitted = 0
 Total Undersize Packets Transmitted = 0
 Total Transmit Errors = 0
 Total Transmit FIFO Overruns = 0
MAC statistics from interface GigabitEthernet0/0
 Media Type = TX
 Link Status = Up
 Link Speed = Auto_100
 Link Duplex = Auto_Full
 Total Packets Received = 1822323
 Total Bytes Received = 131098876
 Total Multicast Packets Received = 20

```

```

Total Receive Errors = 0
Total Receive FIFO Overruns = 0
Total Packets Transmitted = 219260
Total Bytes Transmitted = 103668610
Total Transmit Errors = 0
Total Transmit FIFO Overruns = 0
sensor#

```

**Step 3** Make sure the sensor cabling is correct.

**Step 4** Make sure the IP address is correct.

---

#### For More Information

- To make sure the sensor cabling is correct, refer to the chapter for your sensor in *Cisco Intrusion Prevention System Appliance and Module Installation Guide for IPS 7.3*.
- For the procedure for making sure the IP address is correct, see *Changing Network Settings*, page 4-2.

## The SensorApp and Alerting

This section helps you troubleshoot issues with the SensorApp and alerting. It contains the following topics:

- [The SensorApp is Not Running](#), page C-28
- [Physical Connectivity, SPAN, or VACL Port Issue](#), page C-30
- [Unable to See Alerts](#), page C-31
- [Sensor Not Seeing Packets](#), page C-33
- [Cleaning Up a Corrupted SensorApp Configuration](#), page C-34

## The SensorApp is Not Running

The sensing process, SensorApp, should always be running. If it is not, you do not receive any alerts. The SensorApp is part of the Analysis Engine, so you must make sure the Analysis Engine is running.

To make sure the Analysis Engine is running, follow these steps:

---

**Step 1** Log in to the CLI.

**Step 2** Determine the status of the Analysis Engine service and whether you have the latest software updates.

```

sensor# show version
Application Partition:

Cisco Intrusion Prevention System, Version 7.3(1)E4

Host:
 Realm Keys key1.0
Signature Definition:
 Signature Update S741.0 2013-09-10
 Threat Profile Version 2
OS Version: 2.6.29.1
Platform: IPS-4360
Serial Number: FGL1702401M

```



```

Licensed, expires: 21-Nov-2014 UTC
Sensor up-time is 22:09.
Using 14372M out of 15943M bytes of available memory (90% usage)
system is using 32.4M out of 160.0M bytes of available disk space (20% usage)
application-data is using 85.6M out of 376.4M bytes of available disk space (24% usage)
boot is using 63.1M out of 70.2M bytes of available disk space (95% usage)
application-log is using 494.0M out of 513.0M bytes of available disk space (96% usage)

```

```

MainApp C-2013_12_16_14_00_7_3_0_143 (Release) 2013-12-16T14:06:20-0600
Running
AnalysisEngine C-2013_12_16_14_00_7_3_0_143 (Release) 2013-12-16T14:06:20-0600
Running
CollaborationApp C-2013_12_16_14_00_7_3_0_143 (Release) 2013-12-16T14:06:20-0600
Running
CLI C-2013_12_16_14_00_7_3_0_143 (Release) 2013-12-16T14:06:20-0600

```

Upgrade History:

```
IPS-K9-7.3-1-E4 11:22:07 UTC Sat Jan 19 2013
```

Recovery Partition Version 1.1 - 7.3(1)E4

Host Certificate Valid from: 09-Oct-2014 to 09-Oct-2016

sensor#

**Step 3** If the Analysis Engine is not running, look for any errors connected to it.

```

sensor# show events error fatal past 13:00:00 | include AnalysisEngine
evError: eventId=1077219258696330005 severity=warning

```

```

originator:
hostId: sensor
appName: sensorApp
appInstanceId: 1045
time: 2004/02/19 19:34:20 2004/02/19 19:34:20 UTC
errorMessage: name=errUnclassified Generating new Analysis Engine configuration file.

```



**Note** The date and time of the last restart is listed. In this example, the last restart was on 2-19-2004 at 7:34.

- Step 4** If you do not have the latest software updates, download them from Cisco.com. Read the Readme that accompanies the software upgrade for any known DDTS for the SensorApp or the Analysis Engine.
- Step 5** If the Analysis Engine is still not running, enter **show tech-support** and save the output.
- Step 6** Reboot the sensor.
- Step 7** Enter **show version** after the sensor has stabilized to see if the issue is resolved.
- Step 8** If the Analysis Engine still reads `Not Running`, contact TAC with the original **show tech support** command output.

**For More Information**

- For more information on IPS system architecture, see [Chapter A, “System Architecture.”](#)
- For the procedure for obtaining the latest Cisco IPS software, see [Obtaining Cisco IPS Software, page 20-1.](#)

**Physical Connectivity, SPAN, or VACL Port Issue**

If the sensor is not connected properly, you do not receive any alerts.

To make sure the sensor is connected properly, follow these steps:

- 
- Step 1** Log in to the CLI.
  - Step 2** Make sure the interfaces are up and that the packet count is increasing.

```

sensor# show interfaces
Interface Statistics
 Total Packets Received = 0
 Total Bytes Received = 0
 Missed Packet Percentage = 0
 Current Bypass Mode = Auto_off
MAC statistics from interface GigabitEthernet0/1
 Media Type = backplane
 Missed Packet Percentage = 0
 Inline Mode = Unpaired
 Pair Status = N/A
 Link Status = Up
 Link Speed = Auto_1000
 Link Duplex = Auto_Full
 Total Packets Received = 0
 Total Bytes Received = 0
 Total Multicast Packets Received = 0
 Total Broadcast Packets Received = 0
 Total Jumbo Packets Received = 0
 Total Undersize Packets Received = 0
 Total Receive Errors = 0
 Total Receive FIFO Overruns = 0
 Total Packets Transmitted = 0
 Total Bytes Transmitted = 0
 Total Multicast Packets Transmitted = 0
 Total Broadcast Packets Transmitted = 0
 Total Jumbo Packets Transmitted = 0
 Total Undersize Packets Transmitted = 0
 Total Transmit Errors = 0
 Total Transmit FIFO Overruns = 0
MAC statistics from interface GigabitEthernet0/0
 Media Type = TX
 Link Status = Up
 Link Speed = Auto_100
 Link Duplex = Auto_Full
 Total Packets Received = 1830137
 Total Bytes Received = 131624465
 Total Multicast Packets Received = 20
 Total Receive Errors = 0
 Total Receive FIFO Overruns = 0
 Total Packets Transmitted = 220052
 Total Bytes Transmitted = 103796666
 Total Transmit Errors = 0
 Total Transmit FIFO Overruns = 0
sensor#

```

- Step 3** If the Link Status is down, make sure the sensing port is connected properly.
- Step 4** Verify the interface configuration:
- Make sure you have the interfaces configured properly.
  - Verify the SPAN and VACL capture port configuration on the Cisco switch. Refer to your switch documentation for the procedure.
- Step 5** Verify again that the interfaces are up and that the packet count is increasing.

```
sensor# show interfaces
```

---

#### For More Information

- For the procedure for properly installing the sensing interface on your sensor, refer to the chapter on your appliance in *Cisco Intrusion Prevention System Appliance and Module Installation Guide for IPS 7.3*.
- For the procedures for configuring interfaces on your sensor, see [Chapter 5, “Configuring Interfaces.”](#)

## Unable to See Alerts

If you are not seeing alerts, try the following:

- Make sure the signature is enabled
- Make sure the signature is not retired
- Make sure that you have Produce Alert configured as an action



**Note** If you choose Produce Alert, but come back later and add another event action and do not add Produce Alert to the new configuration, alerts are not sent to the Event Store. Every time you configure a signature, the new configuration overwrites the old one, so make sure you have configured all the event actions you want for each signature.

---

- Make sure the sensor is seeing packets
- Make sure that alerts are being generated
- Make sure the sensing interface is in a virtual sensor

To make sure you can see alerts, follow these steps:

---

**Step 1** Log in to the CLI.

**Step 2** Make sure the signature is enabled.

```
sensor# configure terminal
sensor(config)# service signature-definition sig0
sensor(config-sig)# signatures 1300 0
sensor(config-sig-sig)# status
sensor(config-sig-sig-sta)# show settings
status

enabled: true <defaulted>
retired: false <defaulted>

sensor(config-sig-sig-sta)#
```

**Step 3** Make sure you have Produce Alert configured.

```

sensor# configure terminal
sensor(config)# service signature-definition sig0
sensor(config-sig)# signatures 1300 0
sensor(config-sig-sig)# engine ?
normalizer Signature engine
sensor(config-sig-sig)# engine normalizer
sensor(config-sig-sig-nor)# event-action produce-alert
sensor(config-sig-sig-nor)# show settings
normalizer

event-action: produce-alert default: produce-alert|deny-connection-inline
edit-default-sigs-only

sensor#

```

**Step 4** Make sure the sensor is seeing packets.

```

sensor# show interfaces FastEthernet0/1
MAC statistics from interface FastEthernet0/1
Media Type = backplane
Missed Packet Percentage = 0
Inline Mode = Unpaired
Pair Status = N/A
Link Status = Up
Link Speed = Auto_100
Link Duplex = Auto_Full
Total Packets Received = 267581
Total Bytes Received = 24886471
Total Multicast Packets Received = 0
Total Broadcast Packets Received = 0
Total Jumbo Packets Received = 0
Total Undersize Packets Received = 0
Total Receive Errors = 0
Total Receive FIFO Overruns = 0
Total Packets Transmitted = 57301
Total Bytes Transmitted = 3441000
Total Multicast Packets Transmitted = 0
Total Broadcast Packets Transmitted = 0
Total Jumbo Packets Transmitted = 0
Total Undersize Packets Transmitted = 0
Total Transmit Errors = 1
Total Transmit FIFO Overruns = 0
sensor#

```

**Step 5** Check for alerts.

```

sensor# show statistics virtual-sensor
SigEvent Preliminary Stage Statistics
Number of Alerts received = 0
Number of Alerts Consumed by AlertInterval = 0
Number of Alerts Consumed by Event Count = 0
Number of FireOnce First Alerts = 0
Number of FireOnce Intermediate Alerts = 0
Number of Summary First Alerts = 0
Number of Summary Intermediate Alerts = 0
Number of Regular Summary Final Alerts = 0
Number of Global Summary Final Alerts = 0
Number of Alerts Output for further processing = 0alertDetails: Traffic Source: int0 ;

```

---

## Sensor Not Seeing Packets

If the sensor is not seeing any packets on the network, you could have the interfaces set up incorrectly.

If the sensor is not seeing packets, follow these steps:

**Step 1** Log in to the CLI.

**Step 2** Make sure the interfaces are up and receiving packets.

```
sensor# show interfaces GigabitEthernet0/1
MAC statistics from interface GigabitEthernet0/1
 Media Type = backplane
 Missed Packet Percentage = 0
 Inline Mode = Unpaired
 Pair Status = N/A
 Link Status = Down
 Link Speed = Auto_1000
 Link Duplex = Auto_Full
 Total Packets Received = 0
 Total Bytes Received = 0
 Total Multicast Packets Received = 0
 Total Broadcast Packets Received = 0
 Total Jumbo Packets Received = 0
 Total Undersize Packets Received = 0
 Total Receive Errors = 0
 Total Receive FIFO Overruns = 0
 Total Packets Transmitted = 0
 Total Bytes Transmitted = 0
 Total Multicast Packets Transmitted = 0
 Total Broadcast Packets Transmitted = 0
 Total Jumbo Packets Transmitted = 0
 Total Undersize Packets Transmitted = 0
 Total Transmit Errors = 0
 Total Transmit FIFO Overruns = 0
sensor#
```

**Step 3** If the interfaces are not up, do the following:

- Check the cabling.
- Enable the interface.

```
sensor# configure terminal
sensor(config)# service interface
sensor(config-int)# physical-interfaces GigabitEthernet0/1
sensor(config-int-phy)# admin-state enabled
sensor(config-int-phy)# show settings
<protected entry>
name: GigabitEthernet0/1

media-type: tx <protected>
description: <defaulted>
admin-state: enabled default: disabled
duplex: auto <defaulted>
speed: auto <defaulted>
alt-tcp-reset-interface

none

sensor(config-int-phy)#
```

**Step 4** Check to see that the interface is up and receiving packets.

```

sensor# show interfaces
MAC statistics from interface GigabitEthernet0/1
 Media Type = TX
 Missed Packet Percentage = 0
 Inline Mode = Unpaired
 Pair Status = N/A
 Link Status = Up
 Link Speed = Auto_100
 Link Duplex = Auto_Full
 Total Packets Received = 3
 Total Bytes Received = 900
 Total Multicast Packets Received = 3
 Total Broadcast Packets Received = 0
 Total Jumbo Packets Received = 0
 Total Undersize Packets Received = 0
 Total Receive Errors = 0
 Total Receive FIFO Overruns = 0
 Total Packets Transmitted = 0
 Total Bytes Transmitted = 0
 Total Multicast Packets Transmitted = 0
 Total Broadcast Packets Transmitted = 0
 Total Jumbo Packets Transmitted = 0
 Total Undersize Packets Transmitted = 0
 Total Transmit Errors = 0
 Total Transmit FIFO Overruns = 0 ...

```

---

**For More Information**

For the procedure for installing the sensor properly, refer to your sensor chapter in *Cisco Intrusion Prevention System Appliance and Module Installation Guide for IPS 7.3*.

## Cleaning Up a Corrupted SensorApp Configuration

If the SensorApp configuration has become corrupted and the SensorApp cannot run, you must delete it entirely and restart the SensorApp.

To delete the SensorApp configuration, follow these steps:

- 
- Step 1** Log in to the service account.
  - Step 2** Su to root.
  - Step 3** Stop the IPS applications.  
`/etc/init.d/cids stop`
  - Step 4** Replace the virtual sensor file.  
`cp /usr/cids/idsRoot/etc/defVirtualSensorConfig.xml /usr/cids/idsRoot/etc/VS-Config/virtualSensor.xml`
  - Step 5** Remove the cache files.  
`rm /usr/cids/idsRoot/var/virtualSensor/*.pmz`
  - Step 6** Exit the service account.
  - Step 7** Log in to the sensor CLI.

**Step 8** Start the IPS services.

```
sensor# cids start
```

**Step 9** Log in to an account with administrator privileges.

**Step 10** Reboot the sensor.

```
sensor# reset
Warning: Executing this command will stop all applications and reboot the node.
Continue with reset? [yes]:yes
Request Succeeded.
sensor#
```

---

### For More Information

To learn more about IPS system architecture, see [Appendix A, “System Architecture.”](#)

## Blocking

This section provides troubleshooting help for blocking and the ARC service. It contains the following topics.

- [Troubleshooting Blocking, page C-35](#)
- [Verifying the ARC is Running, page C-36](#)
- [Verifying ARC Connections are Active, page C-37](#)
- [Device Access Issues, page C-39](#)
- [Verifying the Interfaces and Directions on the Network Device, page C-40](#)
- [Enabling SSH Connections to the Network Device, page C-41](#)
- [Blocking Not Occurring for a Signature, page C-41](#)
- [Verifying the Master Blocking Sensor Configuration, page C-42](#)

## Troubleshooting Blocking

After you have configured the ARC, you can verify if it is running properly by using the **show version** command. To verify that the ARC is connecting to the network devices, use the **show statistics network-access** command.



### Note

The ARC was formerly known as Network Access Controller. Although the name has been changed since IPS 5.1, it still appears in IDM, IME, and the CLI as Network Access Controller, **nac**, and **network-access**.

---

To troubleshoot the ARC, follow these steps:

1. Verify that the ARC is running.
2. Verify that the ARC is connecting to the network devices.
3. Verify that the Event Action is set to Block Host for specific signatures.
4. Verify that the master blocking sensor is properly configured.

**For More Information**

- For the procedure to verify that the ARC is running, see [Verifying the ARC is Running, page C-36](#).
- For the procedure to verify that the ARC is connecting, see [Verifying ARC Connections are Active, page C-37](#).
- For the procedure to verify that the Event Action is set to Block Host, see [Blocking Not Occurring for a Signature, page C-41](#).
- For the procedure to verify that the master blocking sensor is properly configured, see [Verifying the Master Blocking Sensor Configuration, page C-42](#).
- For a discussion of ARC architecture, see [Attack Response Controller, page A-12](#).

## Verifying the ARC is Running

To verify that the ARC is running, use the **show version** command. If the MainApp is not running, the ARC cannot run. The ARC is part of the MainApp.

To verify that the ARC is running, follow these steps:

- 
- Step 1** Log in to the CLI.
  - Step 2** Verify that the MainApp is running.

```

sensor# show version
Application Partition:

Cisco Intrusion Prevention System, Version 7.3(1)E4

Host:
 Realm Keys key1.0
Signature Definition:
 Signature Update S741.0 2013-09-10
 Threat Profile Version 2
OS Version: 2.6.29.1
Platform: IPS-4360
Serial Number: FGL1702401M
Licensed, expires: 21-Nov-2014 UTC
Sensor up-time is 22:09.
Using 14372M out of 15943M bytes of available memory (90% usage)
system is using 32.4M out of 160.0M bytes of available disk space (20% usage)
application-data is using 85.6M out of 376.4M bytes of available disk space (24% usage)
boot is using 63.1M out of 70.2M bytes of available disk space (95% usage)
application-log is using 494.0M out of 513.0M bytes of available disk space (96% usage)

MainApp C-2013_12_16_14_00_7_3_0_143 (Release) 2013-12-16T14:06:20-0600
Running
AnalysisEngine C-2013_12_16_14_00_7_3_0_143 (Release) 2013-12-16T14:06:20-0600
Running
CollaborationApp C-2013_12_16_14_00_7_3_0_143 (Release) 2013-12-16T14:06:20-0600
Running
CLI C-2013_12_16_14_00_7_3_0_143 (Release) 2013-12-16T14:06:20-0600

Upgrade History:

 IPS-K9-7.3-1-E4 11:22:07 UTC Sat Jan 19 2013

Recovery Partition Version 1.1 - 7.3(1)E4

Host Certificate Valid from: 09-Oct-2014 to 09-Oct-2016

```



```
sensor#
```

- Step 3** If the MainApp displays `Not Running`, the ARC has failed. Contact TAC.

### For More Information

To learn more about IPS system architecture, see [Appendix A, “System Architecture.”](#)

## Verifying ARC Connections are Active

If the State is not `Active` in the ARC statistics, there is a problem.

To verify that the State is `Active` in the statistics, follow these steps:

- Step 1** Log in to the CLI.

- Step 2** Verify that the ARC is connecting. Check the State section of the output to verify that all devices are connecting.

```
sensor# show statistics network-access
Current Configuration
 LogAllBlockEventsAndSensors = true
 EnableNvramWrite = false
 EnableAclLogging = false
 AllowSensorBlock = false
 BlockMaxEntries = 250
 MaxDeviceInterfaces = 250
NetDevice
 Type = Cisco
 IP = 10.89.147.54
 NATAddr = 0.0.0.0
 Communications = telnet
 BlockInterface
 InterfaceName = fa0/0
 InterfaceDirection = in
State
 BlockEnable = true
 NetDevice
 IP = 10.89.147.54
 AclSupport = uses Named ACLs
 Version = 12.2
 State = Active
sensor#
```

- Step 3** If the ARC is not connecting, look for recurring errors.

```
sensor# show events error hh:mm:ss month day year | include : nac
```

### Example

```
sensor# show events error 00:00:00 Apr 01 2011 | include : nac
```

- Step 4** Make sure you have the latest software updates.

```
sensor# show version
Application Partition:

Cisco Intrusion Prevention System, Version 7.3(1)E4
```

```

Host:
 Realm Keys key1.0
Signature Definition:
 Signature Update S741.0 2013-09-10
 Threat Profile Version 2
OS Version: 2.6.29.1
Platform: IPS-4360
Serial Number: FGL1702401M
Licensed, expires: 21-Nov-2014 UTC
Sensor up-time is 22:09.
Using 14372M out of 15943M bytes of available memory (90% usage)
system is using 32.4M out of 160.0M bytes of available disk space (20% usage)
application-data is using 85.6M out of 376.4M bytes of available disk space (24% usage)
boot is using 63.1M out of 70.2M bytes of available disk space (95% usage)
application-log is using 494.0M out of 513.0M bytes of available disk space (96% usage)

MainApp C-2013_12_16_14_00_7_3_0_143 (Release) 2013-12-16T14:06:20-0600
Running
AnalysisEngine C-2013_12_16_14_00_7_3_0_143 (Release) 2013-12-16T14:06:20-0600
Running
CollaborationApp C-2013_12_16_14_00_7_3_0_143 (Release) 2013-12-16T14:06:20-0600
Running
CLI C-2013_12_16_14_00_7_3_0_143 (Release) 2013-12-16T14:06:20-0600

Upgrade History:

 IPS-K9-7.3-1-E4 11:22:07 UTC Sat Jan 19 2013

Recovery Partition Version 1.1 - 7.3(1)E4

Host Certificate Valid from: 09-Oct-2014 to 09-Oct-2016

sensor#

```




---

**Note** If you do not have the latest software updates, download them from Cisco.com. Read the Readme that accompanies the software upgrade for any known DDTS for the ARC.

---

- Step 5** Make sure the configuration settings for each device are correct (the username, password, and IP address).
  - Step 6** Make sure the interface and directions for each network device are correct.
  - Step 7** If the network device is using SSH-3DES, make sure that you have enabled SSH connections to the device.
  - Step 8** Verify that each interface and direction on each controlled device is correct.
- 

#### For More Information

- For the procedure for obtaining the latest Cisco IPS software, see [Obtaining Cisco IPS Software, page 20-1](#).
- For more information about configuring devices, see [Device Access Issues, page C-39](#).
- For the procedure for verifying the interfaces and directions for each network device, see [Verifying the Interfaces and Directions on the Network Device, page C-40](#).
- For the procedure for enabling SSH, see [Enabling SSH Connections to the Network Device, page C-41](#).

## Device Access Issues

The ARC may not be able to access the devices it is managing. Make sure the you have the correct IP address and username and password for the managed devices and the correct interface and direction configured.

To troubleshoot device access issues, follow these steps:

- 
- Step 1** Log in to the CLI.
  - Step 2** Verify the IP address for the managed devices.

```

sensor# configure terminal
sensor (config)# service network-access
sensor(config-net)# show settings
 general

 log-all-block-events-and-errors: true <defaulted>
 enable-nvram-write: false <defaulted>
 enable-acl-logging: false <defaulted>
 allow-sensor-block: false <defaulted>
 block-enable: true <defaulted>
 block-max-entries: 250 <defaulted>
 max-interfaces: 250 <defaulted>
 master-blocking-sensors (min: 0, max: 100, current: 0)

 never-block-hosts (min: 0, max: 250, current: 0)

 never-block-networks (min: 0, max: 250, current: 0)

 block-hosts (min: 0, max: 250, current: 0)

 block-networks (min: 0, max: 250, current: 0)

 user-profiles (min: 0, max: 250, current: 1)

 profile-name: r7200

 enable-password: <hidden>
 password: <hidden>
 username: netrangr default:

 cat6k-devices (min: 0, max: 250, current: 0)

 router-devices (min: 0, max: 250, current: 1)

 ip-address: 10.89.147.54

 communication: telnet default: ssh-3des
 nat-address: 0.0.0.0 <defaulted>
 profile-name: r7200
 block-interfaces (min: 0, max: 100, current: 1)

 interface-name: fa0/0
 direction: in

```

```

pre-acl-name: <defaulted>
post-acl-name: <defaulted>

firewall-devices (min: 0, max: 250, current: 0)

sensor(config-net)#

```

- Step 3** Manually connect to the device to make sure you have used the correct username, password, and enable password, and to ensure that the device is reachable from the sensor:
- a. Log in to the service account.
  - b. Telnet or SSH to the network device to verify the configuration.
  - c. Make sure you can reach the device.
  - d. Verify the username and password.
- Step 4** Verify that each interface and direction on each network device is correct.

**For More Information**

For the procedure for verifying the interfaces and directions for each network device, see [Verifying the Interfaces and Directions on the Network Device, page C-40](#).

## Verifying the Interfaces and Directions on the Network Device

To verify that each interface and direction on each controlled device is correct, you can send a manual block to a bogus host and then check to see if deny entries exist for the blocked addresses in the ACL of the router.



**Note**

To perform a manual block using IDM, choose **Monitoring > Sensor Monitoring > Time-Based Actions > Host Blocks**. To perform a manual block using IME, choose **Configuration > sensor\_name > Sensor Monitoring > Time-Based Actions > Host Blocks**.

To initiate a manual block to a bogus host, follow these steps:

- Step 1** Enter ARC general submode.
- ```

sensor# configure terminal
sensor(config)# service network-access
sensor(config-net)# general

```
- Step 2** Start the manual block of the bogus host IP address.
- ```

sensor(config-net-gen)# block-hosts 10.16.0.0

```
- Step 3** Exit general submode.
- ```

sensor(config-net-gen)# exit
sensor(config-net)# exit
Apply Changes:? [yes]:

```

- Step 4** Press **Enter** to apply the changes or type **no** to discard them.
- Step 5** Telnet to the router and verify that a deny entry for the blocked address exists in the router ACL. Refer to the router documentation for the procedure.
- Step 6** Remove the manual block by repeating Steps 1 through 4 except in Step 2 place **no** in front of the command.

```
sensor(config-net-gen)# no block-hosts 10.16.0.0
```

Enabling SSH Connections to the Network Device

If you are using SSH-3DES as the communication protocol for the network device, you must make sure you have enabled it on the device.

To enable SSH-3DES connections to the network device, follow these steps:

- Step 1** Log in to the CLI.
- Step 2** Enter configuration mode.
- ```
sensor# configure terminal
```
- Step 3** Enable SSH-3DES.
- ```
sensor(config)# ssh-3des host blocking_device_ip_address
```
- Step 4** Type **yes** when prompted to accept the device.
-

Blocking Not Occurring for a Signature

If blocking is not occurring for a specific signature, check that the event action is set to block the host.

To make sure blocking is occurring for a specific signature, follow these steps:

- Step 1** Log in to the CLI.
- Step 2** Enter signature definition submode.
- ```
sensor# configure terminal
sensor(config)# service signature-definition sig0
sensor(config-sig)#
```
- Step 3** Make sure the event action is set to block the host.



**Note** If you want to receive alerts, you must always add **produce-alert** any time you configure the event actions.

---

```
sensor(config-sig)# signatures 1300 0
sensor(config-sig-sig)# engine normalizer
sensor(config-sig-sig-nor)# event-action produce-alert|request-block-host
sensor(config-sig-sig-nor)# show settings
normalizer

event-action: produce-alert|request-block-host default: produce-alert|deny
```

```

-connection-inline
 edit-default-sigs-only

 default-signatures-only

 specify-service-ports

 no

 specify-tcp-max-mss

 no

 specify-tcp-min-mss

 no

--MORE--

```

**Step 4** Exit signature definition submode.

```

sensor(config-sig-sig-nor)# exit
sensor(config-sig-sig)# exit
sensor(config-sig)# exit
Apply Changes:[yes]:

```

**Step 5** Press **Enter** to apply the changes or type **no** to discard them.

## Verifying the Master Blocking Sensor Configuration

To verify that a master blocking sensor is set up properly or to troubleshoot a master blocking sensor that is not set up properly, you can use the **show statistics network-access** command. Make sure that the forwarding sensor is set up as TLS trusted host if the remote master blocking sensor is using TLS for web access.

To verify a master blocking sensor configuration, follow these steps:

**Step 1** Log in to the CLI.

**Step 2** View the ARC statistics and verify that the master blocking sensor entries are in the statistics.

```

sensor# show statistics network-access
Current Configuration
 AllowSensorShun = false
 ShunMaxEntries = 250
 MasterBlockingSensor
 SensorIp = 10.89.149.46
 SensorPort = 443
 UseTls = 1
State
 ShunEnable = true
 ShunnedAddr
 Host
 IP = 122.122.122.44
 ShunMinutes = 60

```

```
MinutesRemaining = 59
```

- Step 3** If the master blocking sensor does not show up in the statistics, you need to add it.
- Step 4** Initiate a manual block to a bogus host IP address to make sure the master blocking sensor is initiating blocks.

```
sensor# configure terminal
sensor(config)# service network-access
sensor(config-net)# general
sensor(config-net-gen)# block-hosts 10.16.0.0
```

- Step 5** Exit network access general submode.

```
sensor(config-net-gen)# exit
sensor(config-net)# exit
Apply Changes:? [yes]:
```

- Step 6** Press **Enter** to apply the changes or type **no** to discard them.

- Step 7** Verify that the block shows up in the ARC statistics.

```
sensor# show statistics network-access
Current Configuration
 AllowSensorShun = false
 ShunMaxEntries = 100
State
 ShunEnable = true
 ShunnedAddr
 Host
 IP = 10.16.0.0
 ShunMinutes =
```

- Step 8** Log in to the CLI of the master blocking sensor host, and using the **show statistics network-access** command, verify that the block also shows up in the master blocking sensor ARC statistics.

```
sensor# show statistics network-access
Current Configuration
 AllowSensorShun = false
 ShunMaxEntries = 250
 MasterBlockingSensor
 SensorIp = 10.89.149.46
 SensorPort = 443
 UseTls = 1
State
 ShunEnable = true
 ShunnedAddr
 Host
 IP = 10.16.0.0
 ShunMinutes = 60
 MinutesRemaining = 59
```

- Step 9** If the remote master blocking sensor is using TLS for web access, make sure the forwarding sensor is configured as a TLS host.

```
sensor# configure terminal
sensor(config)# tls trust ip master_blocking_sensor_ip_address
```

### For More Information

For the procedure to configure the sensor to be a master blocking sensor, see [Configuring the Sensor to be a Master Blocking Sensor, page 14-28](#).

# Logging

TAC may suggest that you turn on debug logging for troubleshooting purposes. Logger controls what log messages are generated by each application by controlling the logging severity for different logging zones. By default, debug logging is not turned on. If you enable individual zone control, each zone uses the level of logging that it is configured for. Otherwise, the same logging level is used for all zones. This section contains the following topics:

- [Enabling Debug Logging, page C-44](#)
- [Zone Names, page C-48](#)
- [Directing cidLog Messages to SysLog, page C-48](#)

## Enabling Debug Logging



**Caution** Enabling debug logging seriously affects performance and should only be done when instructed by TAC.

To enable debug logging, follow these steps:

- Step 1** Log in to the service account.
- Step 2** Edit the log.conf file to increase the size of the log to accommodate the additional log statements.
- ```
vi /usr/cids/idsRoot/etc/log.conf
```
- Step 3** Change `fileMaxSizeInK=500` to `fileMaxSizeInK=5000`.
- Step 4** Locate the zone and CID section of the file and set the severity to debug.
- ```
severity=debug
```
- Step 5** Save the file, exit the vi editor, and exit the service account.
- Step 6** Log in to the CLI as administrator.
- Step 7** Enter master control submode.
- ```
sensor# configure terminal
sensor(config)# service logger
sensor(config-log)# master-control
```
- Step 8** Enable debug logging for all zones.
- ```
sensor(config-log-mas)# enable-debug true
sensor(config-log-mas)# show settings
master-control

enable-debug: true default: false
individual-zone-control: false <defaulted>

sensor(config-log-mas)#
```
- Step 9** Turn on individual zone control.
- ```
sensor(config-log-mas)# individual-zone-control true
sensor(config-log-mas)# show settings
master-control
-----
enable-debug: true default: false
individual-zone-control: true default: false
```



```
-----
sensor(config-log-mas)#
```

Step 10 Exit master zone control.

```
sensor(config-log-mas)# exit
```

Step 11 View the zone names.

```
sensor(config-log)# show settings
master-control
-----
enable-debug: false <defaulted>
individual-zone-control: true default: false
-----
zone-control (min: 0, max: 999999999, current: 14)
-----
<protected entry>
zone-name: AuthenticationApp
severity: warning <defaulted>
<protected entry>
zone-name: Cid
severity: debug <defaulted>
<protected entry>
zone-name: Cli
severity: warning <defaulted>
<protected entry>
zone-name: IdapiCtlTrans
severity: warning <defaulted>
<protected entry>
zone-name: IdsEventStore
severity: warning <defaulted>
<protected entry>
zone-name: MpInstaller
severity: warning <defaulted>
<protected entry>
zone-name: cmgr
severity: warning <defaulted>
<protected entry>
zone-name: cplane
severity: warning <defaulted>
<protected entry>
zone-name: csi
severity: warning <defaulted>
<protected entry>
zone-name: ctlTransSource
severity: warning <defaulted>
<protected entry>
zone-name: intf
severity: warning <defaulted>
<protected entry>
zone-name: nac
severity: warning <defaulted>
<protected entry>
zone-name: sensorApp
severity: warning <defaulted>
<protected entry>
zone-name: tls
severity: warning <defaulted>
-----
sensor(config-log)#
```

Step 12 Change the severity level (debug, timing, warning, or error) for a particular zone.

```
sensor(config-log)# zone-control IdsEventStore severity error
```

```

sensor(config-log)# show settings
master-control
-----
enable-debug: true default: false
individual-zone-control: true default: false
-----
zone-control (min: 0, max: 999999999, current: 14)
-----
<protected entry>
zone-name: AuthenticationApp
severity: warning <defaulted>
<protected entry>
zone-name: Cid
severity: debug <defaulted>
<protected entry>
zone-name: Cli
severity: warning <defaulted>
<protected entry>
zone-name: IdapiCtlTrans
severity: warning <defaulted>
<protected entry>
zone-name: IdsEventStore
severity: error default: warning
<protected entry>
zone-name: MpInstaller
severity: warning <defaulted>
<protected entry>
zone-name: cmgr
severity: warning <defaulted>
<protected entry>
zone-name: cplane
severity: warning <defaulted>
<protected entry>
zone-name: csi
severity: warning <defaulted>
<protected entry>
zone-name: ctlTransSource
severity: warning <defaulted>
<protected entry>
zone-name: intf
severity: warning <defaulted>
<protected entry>
zone-name: nac
severity: warning <defaulted>
<protected entry>
zone-name: sensorApp
severity: warning <defaulted>
<protected entry>
zone-name: tls
severity: warning <defaulted>
-----
sensor(config-log)#

```

Step 13 Turn on debugging for a particular zone.

```

sensor(config-log)# zone-control nac severity debug
sensor(config-log)# show settings
master-control
-----
enable-debug: true default: false
individual-zone-control: true default: false
-----
zone-control (min: 0, max: 999999999, current: 14)
-----

```

```

<protected entry>
zone-name: AuthenticationApp
severity: warning <defaulted>
<protected entry>
zone-name: Cid
severity: debug <defaulted>
<protected entry>
zone-name: Cli
severity: warning <defaulted>
<protected entry>
zone-name: IdapiCtlTrans
severity: warning <defaulted>
<protected entry>
zone-name: IdsEventStore
severity: error default: warning
<protected entry>
zone-name: MpInstaller
severity: warning <defaulted>
<protected entry>
zone-name: cmgr
severity: warning <defaulted>
<protected entry>
zone-name: cplane
severity: warning <defaulted>
<protected entry>
zone-name: csi
severity: warning <defaulted>
<protected entry>
zone-name: ctlTransSource
severity: warning <defaulted>
<protected entry>
zone-name: intfci
severity: warning <defaulted>
<protected entry>
zone-name: nac
severity: debug default: warning
<protected entry>
zone-name: sensorApp
severity: warning <defaulted>
<protected entry>
zone-name: tls
severity: warning <defaulted>

```

```
-----
sensor(config-log)#
```

Step 14 Exit the logger submode.

```
sensor(config-log)# exit
Apply Changes:[yes]:
```

Step 15 Press **Enter** to apply changes or type **no** to discard them:

For More Information

For a list of what each zone name refers to, see [Zone Names](#), page C-48.

Zone Names

Table C-2 lists the debug logger zone names:

Table C-2 Debug Logger Zone Names

Zone Name	Description
AD	Anomaly Detection zone
AuthenticationApp	Authentication zone
Cid	General logging zone
Cli	CLI zone
IdapiCtlTrans	All control transactions zone
IdsEventStore	Event Store zone
csi	CIDS Servlet Interface ¹
ctlTransSource	Outbound control transactions zone
intfc	Interface zone
nac	ARC zone
rep	Reputation zone
sched	Automatic update scheduler zone
sensorApp	AnalysisEngine zone
tls	SSL and TLS zone

1. The CIDS servlet interface is the interface layer between the CIDS web server and the servlets.

For More Information

To learn more about the IPS Logger service, see [Logger, page A-19](#).

Directing cidLog Messages to SysLog

It might be useful to direct cidLog messages to syslog.

To direct cidLog messages to syslog, follow these steps:

Step 1 Go to the `idsRoot/etc/log.conf` file.

Step 2 Make the following changes:

a. Set `[logApp] enabled=false`

Comment out the `enabled=true` because `enabled=false` is the default.

b. Set `[drain/main] type=syslog`

The following example shows the logging configuration file:

```
timemode=local
;timemode=utc

[logApp]
;enabled=true
;----- FIFO parameters -----
```

```

fifoName=logAppFifo
fifoSizeInK=240
;----- logApp zone and drain parameters -----
zoneAndDrainName=logApp
fileName=main.log
fileMaxSizeInK=500

[zone/Cid]
severity=warning
drain=main

[zone/IdsEventStore]
severity=debug
drain=main

[drain/main]
type=syslog

```

The syslog output is sent to the syslog facility local6 with the following correspondence to syslog message priorities:

```

LOG_DEBUG,          //  debug
LOG_INFO,           //  timing
LOG_WARNING,       //  warning
LOG_ERR,           //  error
LOG_CRIT,          //  fatal

```



Note Make sure that your /etc/syslog.conf has that facility enabled at the proper priority.



Caution

The syslog is much slower than logApp (about 50 messages per second as opposed to 1000 or so). We recommend that you enable debug severity on one zone at a time.

TCP Reset Not Occurring for a Signature



Note

There is only one sensing interface on the ASA IPS modules (ASA 5500-X IPS SSP and ASA 5585-X IPS SSP), so you cannot designate an alternate TCP reset interface.

If you do not have the event action set to reset, the TCP reset does not occur for a specific signature.



Note

TCP Resets are not supported over MPLS links or the following tunnels: GRE, IPv4 in IPv4, IPv6 in IPv4, or IPv4 in IPv6.

To troubleshoot a reset not occurring for a specific signature, follow these steps:

Step 1 Log in to the CLI.

Step 2 Make sure the event action is set to TCP reset.

```

sensor# configure terminal
sensor(config)# service signature-definition sig0

```

```

sensor(config-sig)# signatures 1000 0
sensor(config-sig-sig)# engine atomic-ip
sensor(config-sig-sig-ato)# event-action reset-tcp-connection|produce-alert
sensor(config-sig-sig-ato)# show settings
  atomic-ip
  -----
  event-action: produce-alert|reset-tcp-connection default: produce-alert
  fragment-status: any <defaulted>
  specify-l4-protocol
  -----
  no
  -----
  -----
  specify-ip-payload-length
  -----
  no
  -----
  -----
  specify-ip-header-length
  -----
  no
  -----
  -----
  specify-ip-tos
  -----
--MORE--

```

Step 3 Exit signature definition submenu.

```

sensor(config-sig-sig-ato)# exit
sensor(config-sig-sig)# exit
sensor(config-sig)# exit
Apply Changes:[yes]:

```

Step 4 Press **Enter** to apply the changes or type **no** to discard them.

Step 5 Make sure the correct alarms are being generated.

```

sensor# show events alert
evAlert: eventId=1047575239898467370 severity=medium
originator:
hostId: sj_4250_40
appName: sensorApp
appInstanceId: 1004
signature: sigId=20000 sigName=STRING.TCP subSigId=0 version=Unknown
addr: locality=OUT 172.16.171.19
port: 32771
victim:
addr: locality=OUT 172.16.171.13 port: 23
actions:
tcpResetSent: true

```

Step 6 Make sure the switch is allowing incoming TCP reset packet from the sensor. Refer to your switch documentation for more information.

Step 7 Make sure the resets are being sent.

```

root# ./tcpdump -i eth0 src host 172.16.171.19
tcpdump: WARNING: eth0: no IPv4 address assigned
tcpdump: listening on eth0
13:58:03.823929 172.16.171.19.32770 > 172.16.171.13.telnet: R 79:79(0) ack 62 win 0
13:58:03.823930 172.16.171.19.32770 > 172.16.171.13.telnet: R 80:80(0) ack 62 win 0

```

```
13:58:03.823930 172.16.171.19.32770 > 172.16.171.13.telnet: R 80:80(0) ack 62 win 0
13:58:03.823930 172.16.171.19.32770 > 172.16.171.13.telnet: R 80:80(0) ack 62 win 0
```

Software Upgrades

This section helps in troubleshooting software upgrades. It contains the following topics:

- [Which Updates to Apply and Their Prerequisites, page C-51](#)
- [Issues With Automatic Update, page C-51](#)
- [Updating a Sensor with the Update Stored on the Sensor, page C-52](#)

Which Updates to Apply and Their Prerequisites

You must have the correct service pack and minor and major version of the software. If you are having trouble with applying new software, make sure that you are applying the proper updates with the proper prerequisites:

- Signature updates require the minimum version and engine version listed in the filename.
- Engine updates require the major or minor version in the engine update filename. Service packs require the correct minor version.
- Minor versions require the correct major version.
- Major versions require the previous major version.

For More Information

To understand how to interpret the IPS software filenames, see [IPS Software Versioning, page 20-3](#).

Issues With Automatic Update

The following list provides suggestions for troubleshooting automatic updates:

- Run TCPDUMP:
 - Create a service account. **Su** to root and run TCPDUMP on the command and control interface to capture packets between the sensor and the FTP server.
 - Use the **upgrade** command to manually upgrade the sensor.
 - Look at the TCPDUMP output for errors coming back from the FTP server.
- Make sure the sensor is in the correct directory. The directory must be specified correctly. This has caused issues with Windows FTP servers. Sometimes an extra “/” or even two “/” are needed in front of the directory name. To verify this, use the same FTP commands you see in the TCPDUMP output through your own FTP connection.
- You must use the Windows FTP server setup option to emulate UNIX file structure and not MS-DOS file structure.
- If you are using SCP, make sure you have added the SSH host key to the known hosts list.

Try the manual **upgrade** command before attempting the automatic update. If it works with the **upgrade** command and does not work with the automatic update, try the following:

- Determine which IPS software version your sensor has.
- Make sure the passwords are configured for automatic update. Make sure they match the same passwords used for manual update.
- Make sure that the filenames in the FTP server are exactly what you see on Downloads on Cisco.com. This includes capitalization. Some Windows FTP servers allow access to the file with the incorrect capitalization but the sensor ultimately rejects the file because the name has changed.
- If necessary, run TCPDUMP on automatic update. You can compare the successful manual update with the unsuccessful automatic update and troubleshoot from there.

For More Information

- For the procedure for creating the service account, see [Creating the Service Account, page 4-28](#).
- For the procedure for reimaging your sensor, see [Chapter 21, “Upgrading, Downgrading, and Installing System Images.”](#)
- For the procedure for adding hosts to the SSH known hosts list, see [Adding Hosts to the SSH Known Hosts List, page 4-47](#).
- For the procedure for determining the software version, see [Displaying Version Information, page C-79](#).

Updating a Sensor with the Update Stored on the Sensor

You can store the update package in the /var directory on the sensor and update the sensor from there if you need to.

To update the sensor with an update stored on the sensor, follow these steps:

Step 1 Log in to the service account.

Step 2 Obtain the update package file from Cisco.com.

Step 3 FTP or SCP the update file to the sensor /usr/cids/idsRoot/var directory.

Step 4 Set the file permissions:.

```
chmod 644 ips_package_file_name
```

Step 5 Exit the service account.

Step 6 Log in to the sensor using an account with administrator privileges.

Step 7 Store the sensor host key.

```
sensor# configure terminal
sensor(config)# service ssh
sensor(config-ssh)# rsal-keys sensor_ip_address
```

Step 8 Upgrade the sensor.

```
sensor(config)# upgrade scp://service@sensor_ip_address/upgrade/ips_package_file_name
Enter password: *****
Re-enter password: *****
```

For More Information

For the procedure for obtaining Cisco IPS software, see [Obtaining Cisco IPS Software, page 20-1](#).

Troubleshooting the IDM

**Note**

These procedures also apply to the IPS section of the ASDM.

**Note**

After you upgrade any IPS software on your sensor, you must restart the IDM to see the latest software features.

This section contains troubleshooting procedures for the IDM. It contains the following topics:

- [Cannot Launch the IDM - Loading Java Applet Failed, page C-53](#)
- [Cannot Launch the IDM-The Analysis Engine Busy, page C-54](#)
- [The IDM, Remote Manager, or Sensing Interfaces Cannot Access Sensor, page C-54](#)
- [Signatures Not Producing Alerts, page C-55](#)

Cannot Launch the IDM - Loading Java Applet Failed

Symptom The browser displays `Loading Cisco IDM. Please wait ...` At the bottom left corner of the window, `Loading Java Applet Failed` is displayed.

Possible Cause This condition can occur if multiple Java Plug-ins are installed on the machine on which you are launching the IDM.

Recommended Action Clear the Java cache and remove temp files and clear history in the browser you are using. The result is that neither of these plug-ins will be used by default and each applet should use the correct plug-in.

To clear the cache, follow these steps:

-
- Step 1** Close all browser windows.
- Step 2** If you have Java Plug-in 1.3.x installed:
- a. Click **Start > Settings > Control Panel > Java Plug-in 1.3.x**.
 - b. Click the **Advanced** tab.
 - c. Under Java Runtime Environment, select **JRE 1.3.x** from the drop-down menu.
 - d. Click the **Cache** tab.
 - e. Click **Clear**.
- Step 3** If you have Java Plug-in 1.4.x installed:
- a. Click **Start > Settings > Control Panel > Java Plug-in 1.4.x**.
 - b. Click the **Advanced** tab.

- c. Under Java Runtime Environment, select **JRE 1.3.x** from the drop-down menu.
- d. Click the **Cache** tab.
- e. Click the **Browser** tab.
- f. Deselect all browser check boxes.
- g. Click **Clear Cache**.

Step 4 Delete the temp files and clear the history in the browser.

Cannot Launch the IDM-The Analysis Engine Busy

Error Message Error connecting to sensor. Failed to load sensor-errNotAvailable-Analysis Engine is busy. Exiting IDM.

Possible Cause This condition can occur if the Analysis Engine in the sensor is busy getting ready to perform a task and so does not respond to the IDM.

Recommended Action Wait for a while and try again to connect.

The IDM, Remote Manager, or Sensing Interfaces Cannot Access Sensor

If the IDM, a remote manager, or sensing interfaces cannot access the sensor, but you can access the sensor CLI using SSH or Telnet (if enabled), follow these steps:

Step 1 Make sure the network configuration allows access to the web server port that is configured on the sensor:

```
sensor# setup
```

```
--- System Configuration Dialog ---
```

```
At any point you may enter a question mark '?' for help.
User ctrl-c to abort configuration dialog at any prompt.
Default settings are in square brackets '[]'.
```

```
Current Configuration:
```

```
service host
network-settings
host-ip 192.168.1.2/24,192.168.1.1
host-name sensor
telnet-option enabled
access-list 0.0.0.0/0
ftp-timeout 300
no login-banner-text
exit
time-zone-settings
offset 0
standard-time-zone-name UTC
```

```
exit
summertime-option disabled
ntp-option disabled
exit
service web-server
port 443
exit
```

- Step 2** If network devices, such as routers, switches, or firewalls, are between the sensor and the workstation, make sure these devices are configured to allow the workstation to access the sensor web server port. All remote management communication is performed by the sensor web server.
-

For More Information

- For the procedure for enabling and disabling Telnet on the sensor, see [Enabling and Disabling Telnet, page 4-5](#).
- For the procedure for configuring the web server, see [Changing Web Server Settings, page 4-15](#).

Signatures Not Producing Alerts



Caution

You cannot add other actions each time you configure the event actions. You are actually replacing the list of event actions every time you configure it, so make sure you choose Produce Alert every time you configure event actions.

If you are not seeing any alerts when signatures are firing, make sure that you have configured Produce Alert as an event action. For example, if you choose Produce Alert, but later add another event action and do not add Produce Alert to the new configuration, alerts are not sent to the Event Store. To make sure you are getting alerts, check the statistics for the virtual sensor and the Event Store.

For More Information

- For more information about event actions, see [Event Actions, page 8-5](#).
- For the procedure for configuring event actions, see [Assigning Actions to Signatures, page 7-20](#).
- For the procedure for obtaining statistics about the virtual sensor and Event Store, see [Displaying Statistics, page 17-30](#).

Troubleshooting the IME

This section describes troubleshooting tools for the IME, and contains the following sections:

- [Time Synchronization on IME and the Sensor, page C-56](#)
- [Not Supported Error Message, page C-56](#)

Time Synchronization on IME and the Sensor

Symptom The IME displays `No Data Available` on the Events dashboard. A historical query does not return any events; however, events are coming in to the IME and they appear in the real-time event viewer.

Possible Cause The time is not synchronized between the sensor and the IME local server. The IME dashboards use a time relative to the IME local time. If these times are not synchronized, the query does not return any results. When you add a sensor to the IME, it checks for the time synchronization and warns you to correct it if it is in wrong. The IME also displays a clock warning in `Home > Devices > Device List` to warn you about problems with synchronization.

Recommended Action Change the time settings on the sensor or the IME local server. In most cases, the time change is required for the sensor because it is configured with the incorrect or default time.

For More Information

- For more information on time and the sensor, see [Time Sources and the Sensor, page C-15](#).
- For the procedure for changing the time on the sensor, see [Correcting Time on the Sensor, page C-17](#).

Not Supported Error Message

Symptom The IME displays `Not Supported` in the device list table and in some gadgets, and no data is included.

Possible Cause Click **Details** to see an explanation for this message. The IME needs IPS 6.1 or later to obtain certain information. The IME still operates with event monitoring and reporting for IPS 5.0 and later and specific IOS IPS versions, but some functions, such as health information and integrated configuration, are not available.

Recommended Action Upgrade to IPS 6.1 or later.

Troubleshooting the ASA 5500-X IPS SSP



Note

Before troubleshooting the ASA 5500-X IPS SSP, check the Caveats section of the Readme for the software version installed on your sensor to see if you are dealing with a known issue.

This section contains troubleshooting information specific to the ASA 5500-X IPS SSP, and contains the following topics:

- [Health and Status Information, page C-57](#)
- [Failover Scenarios, page C-64](#)
- [The ASA 5500-X IPS SSP and the Normalizer Engine, page C-65](#)
- [The ASA 5500-X IPS SSP and Memory Usage, page C-66](#)

- [The ASA 5500-X IPS SSP and Jumbo Packets, page C-66](#)
- [IPS Reloading Messages, page C-67](#)

Health and Status Information

To see the general health of the ASA 5500-X IPS SSP, use the **show module ips details** command.

```
asa# show module ips details
Getting details from the Service Module, please wait...

Card Type:          IPS 5555 Intrusion Prevention System
Model:              IPS5555
Hardware version:   N/A
Serial Number:      FCH1504V0CW
Firmware version:   N/A
Software version:   7.2(1)E4
MAC Address Range: 503d.e59c.7ca0 to 503d.e59c.7ca0
App. name:          IPS
App. Status:        Up
App. Status Desc:   Normal Operation
App. version:        7.2(1)E4
Data Plane Status:  Up
Status:             Up
License:            IPS Module Enabled perpetual
Mgmt IP addr:       192.168.1.2
Mgmt Network mask: 255.255.255.0
Mgmt Gateway:       192.168.1.1
Mgmt web ports:     443
Mgmt TLS enabled:   true
asa#
```

The output shows that the ASA 5500-X IPS SSP is up. If the status reads `Down`, you can reset it using the **sw-module module 1 reset** command.

If you have problems with reimaging the ASA 5500-X IPS SSP, use the **debug module-boot** command to see the output as it boots. Make sure you have the correct IP address for the TFTP server and you have the correct file on the TFTP server. Then use the **sw-module module ips recover** command again to reimage the module.

```
asa-ips# sw-module module ips recover configure image
disk0:/IPS-SSP_5555-K9-sys-1.1-a-7.2-1-E4.aip
Image URL [tftp://192.0.2.1/IPS-5545-K9-sys-1.1-a-7.2-1-E4.aip]:
Port IP Address [192.0.2.226]:
VLAN ID [0]:
Gateway IP Address [192.0.2.254]:

asa-ips# debug module-boot
debug module-boot enabled at level 1
asa-ips# sw-module module ips reload

Reload module ips? [confirm]
Reload issued for module ips.
asa-ips# Mod-ips 228> ***
Mod-ips 229> *** EVENT: The module is reloading.
Mod-ips 230> *** TIME: 08:07:36 CST Jan 17 2012
Mod-ips 231> ***
Mod-ips 232> Mod-ips 233> The system is going down NOW!
Mod-ips 234> Sending SIGTERM to all processes
Mod-ips 235> Sending SIGKILL to all processes
Mod-ips 236> Requesting system reboot
Mod-ips 237> e1000 0000:00:07:0: PCI INT A disabled
```

```

Mod-ips 238> e1000 0000:00:06.0: PCI INT A disabled
Mod-ips 239> e1000 0000:00:05.0: PCI INT A disabled
Mod-ips 240> Restarting system.
Mod-ips 241> machine restart
Mod-ips 242> IVSHMEM: addr = 4093640704 size = 67108864
Mod-ips 243> Booting 'Cisco IPS'
Mod-ips 244> root (hd0,0)
Mod-ips 245> Filesystem type is ext2fs, partition type 0x83
Mod-ips 246> kernel /ips-2.6.1d ro initfsDev=/dev/hda1 init=loader.run rootrw=/dev/hda2
init
Mod-ips 247> fs=runtime-image.cpio.bz2 hda=nodma console=ttyS0 plat=saleen htlblow=1
hugepag
Mod-ips 248> es=3223
Mod-ips 249> [Linux-bzImage, setup=0x2c00, size=0x2bad80]
Mod-ips 250> Linux version 2.6.29.1 (ipsbuild@seti-teambuilder-a) (gcc version 4.3.2
(crosstool
Mod-ips 251> -NG-1.4.1) ) #56 SMP Tue Dec 6 00:46:11 CST 2011
Mod-ips 252> Command line: ro initfsDev=/dev/hda1 init=loader.run rootrw=/dev/hda2
initfs=runti
Mod-ips 253> me-image.cpio.bz2 hda=nodma console=ttyS0 plat=saleen htlblow=1
hugepages=3223
Mod-ips 254> KERNEL supported cpus:
Mod-ips 255> Intel GenuineIntel
Mod-ips 256> AMD AuthenticAMD
Mod-ips 257> Centaur CentaurHauls
Mod-ips 258> BIOS-provided physical RAM map:
Mod-ips 259> BIOS-e820: 0000000000000000 - 000000000009f400 (usable)
Mod-ips 260> BIOS-e820: 000000000009f400 - 00000000000a0000 (reserved)
Mod-ips 261> BIOS-e820: 00000000000f0000 - 0000000000100000 (reserved)
Mod-ips 262> BIOS-e820: 0000000000100000 - 00000000dffffd00 (usable)
Mod-ips 263> BIOS-e820: 00000000dffffd00 - 00000000e0000000 (reserved)
Mod-ips 264> BIOS-e820: 00000000ffffbc000 - 0000000100000000 (reserved)
Mod-ips 265> BIOS-e820: 0000000100000000 - 0000000201400000 (usable)
Mod-ips 266> DMI 2.4 present.
Mod-ips 267> last_pfn = 0x201400 max_arch_pfn = 0x100000000
Mod-ips 268> last_pfn = 0xdffffd max_arch_pfn = 0x100000000
Mod-ips 269> init_memory_mapping: 0000000000000000-00000000dffffd00
Mod-ips 270> last_map_addr: dffffd00 end: dffffd00
Mod-ips 271> init_memory_mapping: 0000000100000000-0000000201400000
Mod-ips 272> last_map_addr: 201400000 end: 201400000
Mod-ips 273> ACPI: RSDP 000F88D0, 0014 (r0 BOCHS )
Mod-ips 274> ACPI: RSDT DFFFFDD0, 0034 (r1 BOCHS BXPCRSDT 1 BXPC 1)
Mod-ips 275> ACPI: FACP DFFFFD90, 0074 (r1 BOCHS BXPCFACP 1 BXPC 1)
Mod-ips 276> FADT: X_PM1a_EVT_BLK.bit_width (16) does not match PM1_EVT_LEN (4)
Mod-ips 277> ACPI: DSDT DFFFFDF10, 1E22 (r1 BXPC BXDSDT 1 INTL 20090123)
Mod-ips 278> ACPI: FACS DFFFFD40, 0040
Mod-ips 279> ACPI: SSDT DFFFFDE90, 0079 (r1 BOCHS BXPCSSDT 1 BXPC 1)
Mod-ips 280> ACPI: APIC DFFFFDD80, 0090 (r1 BOCHS BXPCAPIC 1 BXPC 1)
Mod-ips 281> ACPI: HPET DFFFFDD40, 0038 (r1 BOCHS BXPCHPET 1 BXPC 1)
Mod-ips 282> No NUMA configuration found
Mod-ips 283> Faking a node at 0000000000000000-0000000201400000
Mod-ips 284> Bootmem setup node 0 0000000000000000-0000000201400000
Mod-ips 285> NODE_DATA [0000000000011000 - 000000000001ffff]
Mod-ips 286> bootmap [0000000000020000 - 000000000006027f] pages 41
Mod-ips 287> (6 early reservations) ==> bootmem [0000000000 - 0201400000]
Mod-ips 288> #0 [00000000000 - 0000001000] BIOS data page ==> [0000000000 - 0000001000]
Mod-ips 289> #1 [0000006000 - 0000008000] TRAMPOLINE ==> [0000006000 - 0000008000]
Mod-ips 290> #2 [0000200000 - 0000d55754] TEXT DATA BSS ==> [0000200000 - 0000d55754]
Mod-ips 291> #3 [000009f400 - 0000100000] BIOS reserved ==> [000009f400 - 0000100000]
Mod-ips 292> #4 [0000008000 - 000000c000] PGTABLE ==> [0000008000 - 000000c000]
Mod-ips 293> #5 [000000c000 - 0000011000] PGTABLE ==> [000000c000 - 0000011000]
Mod-ips 294> found SMP MP-table at [ffff8800000f8920] 000f8920
Mod-ips 295> Zone PFN ranges:
Mod-ips 296> DMA 0x00000000 -> 0x00001000

```

```

Mod-ips 297> DMA32 0x00001000 -> 0x00100000
Mod-ips 298> Normal 0x00100000 -> 0x00201400
Mod-ips 299> Movable zone start PFN for each node
Mod-ips 300> early_node_map[3] active PFN ranges
Mod-ips 301> 0: 0x00000000 -> 0x0000009f
Mod-ips 302> 0: 0x00000100 -> 0x000dffff
Mod-ips 303> 0: 0x00100000 -> 0x00201400
Mod-ips 304> ACPI: PM-Timer IO Port: 0xb008
Mod-ips 305> ACPI: LAPIC (acpi_id[0x00] lapic_id[0x00] enabled)
Mod-ips 306> ACPI: LAPIC (acpi_id[0x01] lapic_id[0x01] enabled)
Mod-ips 307> ACPI: LAPIC (acpi_id[0x02] lapic_id[0x02] enabled)
Mod-ips 308> ACPI: LAPIC (acpi_id[0x03] lapic_id[0x03] enabled)
Mod-ips 309> ACPI: LAPIC (acpi_id[0x04] lapic_id[0x04] enabled)
Mod-ips 310> ACPI: LAPIC (acpi_id[0x05] lapic_id[0x05] enabled)
Mod-ips 311> ACPI: IOAPIC (id[0x06] address[0xfec00000] gsi_base[0])
Mod-ips 312> IOAPIC[0]: apic_id 6, version 0, address 0xfec00000, GSI 0-23
Mod-ips 313> ACPI: INT_SRC_OVR (bus 0 bus_irq 5 global_irq 5 high level)
Mod-ips 314> ACPI: INT_SRC_OVR (bus 0 bus_irq 9 global_irq 9 high level)
Mod-ips 315> ACPI: INT_SRC_OVR (bus 0 bus_irq 10 global_irq 10 high level)
Mod-ips 316> ACPI: INT_SRC_OVR (bus 0 bus_irq 11 global_irq 11 high level)
Mod-ips 317> Using ACPI (MADT) for SMP configuration information
Mod-ips 318> ACPI: HPET id: 0x8086a201 base: 0xfed00000
Mod-ips 319> SMP: Allowing 6 CPUs, 0 hotplug CPUs
Mod-ips 320> Allocating PCI resources starting at e2000000 (gap: e0000000:1ffbc000)
Mod-ips 321> NR_CPUS:32 nr_cpumask_bits:32 nr_cpu_ids:6 nr_node_ids:1
Mod-ips 322> PERCPU: Allocating 49152 bytes of per cpu data
Mod-ips 323> Built 1 zonelists in Zone order, mobility grouping on. Total pages: 1939347
Mod-ips 324> Policy zone: Normal
Mod-ips 325> Kernel command line: ro initfsDev=/dev/hda1 init=loader.run rootrw=/dev/hda2
initf
Mod-ips 326> s=runtime-image.cpio.bz2 hda=nodma console=ttyS0 plat=saleen htlblow=1
hugepages=3
Mod-ips 327> 223
Mod-ips 328> hugetlb_lowmem_setup: Allocated 2097152 huge pages (size=0x200000) from
lowmem are
Mod-ips 329> a at 0xffff88002ee00000 phys addr 0x000000002ee00000
Mod-ips 330> Initializing CPU#0
Mod-ips 331> PID hash table entries: 4096 (order: 12, 32768 bytes)
Mod-ips 332> Fast TSC calibration using PIT
Mod-ips 333> Detected 2792.965 MHz processor.
Mod-ips 334> Console: colour VGA+ 80x25
Mod-ips 335> console [ttyS0] enabled
Mod-ips 336> Checking aperture...
Mod-ips 337> No AGP bridge found
Mod-ips 338> PCI-DMA: Using software bounce buffering for IO (SWIOTLB)
Mod-ips 339> Placing 64MB software IO TLB between ffff880020000000 - ffff880024000000
Mod-ips 340> software IO TLB at phys 0x20000000 - 0x24000000
Mod-ips 341> Memory: 7693472k/8409088k available (3164k kernel code, 524688k absent,
190928k re
Mod-ips 342> served, 1511k data, 1032k init)
Mod-ips 343> Calibrating delay loop (skipped), value calculated using timer frequency..
5585.93
Mod-ips 344> BogoMIPS (lpj=2792965)
Mod-ips 345> Dentry cache hash table entries: 1048576 (order: 11, 8388608 bytes)
Mod-ips 346> Inode-cache hash table entries: 524288 (order: 10, 4194304 bytes)
Mod-ips 347> Mount-cache hash table entries: 256
Mod-ips 348> CPU: L1 I cache: 32K, L1 D cache: 32K
Mod-ips 349> CPU: L2 cache: 4096K
Mod-ips 350> CPU 0/0x0 -> Node 0
Mod-ips 351> Freeing SMP alternatives: 29k freed
Mod-ips 352> ACPI: Core revision 20081204
Mod-ips 353> Setting APIC routing to flat
Mod-ips 354> ..TIMER: vector=0x30 apic1=0 pin1=0 apic2=-1 pin2=-1
Mod-ips 355> CPU0: Intel QEMU Virtual CPU version 0.12.5 stepping 03

```

```

Mod-ips 356> Booting processor 1 APIC 0x1 ip 0x6000
Mod-ips 357> Initializing CPU#1
Mod-ips 358> Calibrating delay using timer specific routine.. 5585.16 BogoMIPS
(lpj=2792581)
Mod-ips 359> CPU: L1 I cache: 32K, L1 D cache: 32K
Mod-ips 360> CPU: L2 cache: 4096K
Mod-ips 361> CPU 1/0x1 -> Node 0
Mod-ips 362> CPU1: Intel QEMU Virtual CPU version 0.12.5 stepping 03
Mod-ips 363> checking TSC synchronization [CPU#0 -> CPU#1]:
Mod-ips 364> Measured 1453783140569731 cycles TSC warp between CPUs, turning off TSC
clock.
Mod-ips 365> Marking TSC unstable due to check_tsc_sync_source failed
Mod-ips 366> Booting processor 2 APIC 0x2 ip 0x6000
Mod-ips 367> Initializing CPU#2
Mod-ips 368> Calibrating delay using timer specific routine.. 5580.51 BogoMIPS
(lpj=2790259)
Mod-ips 369> CPU: L1 I cache: 32K, L1 D cache: 32K
Mod-ips 370> CPU: L2 cache: 4096K
Mod-ips 371> CPU 2/0x2 -> Node 0
Mod-ips 372> CPU2: Intel QEMU Virtual CPU version 0.12.5 stepping 03
Mod-ips 373> Booting processor 3 APIC 0x3 ip 0x6000
Mod-ips 374> Initializing CPU#3
Mod-ips 375> Calibrating delay using timer specific routine.. 5585.18 BogoMIPS
(lpj=2792594)
Mod-ips 376> CPU: L1 I cache: 32K, L1 D cache: 32K
Mod-ips 377> CPU: L2 cache: 4096K
Mod-ips 378> CPU 3/0x3 -> Node 0
Mod-ips 379> CPU3: Intel QEMU Virtual CPU version 0.12.5 stepping 03
Mod-ips 380> Booting processor 4 APIC 0x4 ip 0x6000
Mod-ips 381> Initializing CPU#4
Mod-ips 382> Calibrating delay using timer specific routine.. 5585.15 BogoMIPS
(lpj=2792579)
Mod-ips 383> CPU: L1 I cache: 32K, L1 D cache: 32K
Mod-ips 384> CPU: L2 cache: 4096K
Mod-ips 385> CPU 4/0x4 -> Node 0
Mod-ips 386> CPU4: Intel QEMU Virtual CPU version 0.12.5 stepping 03
Mod-ips 387> Booting processor 5 APIC 0x5 ip 0x6000
Mod-ips 388> Initializing CPU#5
Mod-ips 389> Calibrating delay using timer specific routine.. 5585.21 BogoMIPS
(lpj=2792609)
Mod-ips 390> CPU: L1 I cache: 32K, L1 D cache: 32K
Mod-ips 391> CPU: L2 cache: 4096K
Mod-ips 392> CPU 5/0x5 -> Node 0
Mod-ips 393> CPU5: Intel QEMU Virtual CPU version 0.12.5 stepping 03
Mod-ips 394> Brought up 6 CPUs
Mod-ips 395> Total of 6 processors activated (33507.17 BogoMIPS).
Mod-ips 396> net_namespace: 1312 bytes
Mod-ips 397> Booting paravirtualized kernel on bare hardware
Mod-ips 398> NET: Registered protocol family 16
Mod-ips 399> ACPI: bus type pci registered
Mod-ips 400> dca service started, version 1.8
Mod-ips 401> PCI: Using configuration type 1 for base access
Mod-ips 402> mtrr: your CPUs had inconsistent variable MTRR settings
Mod-ips 403> mtrr: your CPUs had inconsistent MTRRdefType settings
Mod-ips 404> mtrr: probably your BIOS does not setup all CPUs.
Mod-ips 405> mtrr: corrected configuration.
Mod-ips 406> bio: create slab <bio-0> at 0
Mod-ips 407> ACPI: Interpreter enabled
Mod-ips 408> ACPI: (supports S0 S5)
Mod-ips 409> ACPI: Using IOAPIC for interrupt routing
Mod-ips 410> ACPI: No dock devices found.
Mod-ips 411> ACPI: PCI Root Bridge [PCI0] (0000:00)
Mod-ips 412> pci 0000:00:01.3: quirk: region b000-b03f claimed by PIIX4 ACPI
Mod-ips 413> pci 0000:00:01.3: quirk: region b100-b10f claimed by PIIX4 SMB

```



```
Mod-ips 414> IVSHMEM: addr = 4093640704 size = 67108864
Mod-ips 415> ACPI: PCI Interrupt Link [LNKA] (IRQs 5 *10 11)
Mod-ips 416> ACPI: PCI Interrupt Link [LNKB] (IRQs 5 *10 11)
Mod-ips 417> ACPI: PCI Interrupt Link [LNKC] (IRQs 5 10 *11)
Mod-ips 418> ACPI: PCI Interrupt Link [LNKD] (IRQs 5 10 *11)
Mod-ips 419> SCSI subsystem initialized
Mod-ips 420> usbcore: registered new interface driver usbfs
Mod-ips 421> usbcore: registered new interface driver hub
Mod-ips 422> usbcore: registered new device driver usb
Mod-ips 423> PCI: Using ACPI for IRQ routing
Mod-ips 424> pnp: PnP ACPI init
Mod-ips 425> ACPI: bus type pnp registered
Mod-ips 426> pnp: PnP ACPI: found 9 devices
Mod-ips 427> ACPI: ACPI bus type pnp unregistered
Mod-ips 428> NET: Registered protocol family 2
Mod-ips 429> IP route cache hash table entries: 262144 (order: 9, 2097152 bytes)
Mod-ips 430> TCP established hash table entries: 524288 (order: 11, 8388608 bytes)
Mod-ips 431> TCP bind hash table entries: 65536 (order: 8, 1048576 bytes)
Mod-ips 432> TCP: Hash tables configured (established 524288 bind 65536)
Mod-ips 433> TCP reno registered
Mod-ips 434> NET: Registered protocol family 1
Mod-ips 435> Adding htlb page ffff88002ee00000 phys 000000002ee00000 page fffffe20000a41000
Mod-ips 436> HugeTLB registered 2 MB page size, pre-allocated 3223 pages
Mod-ips 437> report_hugepages: Using 1 pages from low memory at ffff88002ee00000 HugeTLB
FS
Mod-ips 438> msgmni has been set to 15026
Mod-ips 439> alg: No test for stdrng (krng)
Mod-ips 440> io scheduler noop registered
Mod-ips 441> io scheduler anticipatory registered
Mod-ips 442> io scheduler deadline registered
Mod-ips 443> io scheduler cfq registered (default)
Mod-ips 444> pci 0000:00:00.0: Limiting direct PCI/PCI transfers
Mod-ips 445> pci 0000:00:01.0: PIIX3: Enabling Passive Release
Mod-ips 446> pci 0000:00:01.0: Activating ISA DMA hang workarounds
Mod-ips 447> pci_hotplug: PCI Hot Plug PCI Core version: 0.5
Mod-ips 448> pciehp: PCI Express Hot Plug Controller Driver version: 0.4
Mod-ips 449> acpiphp: ACPI Hot Plug PCI Controller Driver version: 0.5
Mod-ips 450> acpiphp_glue: can't get bus number, assuming 0
Mod-ips 451> decode_hpp: Could not get hotplug parameters. Use defaults
Mod-ips 452> acpiphp: Slot [1] registered
Mod-ips 453> acpiphp: Slot [2] registered
Mod-ips 454> acpiphp: Slot [3] registered
Mod-ips 455> acpiphp: Slot [4] registered
Mod-ips 456> acpiphp: Slot [5] registered
Mod-ips 457> acpiphp: Slot [6] registered
Mod-ips 458> acpiphp: Slot [7] registered
Mod-ips 459> acpiphp: Slot [8] registered
Mod-ips 460> acpiphp: Slot [9] registered
Mod-ips 461> acpiphp: Slot [10] registered
Mod-ips 462> acpiphp: Slot [11] registered
Mod-ips 463> acpiphp: Slot [12] registered
Mod-ips 464> acpiphp: Slot [13] registered
Mod-ips 465> acpiphp: Slot [14] registered
Mod-ips 466> acpiphp: Slot [15] registered
Mod-ips 467> acpiphp: Slot [16] registered
Mod-ips 468> acpiphp: Slot [17] registered
Mod-ips 469> acpiphp: Slot [18] registered
Mod-ips 470> acpiphp: Slot [19] registered
Mod-ips 471> acpiphp: Slot [20] registered
Mod-ips 472> acpiphp: Slot [21] registered
Mod-ips 473> acpiphp: Slot [22] registered
Mod-ips 474> acpiphp: Slot [23] registered
Mod-ips 475> acpiphp: Slot [24] registered
Mod-ips 476> acpiphp: Slot [25] registered
```

```

Mod-ips 477> acpiphp: Slot [26] registered
Mod-ips 478> acpiphp: Slot [27] registered
Mod-ips 479> acpiphp: Slot [28] registered
Mod-ips 480> acpiphp: Slot [29] registered
Mod-ips 481> acpiphp: Slot [30] registered
Mod-ips 482> acpiphp: Slot [31] registered
Mod-ips 483> shpchp: Standard Hot Plug PCI Controller Driver version: 0.4
Mod-ips 484> fakephp: Fake PCI Hot Plug Controller Driver
Mod-ips 485> fakephp: pci_hp_register failed with error -16
Mod-ips 486> fakephp: pci_hp_register failed with error -16
Mod-ips 487> fakephp: pci_hp_register failed with error -16
Mod-ips 488> fakephp: pci_hp_register failed with error -16
Mod-ips 489> fakephp: pci_hp_register failed with error -16
Mod-ips 490> fakephp: pci_hp_register failed with error -16
Mod-ips 491> fakephp: pci_hp_register failed with error -16
Mod-ips 492> processor ACPI_CPU:00: registered as cooling_device0
Mod-ips 493> processor ACPI_CPU:01: registered as cooling_device1
Mod-ips 494> processor ACPI_CPU:02: registered as cooling_device2
Mod-ips 495> processor ACPI_CPU:03: registered as cooling_device3
Mod-ips 496> processor ACPI_CPU:04: registered as cooling_device4
Mod-ips 497> processor ACPI_CPU:05: registered as cooling_device5
Mod-ips 498> hpet_acpi_add: no address or irqs in _CRS
Mod-ips 499> Non-volatile memory driver v1.3
Mod-ips 500> Linux agpgart interface v0.103
Mod-ips 501> ipmi message handler version 39.2
Mod-ips 502> ipmi device interface
Mod-ips 503> IPMI System Interface driver.
Mod-ips 504> ipmi_si: Unable to find any System Interface(s)
Mod-ips 505> IPMI SMB Interface driver
Mod-ips 506> IPMI Watchdog: driver initialized
Mod-ips 507> Copyright (C) 2004 MontaVista Software - IPMI Powerdown via sys_reboot.
Mod-ips 508> Serial: 8250/16550 driver, 4 ports, IRQ sharing enabled
Mod-ips 509> ?serial8250: ttyS0 at I/O 0x3f8 (irq = 4) is a 16550A
Mod-ips 510> serial8250: ttyS1 at I/O 0x2f8 (irq = 3) is a 16550A
Mod-ips 511> 00:06: ttyS0 at I/O 0x3f8 (irq = 4) is a 16550A
Mod-ips 512> 00:07: ttyS1 at I/O 0x2f8 (irq = 3) is a 16550A
Mod-ips 513> brd: module loaded
Mod-ips 514> loop: module loaded
Mod-ips 515> lpc: version 0.1 (Nov 10 2011)
Mod-ips 516> tun: Universal TUN/TAP device driver, 1.6
Mod-ips 517> tun: (C) 1999-2004 Max Krasnyansky <maxk@qualcomm.com>
Mod-ips 518> Uniform Multi-Platform E-IDE driver
Mod-ips 519> piix 0000:00:01.1: IDE controller (0x8086:0x7010 rev 0x00)
Mod-ips 520> piix 0000:00:01.1: not 100native mode: will probe irqs later
Mod-ips 521>     ide0: BM-DMA at 0xc000-0xc007
Mod-ips 522>     ide1: BM-DMA at 0xc008-0xc00f
Mod-ips 523> hda: QEMU HARDDISK, ATA DISK drive
Mod-ips 524> Clocksource tsc unstable (delta = 2851415955127 ns)
Mod-ips 525> hda: MWDMA2 mode selected
Mod-ips 526> hdc: QEMU DVD-ROM, ATAPI CD/DVD-ROM drive
Mod-ips 527> hdc: MWDMA2 mode selected
Mod-ips 528> ide0 at 0x1f0-0x1f7,0x3f6 on irq 14
Mod-ips 529> ide1 at 0x170-0x177,0x376 on irq 15
Mod-ips 530> ide_generic: please use "probe_mask=0x3f" module parameter for probing all
legacy
Mod-ips 531> ISA IDE ports
Mod-ips 532> ide-gd driver 1.18
Mod-ips 533> hda: max request size: 512KiB
Mod-ips 534> hda: 7815168 sectors (4001 MB) w/256KiB Cache, CHS=7753/255/63
Mod-ips 535> hda: cache flushes supported
Mod-ips 536> hda: hda1 hda2 hda3 hda4
Mod-ips 537> Driver 'sd' needs updating - please use bus_type methods
Mod-ips 538> Driver 'sr' needs updating - please use bus_type methods
Mod-ips 539> ehci_hcd: USB 2.0 'Enhanced' Host Controller (EHCI) Driver

```

```
Mod-ips 540> ohci_hcd: USB 1.1 'Open' Host Controller (OHCI) Driver
Mod-ips 541> uhci_hcd: USB Universal Host Controller Interface driver
Mod-ips 542> Initializing USB Mass Storage driver...
Mod-ips 543> usbcore: registered new interface driver usb-storage
Mod-ips 544> USB Mass Storage support registered.
Mod-ips 545> PNP: PS/2 Controller [PNP0303:KBD,PNP0f13:MOU] at 0x60,0x64 irq 1,12
Mod-ips 546> serio: i8042 KBD port at 0x60,0x64 irq 1
Mod-ips 547> serio: i8042 AUX port at 0x60,0x64 irq 12
Mod-ips 548> mice: PS/2 mouse device common for all mice
Mod-ips 549> rtc_cmos 00:01: rtc core: registered rtc_cmos as rtc0
Mod-ips 550> rtc0: alarms up to one day, 114 bytes nvram
Mod-ips 551> input: AT Translated Set 2 keyboard as /class/input/input0
Mod-ips 552> i2c /dev entries driver
Mod-ips 553> piix4_smbus 0000:00:01.3: SMBus Host Controller at 0xb100, revision 0
Mod-ips 554> device-mapper: ioctl: 4.14.0-ioctl (2008-04-23) initialised:
dm-devel@redhat.com
Mod-ips 555> cpuidle: using governor ladder
Mod-ips 556> usbcore: registered new interface driver usbbhid
Mod-ips 557> usbbhid: v2.6:USB HID core driver
Mod-ips 558> TCP cubic registered
Mod-ips 559> IPv6: Loaded, but is disabled by default. IPv6 may be enabled on individual
interf
Mod-ips 560> aces.
Mod-ips 561> NET: Registered protocol family 10
Mod-ips 562> NET: Registered protocol family 17
Mod-ips 563> NET: Registered protocol family 5
Mod-ips 564> rtc_cmos 00:01: setting system clock to 2012-01-17 14:06:34 UTC (1326809194)
Mod-ips 565> Freeing unused kernel memory: 1032k freed
Mod-ips 566> Write protecting the kernel read-only data: 4272k
Mod-ips 567> Loader init started...
Mod-ips 568> kjournald starting. Commit interval 5 seconds
Mod-ips 569> EXT3-fs: mounted filesystem with ordered data mode.
Mod-ips 570> input: ImExPS/2 Generic Explorer Mouse as /class/input/input1
Mod-ips 571> 51216 blocks
Mod-ips 572> Checking rootrw fs: corrected filesystem
Mod-ips 573> kjournald starting. Commit interval 5 seconds
Mod-ips 574> EXT3 FS on hda2, internal journal
Mod-ips 575> EXT3-fs: mounted filesystem with ordered data mode.
Mod-ips 576> mkdir: cannot create directory '/lib/modules': File exists
Mod-ips 577> init started: BusyBox v1.13.1 (2011-11-01 07:21:34 CDT)
Mod-ips 578> starting pid 678, tty '': '/etc/init.d/rc.init'
Mod-ips 579> Checking system fs: no errors
Mod-ips 580> kjournald starting. Commit interval 5 seconds
Mod-ips 581> EXT3-fs: mounted filesystem with ordered data mode.
Mod-ips 582> /etc/init.d/rc.init: line 102: /proc/sys/vm/bdflush: No such file or
directory
Mod-ips 583> starting pid 728, tty '': '/etc/init.d/rcS'
Mod-ips 584> Initializing random number generator... done.
Mod-ips 585> Starting network... ifconfig lo 127.0.0.1 netmask 255.255.255.255 up
Mod-ips 586> starting inetd
Mod-ips 587> done
Mod-ips 588> Starting sshd:
Mod-ips 589> Starting nscd:
Mod-ips 590> Set Irq Affinity ... cpus:
Mod-ips 591> Checking kernel allocated memory: EXT3 FS on hda1, internal journal
Mod-ips 592> [ OK ]
Mod-ips 593> Unloading REGEX-CP drivers ...
Mod-ips 594> Loading REGEX-CP drivers ...
Mod-ips 595> ACPI: PCI Interrupt Link [LNKD] enabled at IRQ 11
Mod-ips 596> cpp_user_kvm 0000:00:04.0: PCI INT A -> Link[LNKD] -> GSI 11 (level, high) ->
IRQ
Mod-ips 597> 11
Mod-ips 598> Detected cpp_user_kvm device with 33554432 bytes of shared memory
Mod-ips 599> Device 0: model=LCPX8640, cpc=T2005, cpe0=None, cpe1=None
```

```

Mod-ips 600> Load cidmodcap:
Mod-ips 601> Create node:
Mod-ips 602> ln: /etc/modprobe.conf: File exists
Mod-ips 603> Shutting down network... ifconfig lo down
Mod-ips 604> ifconfig lo down
Mod-ips 605> done
Mod-ips 606> Load ihm:
Mod-ips 607> Create node:
Mod-ips 608> Load kvm_ivshmem: IVSHMEM: writing 0x0 to 0xc86cf8
Mod-ips 609> IVSHMEM: IntrMask write(w) val = 0xffff
Mod-ips 610> Create node:
Mod-ips 611> Create node:
Mod-ips 612> Create node:
Mod-ips 613> Set Irq Affinity ... cpus: 6
Mod-ips 614> Starting network... ifconfig lo 127.0.0.1 netmask 255.255.255.255 up
Mod-ips 615> done
Mod-ips 616> Creating boot.info[ OK ]
Mod-ips 617> Checking for system modifications since last boot[ OK ]
Mod-ips 618> Checking model identification[ OK ]
Mod-ips 619> Model: ASA-5555
Mod-ips 620> Model=ASA-5555
Mod-ips 621> Unable to set speed and duplex for user mode interfaces
Mod-ips 622> interface type 0x8086:0x100e at pci address 0:6.0(0) is currently named eth1
Mod-ips 623> Renaming eth1 --> ma0_0
Mod-ips 624> interface type 0x8086:0x100e at pci address 0:7.0(0) is currently named po0_0
Mod-ips 625> interface type 0x8086:0x100e at pci address 0:5.0(0) is currently named eth0
Mod-ips 626> Renaming eth0 --> sy0_0
Mod-ips 627> Initializing access list
Mod-ips 628> MGMT_INTFC_CIDS_NAME Management0/0
Mod-ips 629> MGMT_INTFC_OS_NAME ma0_0
Mod-ips 630> SYSTEM_PCI_IDS 0x0030,0x0028
Mod-ips 631> Load rebootkom:
Mod-ips 632> root: Starting SSM controlplane
Mod-ips 633> Starting CIDS:
Mod-ips 634> starting pid 1718, tty '/dev/ttyS0': '/sbin/getty -L ttyS0 9600 vt100'

```

Failover Scenarios

The following failover scenarios apply to the ASA 5500-X series in the event of configuration changes, signature/signature engine updates, service packs, and SensorApp crashes on the ASA 5500-X IPS SSP.

Single ASA in Fail-Open Mode

- If the ASA is configured in fail-open mode for the ASA 5500-X IPS SSP, and the ASA 5500-X IPS SSP experiences a configuration change or signature/signature engine update, traffic is passed through the ASA without being inspected.
- If the ASA is configured in fail-open mode for the ASA 5500-X IPS SSP, and the ASA 5500-X IPS SSP experiences a SensorApp crash or a service pack upgrade, traffic is passed through the ASA without being inspected.

Single ASA in Fail-Close Mode

- If the ASA is configured in fail-close mode for the ASA 5500-X IPS SSP, and the ASA 5500-X IPS SSP experiences a configuration change or a signature/signature engine update, traffic is stopped from passing through the ASA.
- If the ASA is configured in fail-close mode for the ASA 5500-X IPS SSP, and the ASA 5500-X IPS SSP experiences a SensorApp crash or a service pack upgrade, traffic is stopped from passing through the ASA.

Two ASAs in Fail-Open Mode

- If the ASAs are configured in fail-open mode and if the ASA 5500-X IPS SSP on the active ASA experiences a configuration change or a signature/signature engine update, traffic is still passed through the active ASA without being inspected. Failover is not triggered.
- If the ASAs are configured in fail-open mode, and if the ASA 5500-X IPS SSP on the active ASA experiences a SensorApp crash or a service pack upgrade, failover is triggered and traffic passes through the ASA 5500-X IPS SSP that was previously the standby ASA 5500-X IPS SSP.

Two ASAs in Fail-Close Mode

- If the ASAs are configured in fail-close mode, and if the ASA 5500-X IPS SSP on the active ASA experiences a configuration change or a signature/signature engine update, traffic is stopped from passing through the active ASA. No failover is triggered.
- If the ASAs are configured in fail-close mode, and if the ASA 5500-X IPS SSP on the active ASA experiences a SensorApp crash or a service pack upgrade, failover is triggered and traffic passes through the ASA 5500-X IPS SSP that was previously the standby for the ASA 5500-X IPS SSP.

Configuration Examples

Use the following configuration for the primary ASA:

```
interface GigabitEthernet0/7
  description LAN Failover Interface

failover
failover lan unit primary
failover lan interface folink GigabitEthernet0/7
failover interface ip folink 172.27.48.1 255.255.255.0 standby 172.27.48.2
```

Use the following configuration for the secondary ASA:

```
interface GigabitEthernet0/7
  description LAN Failover Interface

failover
failover lan unit secondary
failover lan interface folink GigabitEthernet0/7
failover interface ip folink 172.27.48.1 255.255.255.0 standby 172.27.48.2
```

The ASA 5500-X IPS SSP and the Normalizer Engine

The majority of the features in the Normalizer engine are not used on the ASA 5500-X IPS SSP, because the ASA itself handles the normalization. Packets on the ASA IPS modules go through a special path in the Normalizer that only reassembles fragments and puts packets in the right order for the TCP stream. The Normalizer does not do any of the normalization that is done on an inline IPS appliance, because that causes problems in the way the ASA handles the packets.

The following Normalizer engine signatures are not supported:

- 1300.0
- 1304.0
- 1305.0
- 1307.0
- 1308.0
- 1309.0

- 1311.0
- 1315.0
- 1316.0
- 1317.0
- 1330.0
- 1330.1
- 1330.2
- 1330.9
- 1330.10
- 1330.12
- 1330.14
- 1330.15
- 1330.16
- 1330.17
- 1330.18

The ASA 5500-X IPS SSP and Memory Usage

For the ASA 5500-X IPS SSP, the memory usage is 93%. The default health thresholds for the sensor are 80% for yellow and 91% for red, so the sensor health will be shown as red on these platforms even for normal operating conditions. You can tune the threshold percentage for memory usage so that it reads more accurately for these platforms by configuring the **memory-usage-policy** option in the sensor health metrics.



Note

Make sure you have the **memory-usage-policy** option in the sensor health metrics enabled.

Table C-3 lists the yellow-threshold and the red-threshold health values.

Table C-3 ASA 5500-X IPS SSP Memory Usage Values

Platform	Yellow	Red	Memory Used
ASA 5512-X IPS SSP	85%	91%	28%
ASA 5515-X IPS SSP	88%	92%	14%
ASA 5525-X IPS SSP	88%	92%	14%
ASA 5545-X IPS SSP	93%	96%	13%
ASA 5555-X IPS SSP	95%	98%	17%

The ASA 5500-X IPS SSP and Jumbo Packets

The jumbo packet count in the **show interface** command output from the lines `Total Jumbo Packets Received` and `Total Jumbo Packets Transmitted` for ASA IPS modules may be larger than expected due to some packets that were almost jumbo size on the wire being counted as jumbo size by the IPS.

This miscount is a result of header bytes added to the packet by the ASA before the packet is transmitted to the IPS. For IPv4, 58 bytes of header data are added. For IPv6, 78 bytes of header data are added. The ASA removes the added IPS header before the packet leaves the ASA.

IPS Reloading Messages

The following messages generated during some IPS signature and global correlation updates for IPS 7.1 and later on the ASA 5500-X IPS SSP can cause confusion since the IPS is not reloading:

```
ASA5585-SSP-IPS20 Module in slot 1, application up "IPS", version "7.1(1)E4" Normal
Operation
ASA5585-SSP-IPS20 Module in slot 1, application reloading "IPS", version "7.1(1)E4" Config
Change
```

These messages are generated during some, but not all, of the global correlation updates that are attempted every five minutes. This is expected behavior. There is a global correlation check every five minutes, but there may not be an update available, thus the message appears every hour or so. When a global correlation update actually takes place, a message is sent from the IPS to the ASA indicating that a configuration change is taking place.

Troubleshooting the ASA 5585-X IPS SSP



Note

Before troubleshooting the ASA 5585-X IPS SSP, check the Caveats section of the Readme for the software version installed on your sensor to see if you are dealing with a known issue.

This section contains troubleshooting information specific to the ASA 5585-X IPS SSP, and contains the following topics:

- [Health and Status Information, page C-67](#)
- [Failover Scenarios, page C-70](#)
- [Traffic Flow Stopped on IPS Switchports, page C-72](#)
- [The ASA 5585-X IPS SSP and the Normalizer Engine, page C-72](#)
- [The ASA 5585-X IPS SSP and Jumbo Packets, page C-73](#)
- [IPS Reloading Messages, page C-73](#)

Health and Status Information

To see the general health of the ASA 5585-X IPS SSP, use the **show module 1 details** command.

```
asa# show module 1 details
Getting details from the Service Module, please wait...
ASA 5585-X IPS Security Services Processor-20 with 8GE
Model:                ASA5585-SSP-IPS20
Hardware version:     1.0
Serial Number:        ABC1234DEFG
Firmware version:     2.0(1)3
Software version:     7.2(1)E4
MAC Address Range:    8843.e12f.5414 to 8843.e12f.541f
App. name:            IPS
```

```

App. Status:      Up
App. Status Desc: Normal Operation
App. version:     7.2(1)E4
Data plane Status: Up
Status:          Up
Mgmt IP addr:    192.0.2.3
Mgmt Network mask: 255.255.255.0
Mgmt Gateway:    192.0.2.254
Mgmt Access List: 10.0.0.0/8
Mgmt Access List: 64.0.0.0/8
Mgmt web ports:  443
Mgmt TLS enabled true
asa

```

The output shows that the ASA 5585-X IPS SSP is up. If the status reads *Down*, you can reset it using the **hw-module module 1 reset** command.

```

asa# hw-module module 1 reset
The module in slot 1 should be shut down before
resetting it or loss of configuration may occur.
Reset module in slot 1? [confirm]
Reset issued for module in slot 1
asa# show module 1 details
Getting details from the Service Module, please wait...
Unable to read details from slot 1
ASA 5585-X IPS Security Services Processor-20 with 8GE
Model:          ASA5585-SSP-IPS20
Hardware version: 1.0
Serial Number:  ABC1234DEFG
Firmware version: 2.0(7)0
Software version: 7.2(1)E4
MAC Address Range: 5475.d029.7f9c to 5475.d029.7fa7
App. name:      IPS
App. Status:    Not Applicable
App. Status Desc: Not Applicable
App. version:   7.2(1)E4
Data plane Status: Not Applicable
Status:        Shutting Down
asa# show module 1 details
Getting details from the Service Module, please wait...
Unable to read details from slot 1
ASA 5585-X IPS Security Services Processor-20 with 8GE
Model:          ASA5585-SSP-IPS20
Hardware version: 1.0
Serial Number:  ABC1234DEFG
Firmware version: 2.0(7)0
Software version: 7.2(1)E4
MAC Address Range: 5475.d029.7f9c to 5475.d029.7fa7
App. name:      IPS
App. Status:    Not Applicable
App. Status Desc: Not Applicable
App. version:   7.2(1)E4
Data plane Status: Not Applicable
Status:        Down
asa# show module 1 details
Getting details from the Service Module, please wait...
Unable to read details from slot 1
ASA 5585-X IPS Security Services Processor-20 with 8GE
Model:          ASA5585-SSP-IPS20
Hardware version: 1.0
Serial Number:  ABC1234DEFG
Firmware version: 2.0(7)0
Software version: 7.2(1)E4
MAC Address Range: 5475.d029.7f9c to 5475.d029.7fa7

```



```

App. name:          IPS
App. Status:       Not Applicable
App. Status Desc:  Not Applicable
App. version:      7.2(1)E4
Data plane Status: Not Applicable
Status:           Init
asa# show module 1 details
Getting details from the Service Module, please wait...
ASA 5585-X IPS Security Services Processor-20 with 8GE
Model:            ASA5585-SSP-IPS20
Hardware version: 1.0
Serial Number:    ABC1234DEFG
Firmware version: 2.0(7)0
Software version: 7.2(1)E4
MAC Address Range: 5475.d029.7f9c to 5475.d029.7fa7
App. name:       IPS
App. Status:     Reload
App. Status Desc: Starting up
App. version:    7.2(1)E4
Data plane Status: Down
Status:         Up
Mgmt IP addr:   192.0.2.3
Mgmt Network mask: 255.255.255.0
Mgmt Gateway:   192.0.2.254
Mgmt Access List: 0.0.0.0/0
Mgmt web ports: 443
Mgmt TLS enabled: true
asa# show module 1 details
Getting details from the Service Module, please wait...
ASA 5585-X IPS Security Services Processor-20 with 8GE
Model:            ASA5585-SSP-IPS20
Hardware version: 1.0
Serial Number:    ABC1234DEFG
Firmware version: 2.0(7)0
Software version: 7.2(1)E4
MAC Address Range: 5475.d029.7f9c to 5475.d029.7fa7
App. name:       IPS
App. Status:     Up
App. Status Desc: Normal Operation
App. version:    7.2(1)E4
Data plane Status: Up
Status:         Up
Mgmt IP addr:   192.0.2.3
Mgmt Network mask: 255.255.255.0
Mgmt Gateway:   192.0.2.254
Mgmt Access List: 0.0.0.0/0
Mgmt web ports: 443
Mgmt TLS enabled: true
asa#

```

If you have problems with reimaging the ASA 5585-X IPS SSP, use the **debug module-boot** command to see the output as it boots. Make sure you have the correct IP address for the TFTP server and you have the correct file on the TFTP server. Then use the **hw-module module 1 recover** command again to reimage the module.

```

ips-ssp# hw-module module 1 recover configure
Image URL [tftp://0.0.0.0/]: tftp://10.10.10.10//IPS-SSP_20-K9-sys-1.1-a-7.2-1-E4.img
Port IP Address [0.0.0.0]: 10.10.10.11
VLAN ID [0]:
Gateway IP Address [0.0.0.0]: 10.10.10.254

asa(config)# debug module-boot
debug module-boot enabled at level 1

```

```

asa(config)# hw-module module 1 recover boot
The module in slot 1 will be recovered. This may erase all configuration and all data on
that device and attempt to download a new image for it.
Recover module in slot 1? [confirm]
Recover issued for module in slot 1
asa(config)# Slot-1 140> Cisco Systems ROMMON Version (1.0(10)0) #0: Fri Mar 25 23:02:10
PST 2010
Slot-1 141> Platform ASA5585-SSP-IPS20
Slot-1 142> GigabitEthernet0/0
Slot-1 143> Link is UP
Slot-1 144> MAC Address: 000b.fcf8.0176
Slot-1 145> ROMMON Variable Settings:
Slot-1 146> ADDRESS=192.0.2.3
Slot-1 147> SERVER=192.0.2.15
Slot-1 148> GATEWAY=192.0.2.254
Slot-1 149> PORT=GigabitEthernet0/0
Slot-1 150> VLAN=untagged
Slot-1 151> IMAGE=IPS-SSP-K9-sys-1.1-a-7.2-1.img
Slot-1 152> CONFIG=
Slot-1 153> LINKTIMEOUT=20
Slot-1 154> PKTTIMEOUT=4
Slot-1 155> RETRY=20
Slot-1 156> tftp IPS-SSP_10-K9-sys-1.1-a-7.2-1.img@192.0.2.15 via 192.0.2.254
Slot-1 157> TFTP failure: Packet verify failed after 20 retries
Slot-1 158> Rebooting due to Autoboot error ...
Slot-1 159> Rebooting...
Slot-1 160> Cisco Systems ROMMON Version (1.0(10)0) #0: Fri Mar 25 23:02:10 PST 2010
Slot-1 161> Platform ASA5585-SSP-IPS20
Slot-1 162> GigabitEthernet0/0
Slot-1 163> Link is UP
Slot-1 164> MAC Address: 000b.fcf8.0176
Slot-1 165> ROMMON Variable Settings:
Slot-1 166> ADDRESS=192.0.2.3
Slot-1 167> SERVER=192.0.2.15
Slot-1 168> GATEWAY=192.0.2.254
Slot-1 169> PORT=GigabitEthernet0/0
Slot-1 170> VLAN=untagged
Slot-1 171> IMAGE=IPS-SSP_10-K9-sys-1.1-a-7.2-1.img
Slot-1 172> CONFIG=
Slot-1 173> LINKTIMEOUT=20
Slot-1 174> PKTTIMEOUT=4
Slot-1 175> RETRY=20
Slot-1 176> tftp IPS-SSP_10-K9-sys-1.1-a-7.2-1.img@192.0.2.15 via 192.0.2.254

```

Failover Scenarios

The following failover scenarios apply to the ASA 5585-X in the event of configuration changes, signature/signature engine updates, service packs, and SensorApp crashes on the ASA 5585-X IPS SSP.

Single ASA 5585-X in Fail-Open Mode

- If the ASA is configured in fail-open mode for the ASA 5585-X IPS SSP, and the ASA 5585-X IPS SSP experiences a configuration change or signature/signature engine update, traffic is passed through the ASA without being inspected.
- If the ASA is configured in fail-open mode for the ASA 5585-X IPS SSP, and the ASA 5585-X IPS SSP experiences a SensorApp crash or a service pack upgrade, traffic is passed through the ASA without being inspected.

Single ASA 5585-X in Fail-Close Mode

- If the ASA is configured in fail-close mode for the ASA 5585-X IPS SSP, and the ASA 5585-X IPS SSP experiences a configuration change or a signature/signature engine update, traffic is stopped from passing through the ASA.
- If the ASA is configured in fail-close mode for the ASA 5585-X IPS SSP, and the ASA 5585-X IPS SSP experiences a SensorApp crash or a service pack upgrade, traffic is stopped from passing through the ASA.

Two ASA 5585-Xs in Fail-Open Mode

- If the ASAs are configured in fail-open mode and if the ASA 5585-X IPS SSP on the active ASA experiences a configuration change or a signature/signature engine update, traffic is still passed through the active ASA without being inspected. Failover is not triggered.
- If the ASAs are configured in fail-open mode, and if the ASA 5585-X IPS SSP on the active ASA experiences a SensorApp crash or a service pack upgrade, failover is triggered and traffic passes through the ASA 5585-X IPS SSP that was previously the standby ASA 5585-X IPS SSP.

Two ASA 5585-Xs in Fail-Close Mode

- If the ASAs are configured in fail-close mode, and if the ASA 5585-X IPS SSP on the active ASA experiences a configuration change or a signature/signature engine update, traffic is stopped from passing through the active ASA. No failover is triggered.
- If the ASAs are configured in fail-close mode, and if the ASA 5585-X IPS SSP on the active ASA experiences a SensorApp crash or a service pack upgrade, failover is triggered and traffic passes through the ASA 5585-X IPS SSP that was previously the standby for the ASA 5585-X IPS SSP.

Configuration Examples

Use the following configuration for the primary ASA:

```
interface GigabitEthernet0/7
  description LAN Failover Interface

failover
failover lan unit primary
failover lan interface folink GigabitEthernet0/7
failover interface ip folink 172.27.48.1 255.255.255.0 standby 172.27.48.2
```

Use the following configuration for the secondary ASA:

```
interface GigabitEthernet0/7
  description LAN Failover Interface

failover
failover lan unit secondary
failover lan interface folink GigabitEthernet0/7
failover interface ip folink 172.27.48.1 255.255.255.0 standby 172.27.48.2
```

Traffic Flow Stopped on IPS Switchports

Problem Traffic on any port located on the ASA 5585-X IPS SSP (1/x) no longer passes through the adaptive security appliance when the ASA 5585-X IPS SSP is reset or shut down. This affects all traffic through these ports regardless of whether or not the traffic would have been monitored by the IPS. The link on the ports will link down when the ASA 5585-X IPS SSP is reset or shut down.

Possible Cause Using the ports located on the ASA 5585-X IPS SSP (1/x), and resetting or shutting it down via any mechanism.

Use the ports on the adaptive security appliance (0/x) instead because those ports do not lose their link when the ASA 5585-X IPS SSP is reset or shut down.

The ASA 5585-X IPS SSP and the Normalizer Engine

The majority of the features in the Normalizer engine are not used on the ASA 5585-X IPS SSP, because the ASA itself handles the normalization. Packets on the ASA IPS modules go through a special path in the Normalizer that only reassembles fragments and puts packets in the right order for the TCP stream. The Normalizer does not do any of the normalization that is done on an inline IPS appliance, because that causes problems in the way the ASA handles the packets.

The following Normalizer engine signatures are not supported:

- 1300.0
- 1304.0
- 1305.0
- 1307.0
- 1308.0
- 1309.0
- 1311.0
- 1315.0
- 1316.0
- 1317.0
- 1330.0
- 1330.1
- 1330.2
- 1330.9
- 1330.10
- 1330.12
- 1330.14
- 1330.15
- 1330.16
- 1330.17
- 1330.18

The ASA 5585-X IPS SSP and Jumbo Packets

The jumbo packet count in the **show interface** command output from the lines `Total Jumbo Packets Received` and `Total Jumbo Packets Transmitted` for ASA IPS modules may be larger than expected due to some packets that were almost jumbo size on the wire being counted as jumbo size by the IPS. This miscount is a result of header bytes added to the packet by the ASA before the packet is transmitted to the IPS. For IPv4, 58 bytes of header data are added. For IPv6, 78 bytes of header data are added. The ASA removes the added IPS header before the packet leaves the ASA.

IPS Reloading Messages

The following messages generated during some IPS signature and global correlation updates for IPS 7.1 and later on the ASA 5585-X IPS SSP can cause confusion since the IPS is not reloading:

```
ASA5585-SSP-IPS20 Module in slot 1, application up "IPS", version "7.1(1)E4" Normal
Operation
ASA5585-SSP-IPS20 Module in slot 1, application reloading "IPS", version "7.1(1)E4" Config
Change
```

These messages are generated during some, but not all, of the global correlation updates that are attempted every five minutes. This is expected behavior. There is a global correlation check every five minutes, but there may not be an update available, thus the message appears every hour or so. When a global correlation update actually takes place, a message is sent from the IPS to the ASA indicating that a configuration change is taking place.

Gathering Information

You can use the following CLI commands and scripts to gather information and diagnose the state of the sensor when problems occur. You can use the **show tech-support** command to gather all the information of the sensor, or you can use the other individual commands listed in this section for specific information. This section contains the following topics:

- [Health and Network Security Information, page C-73](#)
- [Tech Support Information, page C-74](#)
- [Version Information, page C-79](#)
- [Statistics Information, page C-82](#)
- [Interfaces Information, page C-94](#)
- [Events Information, page C-98](#)
- [cidDump Script, page C-102](#)
- [Uploading and Accessing Files on the Cisco FTP Site, page C-103](#)

Health and Network Security Information



Caution

When the sensor is first starting, it is normal for certain health metric statuses to be red until the sensor is fully up and running.

**Note**

The ASA 5500-X IPS SSP and ASA 5585-X IPS SSP do not support bypass mode. The adaptive security appliance will either fail open, fail close, or fail over depending on the configuration of the adaptive security appliance and the type of activity being done on the IPS.

Use the **show health** command in privileged EXEC mode to display the overall health status information of the sensor. The health status categories are rated by red and green with red being critical.

To display the overall health status of the sensor, follow these steps:

- Step 1** Log in to the CLI.
- Step 2** Show the health and security status of the sensor.

```

sensor# show health
Overall Health Status                               Red
Health Status for Failed Applications                Green
Health Status for Signature Updates                  Green
Health Status for License Key Expiration             Red
Health Status for Running in Bypass Mode             Green
Health Status for Interfaces Being Down              Red
Health Status for the Inspection Load                Green
Health Status for the Time Since Last Event Retrieval Green
Health Status for the Number of Missed Packets       Green
Health Status for the Memory Usage                   Not Enabled
Health Status for Global Correlation                 Red
Health Status for Network Participation              Not Enabled

Security Status for Virtual Sensor vs0               Green
sensor#

```

Tech Support Information

The **show tech-support** command is useful for capturing all sensor status and configuration information. This section describes the **show tech-support** command, and contains the following topics:

- [Understanding the show tech-support Command, page C-74](#)
- [Displaying Tech Support Information, page C-75](#)
- [Tech Support Command Output, page C-76](#)

Understanding the show tech-support Command

**Note**

The `/var/log/messages` file is persistent across reboots and the information is displayed in the output of the `show tech-support` command.

**Note**

The **show tech-support** command displays historical interface data for each interface for the past 72 hours.

The **show tech-support** command captures all status and configuration information on the sensor and includes the current configuration, version information, and cidDump information. The output can be large, over 1 MB. You can transfer the output to a remote system. For the procedure for copying the output to a remote system, see [Displaying Tech Support Information, page C-75](#).



Note Always run the **show tech-support** command before contacting TAC.

Displaying Tech Support Information



Note The `/var/log/messages` file is persistent across reboots and the information is displayed in the output of the `show tech-support` command.



Note The **show tech-support** command displays historical interface data for each interface for the past 72 hours.

Use the **show tech-support [page] [destination-url destination_url]** command to display system information on the screen or have it sent to a specific URL. You can use the information as a troubleshooting tool with the TAC.

The following parameters are optional:

- **page**—Displays the output, one page of information at a time. Press **Enter** to display the next line of output or use the spacebar to display the next page of information.
- **destination-url**—Indicates the information should be formatted as HTML and sent to the destination that follows this command. If you use this keyword, the output is not displayed on the screen.
- *destination_url*—Indicates the information should be formatted as HTML. The URL specifies where the information should be sent. If you do not use this keyword, the information is displayed on the screen.
- You can specify the following destination types:
 - **ftp**—Destination URL for FTP network server. The syntax for this prefix is:
`ftp://[[username@location]/relativeDirectory]/filename` OR
`ftp://[[username@location]//absoluteDirectory]/filename`
 - **scp**—Destination URL for the SCP network server. The syntax for this prefix is:
`scp://[[username@]location]/relativeDirectory]/filename` OR
`scp://[[username@]location]//absoluteDirectory]/filename`

Varlog Files

The `/var/log/messages` file has the latest logs. A new softlink called `varlog` has been created under the `/usr/cids/idsRoot/log` folder that points to the `/var/log/messages` file. Old logs are stored in `varlog.1` and `varlog.2` files. The maximum size of these `varlog` files is 200 KB. Once they cross the size limit the content is rotated. The content of `varlog`, `varlog.1`, and `varlog.2` is displayed in the output of the **show tech-support** command.

Displaying Tech Support Information

To display tech support information, follow these steps:

-
- Step 1** Log in to the CLI using an account with administrator privileges.
- Step 2** View the output on the screen. The system information appears on the screen, one page at a time. Press the spacebar to view the next page or press **Ctrl-C** to return to the prompt

```
sensor# show tech-support page
```

- Step 3** To send the output (in HTML format) to a file:

- a. Enter the following command, followed by a valid destination. The `password:` prompt appears.

```
sensor# show tech-support destination-url destination_url
```

Example

To send the tech support output to the file `/absolute/reports/sensor1Report.html`:

```
sensor# show tech support dest
ftp://csidsuser@10.2.1.2//absolute/reports/sensor1Report.html
```

- b. Enter the password for this user account. The `Generating report:` message is displayed.
-

Tech Support Command Output

The following is an example of the `show tech-support` command output:



Note

This output example shows the first part of the command and lists the information for the interfaces, authentication, and the Analysis Engine.

```
sensor# show tech-support page
System Status Report
This Report was generated on Thu Oct 17 14:35:52 2013.
Output from show version
Application Partition:

Cisco Intrusion Prevention System, Version 7.3(1)E4

Host:
  Realm Keys                key1.0
Signature Definition:
  Signature Update           S741.0          2013-09-10
  Threat Profile Version    2
OS Version:                 2.6.29.1
Platform:                   IPS-4360
Serial Number:              FGL1702401M
Licensed, expires:         21-Nov-2014 UTC
Sensor up-time is 22:09.
Using 14372M out of 15943M bytes of available memory (90% usage)
system is using 32.4M out of 160.0M bytes of available disk space (20% usage)
application-data is using 85.6M out of 376.4M bytes of available disk space (24% usage)
boot is using 63.1M out of 70.2M bytes of available disk space (95% usage)
application-log is using 494.0M out of 513.0M bytes of available disk space (96% usage)
```



```

MainApp          C-2013_12_16_14_00_7_3_0_143  (Release)  2013-12-16T14:06:20-0600
Running
AnalysisEngine  C-2013_12_16_14_00_7_3_0_143  (Release)  2013-12-16T14:06:20-0600
Running
CollaborationApp C-2013_12_16_14_00_7_3_0_143  (Release)  2013-12-16T14:06:20-0600
Running
CLI              C-2013_12_16_14_00_7_3_0_143  (Release)  2013-12-16T14:06:20-0600

```

Upgrade History:

```
IPS-K9-7.3-1-E4  11:22:07 UTC Sat Jan 19 2013
```

```
Recovery Partition Version 1.1 - 7.3(1)E4
```

```
Host Certificate Valid from: 09-Oct-2014 to 09-Oct-2016
```

Output from show interfaces

Interface Statistics

```

Total Packets Received = 141568
Total Bytes Received = 13807286
Missed Packet Percentage = 0
Current Bypass Mode = Off

```

MAC statistics from interface GigabitEthernet0/0

```
Interface function = Sensing interface
```

```
Description =
```

```
Media Type = TX
```

```
Default VLAN = 0
```

```
Inline Mode = Unpaired
```

```
Pair Status = N/A
```

```
Hardware Bypass Capable = No
```

```
Hardware Bypass Paired = N/A
```

```
Link Status = Down
```

```
Admin Enabled Status = Disabled
```

```
Link Speed = N/A
```

```
Link Duplex = N/A
```

```
Missed Packet Percentage = 0
```

```
Total Packets Received = 0
```

```
Total Bytes Received = 0
```

```
Total Multicast Packets Received = 0
```

```
Total Broadcast Packets Received = 0
```

```
Total Jumbo Packets Received = 0
```

```
Total Undersize Packets Received = 0
```

```
Total Receive Errors = 0
```

```
Total Receive FIFO Overruns = 0
```

```
Total Packets Transmitted = 0
```

```
Total Bytes Transmitted = 0
```

```
Total Multicast Packets Transmitted = 0
```

```
Total Broadcast Packets Transmitted = 0
```

```
Total Jumbo Packets Transmitted = 0
```

```
Total Undersize Packets Transmitted = 0
```

```
Total Transmit Errors = 0
```

```
Total Transmit FIFO Overruns = 0
```

MAC statistics from interface Management0/0

```
Interface function = Command-control interface
```

```
Description =
```

```
Media Type = TX
```

```
Default VLAN = 0
```

```
Link Status = Up
```

```
Link Speed = Auto_100
```

```
Link Duplex = Auto_Full
```

```
Total Packets Received = 428857
```

```
Total Bytes Received = 29967463
```

```
Total Multicast Packets Received = 0
```

```

Total Receive Errors = 0
Total Receive FIFO Overruns = 0
Total Packets Transmitted = 5303
Total Bytes Transmitted = 1040574
Total Transmit Errors = 0
Total Transmit FIFO Overruns = 0
MAC statistics from interface GigabitEthernet0/1
Interface function = Sensing interface
Description =
Media Type = TX
Default VLAN = 0
Inline Mode = Unpaired
Pair Status = N/A
Hardware Bypass Capable = No
Hardware Bypass Paired = N/A
Link Status = Down
Admin Enabled Status = Disabled
Link Speed = N/A
Link Duplex = N/A
Missed Packet Percentage = 0
Total Packets Received = 0
Total Bytes Received = 0
Total Multicast Packets Received = 0
Total Broadcast Packets Received = 0
Total Jumbo Packets Received = 0
Total Undersize Packets Received = 0
Total Receive Errors = 0
Total Receive FIFO Overruns = 0
Total Packets Transmitted = 0
Total Bytes Transmitted = 0
Total Multicast Packets Transmitted = 0
Total Broadcast Packets Transmitted = 0
Total Jumbo Packets Transmitted = 0
Total Undersize Packets Transmitted = 0
Total Transmit Errors = 0
Total Transmit FIFO Overruns = 0
MAC statistics from interface GigabitEthernet0/2
Interface function = Sensing interface
Description =
Media Type = TX
Default VLAN = 0
Inline Mode = Unpaired
Pair Status = N/A
Hardware Bypass Capable = No
Hardware Bypass Paired = N/A
Link Status = Down
Admin Enabled Status = Disabled
Link Speed = N/A
Link Duplex = N/A
Missed Packet Percentage = 0
Total Packets Received = 0
Total Bytes Received = 0
Total Multicast Packets Received = 0
Total Broadcast Packets Received = 0
Total Jumbo Packets Received = 0
Total Undersize Packets Received = 0
Total Receive Errors = 0
Total Receive FIFO Overruns = 0
Total Packets Transmitted = 0
Total Bytes Transmitted = 0
Total Multicast Packets Transmitted = 0
Total Broadcast Packets Transmitted = 0
Total Jumbo Packets Transmitted = 0
Total Undersize Packets Transmitted = 0

```

```

Total Transmit Errors = 0
Total Transmit FIFO Overruns = 0
MAC statistics from interface GigabitEthernet0/3
Interface function = Sensing interface
Description =
Media Type = TX
Default VLAN = 0
Inline Mode = Unpaired
Pair Status = N/A
Hardware Bypass Capable = No
Hardware Bypass Paired = N/A
Link Status = Down
Admin Enabled Status = Disabled
Link Speed = N/A
Link Duplex = N/A
Missed Packet Percentage = 0
Total Packets Received = 0
Total Bytes Received = 0
Total Multicast Packets Received = 0
Total Broadcast Packets Received = 0
Total Jumbo Packets Received = 0
Total Undersize Packets Received = 0
Total Receive Errors = 0
--MORE--

```

Version Information

The **show version** command is useful for obtaining sensor information. This section describes the **show version** command, and contains the following topics:

- [Understanding the show version Command, page C-79](#)
- [Displaying Version Information, page C-79](#)

Understanding the show version Command

The **show version** command shows the basic sensor information and can indicate where a failure is occurring. It gives the following information:

- Which applications are running
- Versions of the applications
- Disk and memory usage
- Upgrade history of the applications



Note

To get the same information from IDM, choose **Monitoring > Sensor Monitoring > Support Information > Diagnostics Report**. To get the same information from IME, choose **Configuration > sensor_name > Sensor Monitoring > Support Information > Diagnostics Report**.

Displaying Version Information

Use the **show version** command to display version information for all installed operating system packages, signature packages, and IPS processes running on the system. To view the configuration for the entire system, use the **more current-config** command.

**Note**

The CLI output is an example of what your configuration may look like. It will not match exactly due to the optional setup choices, sensor model, and IPS version you have installed.

**Note**

For the IPS 4500 series sensors, the **show version** command output contains an extra application called the SwitchApp.

To display the version and configuration, follow these steps:

Step 1 Log in to the CLI.

Step 2 View version information.

```
sensor# show version
Application Partition:
```

```
Cisco Intrusion Prevention System, Version 7.3(1)E4
```

```
Host:
```

```
  Realm Keys          key1.0
```

```
Signature Definition:
```

```
  Signature Update    S741.0          2013-09-10
```

```
  Threat Profile Version 2
```

```
OS Version:          2.6.29.1
```

```
Platform:            IPS-4360
```

```
Serial Number:       FGL1702401M
```

```
Licensed, expires:   21-Nov-2014 UTC
```

```
Sensor up-time is 22:09.
```

```
Using 14372M out of 15943M bytes of available memory (90% usage)
```

```
system is using 32.4M out of 160.0M bytes of available disk space (20% usage)
```

```
application-data is using 85.6M out of 376.4M bytes of available disk space (24% usage)
```

```
boot is using 63.1M out of 70.2M bytes of available disk space (95% usage)
```

```
application-log is using 494.0M out of 513.0M bytes of available disk space (96% usage)
```

```
MainApp              C-2013_12_16_14_00_7_3_0_143  (Release)  2013-12-16T14:06:20-0600
Running
AnalysisEngine       C-2013_12_16_14_00_7_3_0_143  (Release)  2013-12-16T14:06:20-0600
Running
CollaborationApp     C-2013_12_16_14_00_7_3_0_143  (Release)  2013-12-16T14:06:20-0600
Running
CLI                   C-2013_12_16_14_00_7_3_0_143  (Release)  2013-12-16T14:06:20-0600
```

```
Upgrade History:
```

```
  IPS-K9-7.3-1-E4    11:22:07 UTC Sat Jan 19 2013
```

```
Recovery Partition Version 1.1 - 7.3(1)E4
```

```
Host Certificate Valid from: 09-Oct-2014 to 09-Oct-2016
```

```
sensor#
```

**Note**

If the **--MORE--** prompt is displayed, press the spacebar to see more information or **Ctrl-C** to cancel the output and get back to the CLI prompt.

Step 3 View configuration information.**Note** You can use the **more current-config** or **show configuration** commands.

```

sensor# more current-config
! -----
! Current configuration last modified Fri Oct 10 09:41:17 2014
! -----
! Version 7.3(1)
! Host:
!   Realm Keys                key1.0
! Signature Definition:
!   Signature Update          S741.0   2013-09-10
!   Threat Profile Version    2
! -----
service interface
exit
! -----
service authentication
exit
! -----
service event-action-rules rules0
exit
! -----
service host
network-settings
host-ip 10.106.1.4/24,10.106.1.1
host-name sensor
telnet-option enabled
access-list 0.0.0.0/0
dns-primary-server disabled
exit
time-zone-settings
offset 0
standard-time-zone-name UTC
exit
exit
! -----
service logger
exit
! -----
service network-access
exit
! -----
service notification
exit
! -----
service signature-definition sig0
exit
! -----
service ssh-known-hosts
exit
! -----
service trusted-certificates
exit
! -----
service web-server
exit
! -----
service anomaly-detection ad0
exit
! -----

```

```
service external-product-interface
exit
! -----
service health-monitor
exit
! -----
service global-correlation
exit
! -----
service aaa
exit
! -----
service analysis-engine
exit
sensor#
```

Statistics Information

The **show statistics** command is useful for examining the state of the sensor services. This section describes the **show statistics** command, and contains the following topics:

- [Understanding the show statistics Command, page C-82](#)
- [Displaying Statistics, page C-83](#)

Understanding the show statistics Command

The **show statistics** command provides a snapshot of the state of the sensor services. The following services provide statistics:

- AnalysisEngine
- Authentication
- Denied Attackers
- Event Server
- Event Store
- Host
- Logger
- Attack Response (formerly known as Network Access)
- Notification
- SDEE Server
- Transaction Server
- Transaction Source
- Virtual Sensor
- Web Server

**Note**

To get the same information from IDM, choose **Monitoring > Sensor Monitoring > Support Information > Statistics**. To get the same information from IME, choose **Configuration > sensor_name > Sensor Monitoring > Support Information > Statistics**.

Displaying Statistics

Use the **show statistics [analysis-engine | anomaly-detection | authentication | denied-attackers | event-server | event-store | external-product-interface | global-correlation | host | logger | network-access | notification | os-identification | sdee-server | transaction-server | virtual-sensor | web-server] [clear]** command to display statistics for each sensor application.

Use the **show statistics {anomaly-detection | denied-attackers | os-identification | virtual-sensor} [name | clear]** command to display statistics for these components for all virtual sensors. If you provide the virtual sensor name, the statistics for that virtual sensor only are displayed.

**Note**

The **clear** option is not available for the analysis engine, anomaly detection, host, network access, or OS identification applications.

For the IPS 4500 series, at the end of the command output, there are extra details for the Ethernet controller statistics, such as the total number of packets received at the Ethernet controller, the total number of packets dropped at the Ethernet controller under high load conditions, and the total packets transmitted including the customer traffic packets and the internal keepalive packet count.

**Note**

The Ethernet controller statistics are polled at an interval of 5 seconds from the hardware side. The keepalives are sent or updated at an interval of 10 ms. Because of this, there may be a disparity in the actual count reflected in the total packets transmitted. At times, it is even possible that the total packets transmitted may be less than the keepalive packets transmitted.

To display statistics for the sensor, follow these steps:

Step 1 Log in to the CLI.

Step 2 Display the statistics for the Analysis Engine.

```
sensor# show statistics analysis-engine
Analysis Engine Statistics
Number of seconds since service started = 318375
Processing Load Percentage
  Thread    5 sec    1 min    5 min
  0          1         1         1
  1          1         1         1
  2          1         1         1
  3          1         1         1
  Average   1         1         1

The rate of TCP connections tracked per second = 0
The rate of packets per second = 0
The rate of bytes per second = 0
Receiver Statistics
  Total number of packets processed since reset = 0
  Total number of IP packets processed since reset = 0
Transmitter Statistics
  Total number of packets transmitted = 129766
```

```

Total number of packets denied = 0
Total number of packets reset = 0
Fragment Reassembly Unit Statistics
  Number of fragments currently in FRU = 0
  Number of datagrams currently in FRU = 0
TCP Stream Reassembly Unit Statistics
  TCP streams currently in the embryonic state = 0
  TCP streams currently in the established state = 0
  TCP streams currently in the closing state = 0
  TCP streams currently in the system = 0
  TCP Packets currently queued for reassembly = 0
The Signature Database Statistics.
  Total nodes active = 0
  TCP nodes keyed on both IP addresses and both ports = 0
  UDP nodes keyed on both IP addresses and both ports = 0
  IP nodes keyed on both IP addresses = 0
Statistics for Signature Events
  Number of SigEvents since reset = 0
Statistics for Actions executed on a SigEvent
  Number of Alerts written to the IdsEventStore = 0
Inspection Stats

GlobalCorrelationStats
  SwVersion = 7.3(0.119)E4
  SigVersion = 741.0
  DatabaseRecordCount = 0
  DatabaseVersion = 0
  RuleVersion = 0
  ReputationFilterVersion = 0
  AlertsWithHit = 0
  AlertsWithMiss = 0
  AlertsWithModifiedRiskRating = 0
  AlertsWithGlobalCorrelationDenyAttacker = 0
  AlertsWithGlobalCorrelationDenyPacket = 0
  AlertsWithGlobalCorrelationOtherAction = 0
  AlertsWithAuditRepDenies = 0
  ReputationForcedAlerts = 0
  EventStoreInsertTotal = 0
  EventStoreInsertWithHit = 0
  EventStoreInsertWithMiss = 0
  EventStoreDenyFromGlobalCorrelation = 0
  EventStoreDenyFromOverride = 0
  EventStoreDenyFromOverlap = 0
  EventStoreDenyFromOther = 0
  ReputationFilterDataSize = 0
  ReputationFilterPacketsInput = 0
  ReputationFilterRuleMatch = 0
  DenyFilterHitsNormal = 0
  DenyFilterHitsGlobalCorrelation = 0
  SimulatedReputationFilterPacketsInput = 0
  SimulatedReputationFilterRuleMatch = 0
  SimulatedDenyFilterInsert = 0
  SimulatedDenyFilterPacketsInput = 0
  SimulatedDenyFilterRuleMatch = 0
  TcpDeniesDueToGlobalCorrelation = 0
  TcpDeniesDueToOverride = 0
  TcpDeniesDueToOverlap = 0
  TcpDeniesDueToOther = 0
  SimulatedTcpDeniesDueToGlobalCorrelation = 0
  SimulatedTcpDeniesDueToOverride = 0
  SimulatedTcpDeniesDueToOverlap = 0
  SimulatedTcpDeniesDueToOther = 0
  LateStageDenyDueToGlobalCorrelation = 0

```



```

LateStageDenyDueToOverride = 0
LateStageDenyDueToOverlap = 0
LateStageDenyDueToOther = 0
SimulatedLateStageDenyDueToGlobalCorrelation = 0
SimulatedLateStageDenyDueToOverride = 0
SimulatedLateStageDenyDueToOverlap = 0
SimulatedLateStageDenyDueToOther = 0
AlertHistogram
RiskHistogramEarlyStage
RiskHistogramLateStage
ConfigAggressiveMode = 0
ConfigAuditMode = 0
RegexAccelerationStats
Status = Enabled
DriverVersion = 6.3.1
Devices = 1
Agents = 12
Flows = 7
Channels = 0
SubmittedJobs = 193
CompletedJobs = 193
SubmittedBytes = 2836764
CompletedBytes = 0
TCPFlowsWithoutLCB = 0
UDPFlowsWithoutLCB = 0
TCPMissedPacketsDueToUpdate = 0
UDPMissedPacketsDueToUpdate = 0
MemorySize = 1073741824
HostDirectMemSize = 0
MaliciousSiteDenyHitCounts
MaliciousSiteDenyHitCountsAUDIT
Ethernet Controller Statistics
Total Packets Received = 0
Total Received Packets Dropped = 0
Total Packets Transmitted = 13643"
sensor#

```

Step 3 Display the statistics for anomaly detection.

```

sensor# show statistics anomaly-detection
Statistics for Virtual Sensor vs0
No attack
Detection - ON
Learning - ON
Next KB rotation at 10:00:01 UTC Sat Jan 18 2008
Internal Zone
  TCP Protocol
  UDP Protocol
  Other Protocol
External Zone
  TCP Protocol
  UDP Protocol
  Other Protocol
Illegal Zone
  TCP Protocol
  UDP Protocol
  Other Protocol
Statistics for Virtual Sensor vs1
No attack
Detection - ON
Learning - ON
Next KB rotation at 10:00:00 UTC Sat Jan 18 2008
Internal Zone
  TCP Protocol

```

```

        UDP Protocol
        Other Protocol
    External Zone
        TCP Protocol
        UDP Protocol
        Other Protocol
    Illegal Zone
        TCP Protocol
        UDP Protocol
        Other Protocol
sensor#

```

Step 4 Display the statistics for authentication.

```

sensor# show statistics authentication
General
    totalAuthenticationAttempts = 128
    failedAuthenticationAttempts = 0
sensor#

```

Step 5 Display the statistics for the denied attackers in the system.

```

sensor# show statistics denied-attackers
Denied Attackers and hit count for each.
Denied Attackers and hit count for each.
Statistics for Virtual Sensor vs0
    Denied Attackers with percent denied and hit count for each.

    Denied Attackers with percent denied and hit count for each.

Statistics for Virtual Sensor vs1
    Denied Attackers with percent denied and hit count for each.

    Denied Attackers with percent denied and hit count for each.

sensor#

```

Step 6 Display the statistics for the Event Server.

```

sensor# show statistics event-server
General
    openSubscriptions = 0
    blockedSubscriptions = 0
Subscriptions
sensor#

```

Step 7 Display the statistics for the Event Store.

```

sensor# show statistics event-store
EEvent store statistics
    General information about the event store
        The current number of open subscriptions = 2
        The number of events lost by subscriptions and queries = 0
        The number of filtered events not written to the event store = 850763
        The number of queries issued = 0
        The number of times the event store circular buffer has wrapped = 0
    Number of events of each type currently stored
        Status events = 4257
        Shun request events = 0
        Error events, warning = 669
        Error events, error = 8

```

```

Error events, fatal = 0
Alert events, informational = 0
Alert events, low = 0
Alert events, medium = 0
Alert events, high = 0
Alert events, threat rating 0-20 = 0
Alert events, threat rating 21-40 = 0
Alert events, threat rating 41-60 = 0
Alert events, threat rating 61-80 = 0
Alert events, threat rating 81-100 = 0
Cumulative number of each type of event
Status events = 4257
Shun request events = 0
Error events, warning = 669
Error events, error = 8
Error events, fatal = 0
Alert events, informational = 0
Alert events, low = 0
Alert events, medium = 0
Alert events, high = 0
Alert events, threat rating 0-20 = 0
Alert events, threat rating 21-40 = 0
Alert events, threat rating 41-60 = 0
Alert events, threat rating 61-80 = 0
Alert events, threat rating 81-100 = 0
sensor#

```

Step 8 Display the statistics for global correlation.

```

sensor# show statistics global-correlation
Network Participation:
  Counters:
    Total Connection Attempts = 0
    Total Connection Failures = 0
    Connection Failures Since Last Success = 0
  Connection History:
Updates:
  Status Of Last Update Attempt = Disabled
  Time Since Last Successful Update = never
  Counters:
    Update Failures Since Last Success = 0
    Total Update Attempts = 0
    Total Update Failures = 0
  Update Interval In Seconds = 300
  Update Server = update-manifests.ironport.com
  Update Server Address = Unknown
  Current Versions:
Warnings:
  Unlicensed = Global correlation inspection and reputation filtering have been
  disabled because the sensor is unlicensed.
  Action Required = Obtain a new license from http://www.cisco.com/go/license.
sensor#

```

Step 9 Display the statistics for the host.

```

sensor# show statistics host
General Statistics
  Last Change To Host Config (UTC) = 15-Oct-2013 09:11:13
  Command Control Port Device = Management0/0
Network Statistics
  = ma0_0      Link encap:Ethernet  HWaddr 00:06:F6:2B:48:4C
  =            inet addr:10.106.132.150  Bcast:10.106.133.255  Mask:255.255.254.0
  =            UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
  =            RX packets:429575 errors:0 dropped:0 overruns:0 frame:0

```

```

=          TX packets:5492 errors:0 dropped:0 overruns:0 carrier:0
=          collisions:0 txqueuelen:100
=          RX bytes:30017118 (28.6 MiB) TX bytes:1064828 (1.0 MiB)
=          Base address:0xe000 Memory:fb800000-fb820000
NTP Statistics
  status = Not applicable
Memory Usage
  usedBytes = 6630588416
  freeBytes = 1615556608
  totalBytes = 8246145024
CPU Statistics
  Note: CPU Usage statistics are not a good indication of the sensor processing
  load. The Inspection Load Percentage in the output of 'show inspection-load' sh
  ould be used instead.
  Usage over last 5 seconds = 100
  Usage over last minute = 100
  Usage over last 5 minutes = 99
  Usage over last 5 seconds = 2
  Usage over last minute = 1
  Usage over last 5 minutes = 1
  Usage over last 5 seconds = 100
  Usage over last minute = 100
  Usage over last 5 minutes = 100
  Usage over last 5 seconds = 100
  Usage over last minute = 100
  Usage over last 5 minutes = 100
Memory Statistics
  Memory usage (bytes) = 6630588416
  Memory free (bytes) = 1615556608
Auto Update Statistics
  lastDirectoryReadAttempt = N/A
  lastDownloadAttempt = N/A
  lastInstallAttempt = N/A
  nextAttempt = N/A
Auxiliary Processors Installed
sensor#

```

Step 10 Display the statistics for the logging application.

```

sensor# show statistics logger
The number of Log interprocessor FIFO overruns = 0
The number of syslog messages received = 11
The number of <evError> events written to the event store by severity
  Fatal Severity = 0
  Error Severity = 64
  Warning Severity = 35
  TOTAL = 99
The number of log messages written to the message log by severity
  Fatal Severity = 0
  Error Severity = 64
  Warning Severity = 24
  Timing Severity = 311
  Debug Severity = 31522
  Unknown Severity = 7
  TOTAL = 31928
sensor#

```

Step 11 Display the statistics for the ARC.

```

sensor# show statistics network-access
Current Configuration
  LogAllBlockEventsAndSensors = true
  EnableNvramWrite = false
  EnableAclLogging = false

```

```

AllowSensorBlock = false
BlockMaxEntries = 11
MaxDeviceInterfaces = 250
NetDevice
  Type = PIX
  IP = 10.89.150.171
  NATAddr = 0.0.0.0
  Communications = ssh-3des
NetDevice
  Type = PIX
  IP = 192.0.2.4
  NATAddr = 0.0.0.0
  Communications = ssh-3des
NetDevice
  Type = PIX
  IP = 192.0.2.5
  NATAddr = 0.0.0.0
  Communications = telnet
NetDevice
  Type = Cisco
  IP = 192.0.2.6
  NATAddr = 0.0.0.0
  Communications = telnet
  BlockInterface
    InterfaceName = ethernet0/1
    InterfaceDirection = out
    InterfacePostBlock = Post_Acl_Test
  BlockInterface
    InterfaceName = ethernet0/1
    InterfaceDirection = in
    InterfacePreBlock = Pre_Acl_Test
    InterfacePostBlock = Post_Acl_Test
NetDevice
  Type = CAT6000_VACL
  IP = 192.0.2.1
  NATAddr = 0.0.0.0
  Communications = telnet
  BlockInterface
    InterfaceName = 502
    InterfacePreBlock = Pre_Acl_Test
  BlockInterface
    InterfaceName = 507
    InterfacePostBlock = Post_Acl_Test
State
  BlockEnable = true
NetDevice
  IP = 192.0.2.3
  AclSupport = Does not use ACLs
  Version = 6.3
  State = Active
  Firewall-type = PIX
NetDevice
  IP = 192.0.2.7
  AclSupport = Does not use ACLs
  Version = 7.0
  State = Active
  Firewall-type = ASA
NetDevice
  IP = 102.0.2.8
  AclSupport = Does not use ACLs
  Version = 2.2
  State = Active
  Firewall-type = FWSM
NetDevice

```

```

IP = 192.0.2.9
AclSupport = uses Named ACLs
Version = 12.2
State = Active
NetDevice
IP = 192.0.2.10
AclSupport = Uses VACLs
Version = 8.4
State = Active
BlockedAddr
Host
  IP = 203.0.113.1
  Vlan =
  ActualIp =
  BlockMinutes =
Host
  IP = 203.0.113.2
  Vlan =
  ActualIp =
  BlockMinutes =
Host
  IP = 203.0.113.4
  Vlan =
  ActualIp =
  BlockMinutes = 60
  MinutesRemaining = 24
Network
  IP = 203.0.113.9
  Mask = 255.255.0.0
  BlockMinutes =
sensor#

```

Step 12 Display the statistics for the notification application.

```

sensor# show statistics notification
General
  Number of SNMP set requests = 0
  Number of SNMP get requests = 0
  Number of error traps sent = 0
  Number of alert traps sent = 0
sensor#

```

Step 13 Display the statistics for OS identification.

```

sensor# show statistics os-identification
Statistics for Virtual Sensor vs0
  OS Identification
    Configured
    Imported
    Learned
sensor#

```

Step 14 Display the statistics for the SDEE server.

```

sensor# show statistics sdee-server
General
  Open Subscriptions = 1
  Blocked Subscriptions = 1
  Maximum Available Subscriptions = 5
  Maximum Events Per Retrieval = 500
Subscriptions
  sub-4-d074914f
    State = Read Pending
    Last Read Time = 23:54:16 UTC Wed Nov 30 2011
    Last Read Time (nanoseconds) = 1322697256078549000

```

```
sensor#
```

Step 15 Display the statistics for the transaction server.

```
sensor# show statistics transaction-server
General
  totalControlTransactions = 35
  failedControlTransactions = 0
sensor#
```

Step 16 Display the statistics for a virtual sensor.

```
sensor# show statistics virtual-sensor vs0

Name of current Signature-Definition instance = sig0
Name of current Event-Action-Rules instance = rules0
List of interfaces monitored by this virtual sensor =
General Statistics for this Virtual Sensor
  Number of seconds since a reset of the statistics = 318712
  MemoryAlloPercent = 13
  MemoryUsedPercent = 13
  MemoryMaxCapacity = 5500000
  MemoryMaxHighUsed = 750721
  MemoryCurrentAllo = 769921
  MemoryCurrentUsed = 750721
  Inspection Load Percentage = 0
  Total packets processed since reset = 0
  Total IP packets processed since reset = 0
  Total IPv4 packets processed since reset = 0
  Total IPv6 packets processed since reset = 0
  Total IPv6 AH packets processed since reset = 0
  Total IPv6 ESP packets processed since reset = 0
  Total IPv6 Fragment packets processed since reset = 0
  Total IPv6 Routing Header packets processed since reset = 0
  Total IPv6 ICMP packets processed since reset = 0
  Total packets that were not IP processed since reset = 0
  Total TCP packets processed since reset = 0
  Total UDP packets processed since reset = 0
  Total ICMP packets processed since reset = 0
  Total packets that were not TCP, UDP, or ICMP processed since reset = 0
  Total ARP packets processed since reset = 0
  Total ISL encapsulated packets processed since reset = 0
  Total 802.1q encapsulated packets processed since reset = 0
  Total GRE Packets processed since reset = 0
  Total GRE Fragment Packets processed since reset = 0
  Total GRE Packets skipped since reset = 0
  Total GRE Packets with Bad Header skipped since reset = 0
  Total IpIp Packets with Bad Header skipped since reset = 0
  Total Encapsulated Tunnel Packets with Bad Header skipped since reset = 0
  Total packets with bad IP checksums processed since reset = 0
  Total packets with bad layer 4 checksums processed since reset = 0
  Total cross queue TCP packets processed since reset = 0
  Total cross queue UDP packets processed since reset = 0
  HTTP transfer encoding errors = 0
  HTTP content encoding errors = 0
  HTTP character encoding errors = 0
  HTTP connection out of sync = 0
  HTTP pipelining or persistence out of sync = 0
  Total number of bytes processed since reset = 0
  The rate of packets per second since reset = 0
  The rate of bytes per second since reset = 0
  The average bytes per packet since reset = 0
Denied Address Information
  Number of Active Denied Attackers = 0
```

```

Number of Denied Attackers Inserted = 0
Number of Denied Attacker Victim Pairs Inserted = 0
Number of Denied Attacker Service Pairs Inserted = 0
Number of Denied Attackers Total Hits = 0
Number of times max-denied-attackers limited creation of new entry = 0
Number of exec Clear commands during uptime = 0
Denied Attackers and hit count for each.
Denied Attackers with percent denied and hit count for each.

```

AlarmDB Statistics for this Virtual Sensor

```

Total IP nodes keyed on both IP addresses since reset = 0
Active IP nodes keyed on both IP addresses = 0
IP nodes prevented since reset = 0

```

The Signature Database Statistics.

The Number of each type of node active in the system

```

Total nodes active = 0
TCP nodes keyed on both IP addresses and both ports = 0
UDP nodes keyed on both IP addresses and both ports = 0
IP nodes keyed on both IP addresses = 0

```

The number of each type of node inserted since reset

```

Total nodes inserted = 0
TCP nodes keyed on both IP addresses and both ports = 0
UDP nodes keyed on both IP addresses and both ports = 0
IP nodes keyed on both IP addresses = 0

```

The rate of nodes per second for each time since reset

```

Nodes per second = 0
TCP nodes keyed on both IP addresses and both ports per second = 0
UDP nodes keyed on both IP addresses and both ports per second = 0
IP nodes keyed on both IP addresses per second = 0

```

The number of root nodes forced to expire because of memory constraints

```

TCP nodes keyed on both IP addresses and both ports = 0

```

```

Packets dropped because they would exceed Database insertion rate limits = 0

```

Fragment Reassembly Unit Statistics for this Virtual Sensor

```

Number of fragments currently in FRU = 0
Number of datagrams currently in FRU = 0
Number of fragments received since reset = 0
Number of fragments forwarded since reset = 0
Number of fragments dropped since last reset = 0
Number of fragments modified since last reset = 0
Number of complete datagrams reassembled since last reset = 0
Fragments hitting too many fragments condition since last reset = 0
Number of overlapping fragments since last reset = 0
Number of Datagrams too big since last reset = 0
Number of overwriting fragments since last reset = 0
Number of Initial fragment missing since last reset = 0
Fragments hitting the max partial datagrams limit since last reset = 0
Fragments too small since last reset = 0
Too many fragments per datagram limit since last reset = 0
Number of datagram reassembly timeout since last reset = 0
Too many fragments claiming to be the last since last reset = 0
Fragments with bad fragment flags since last reset = 0

```

TCP Normalizer stage statistics

```

Packets Input = 0
Packets Modified = 0
Dropped packets from queue = 0
Dropped packets due to deny-connection = 0
Duplicate Packets = 0
Current Streams = 0
Current Streams Closed = 0
Current Streams Closing = 0
Current Streams Embryonic = 0
Current Streams Established = 0
Current Streams Denied = 0

```



```

Total SendAck Limited Packets = 0
Total SendAck Limited Streams = 0
Total SendAck Packets Sent = 0
Statistics for the TCP Stream Reassembly Unit
Current Statistics for the TCP Stream Reassembly Unit
  TCP streams currently in the embryonic state = 0
  TCP streams currently in the established state = 0
  TCP streams currently in the closing state = 0
  TCP streams currently in the system = 0
  TCP Packets currently queued for reassembly = 0
Cumulative Statistics for the TCP Stream Reassembly Unit since reset
  TCP streams that have been tracked since last reset = 0
  TCP streams that had a gap in the sequence jumped = 0
  TCP streams that was abandoned due to a gap in the sequence = 0
  TCP packets that arrived out of sequence order for their stream = 0
  TCP packets that arrived out of state order for their stream = 0
  The rate of TCP connections tracked per second since reset = 0
SigEvent Preliminary Stage Statistics
  Number of Alerts received = 0
  Number of Alerts Consumed by AlertInterval = 0
  Number of Alerts Consumed by Event Count = 0
  Number of FireOnce First Alerts = 0
  Number of FireOnce Intermediate Alerts = 0
  Number of Summary First Alerts = 0
  Number of Summary Intermediate Alerts = 0
  Number of Regular Summary Final Alerts = 0
  Number of Global Summary Final Alerts = 0
  Number of Active SigEventDataNodes = 0
  Number of Alerts Output for further processing = 0
--MORE--

```

Step 17 Display the statistics for the web server.

```

sensor# show statistics web-server
listener-443
  session-11
    remote host = 64.101.182.167
    session is persistent = no
    number of requests serviced on current connection = 1
    last status code = 200
    last request method = GET
    last request URI = cgi-bin/sdee-server
    last protocol version = HTTP/1.1
    session state = processingGetServlet
  number of server session requests handled = 957134
  number of server session requests rejected = 0
  total HTTP requests handled = 365871
  maximum number of session objects allowed = 40
  number of idle allocated session objects = 12
  number of busy allocated session objects = 1
  summarized log messages
    number of TCP socket failure messages logged = 0
    number of TLS socket failure messages logged = 0
    number of TLS protocol failure messages logged = 0
    number of TLS connection failure messages logged = 595015
    number of TLS crypto warning messages logged = 0
    number of TLS expired certificate warning messages logged = 0
    number of receipt of TLS fatal alert message messages logged = 594969
  crypto library version = 6.2.1.0
sensor#

```

Step 18 Clear the statistics for an application, for example, the logging application. The statistics are retrieved and cleared.

```

sensor# show statistics logger clear
The number of Log interprocessor FIFO overruns = 0
The number of syslog messages received = 141
The number of <evError> events written to the event store by severity
  Fatal Severity = 0
  Error Severity = 14
  Warning Severity = 142
  TOTAL = 156
The number of log messages written to the message log by severity
  Fatal Severity = 0
  Error Severity = 14
  Warning Severity = 1
  Timing Severity = 0
  Debug Severity = 0
  Unknown Severity = 28
  TOTAL = 43

```

Step 19 Verify that the statistics have been cleared. The statistics now all begin from 0.

```

sensor# show statistics logger
The number of Log interprocessor FIFO overruns = 0
The number of syslog messages received = 0
The number of <evError> events written to the event store by severity
  Fatal Severity = 0
  Error Severity = 0
  Warning Severity = 0
  TOTAL = 0
The number of log messages written to the message log by severity
  Fatal Severity = 0
  Error Severity = 0
  Warning Severity = 0
  Timing Severity = 0
  Debug Severity = 0
  Unknown Severity = 0
  TOTAL = 0
sensor#

```

Interfaces Information

The **show interfaces** command is useful for gathering information on the sensing and command and control interfaces. This section describes the **show interfaces** command, and contains the following topics:

- [Understanding the show interfaces Command, page C-94](#)
- [Interfaces Command Output, page C-95](#)
- [Displaying Interface Traffic History, page C-96](#)

Understanding the show interfaces Command

You can learn the following information from the **show interfaces** command:

- Whether the interface is up or down
- Whether or not packets are being seen, and on which interfaces

- Whether or not packets are being dropped by SensorApp
- Whether or not there are errors being reported by the interfaces that can result in packet drops

The **show interfaces** command displays statistics for all system interfaces. Or you can use the individual commands to display statistics for the command and control interface (**show interfaces command_control_interface_name**), the sensing interface (**show interfaces interface_name**).

Interfaces Command Output

The following example shows the output from the **show interfaces** command:

```

sensor# show interfaces
Interface Statistics
  Total Packets Received = 0
  Total Bytes Received = 0
  Missed Packet Percentage = 0
  Current Bypass Mode = Auto_off
MAC statistics from interface GigabitEthernet0/1
  Media Type = backplane
  Missed Packet Percentage = 0
  Inline Mode = Unpaired
  Pair Status = N/A
  Link Status = Up
  Link Speed = Auto_1000
  Link Duplex = Auto_Full
  Total Packets Received = 0
  Total Bytes Received = 0
  Total Multicast Packets Received = 0
  Total Broadcast Packets Received = 0
  Total Jumbo Packets Received = 0
  Total Undersize Packets Received = 0
  Total Receive Errors = 0
  Total Receive FIFO Overruns = 0
  Total Packets Transmitted = 0
  Total Bytes Transmitted = 0
  Total Multicast Packets Transmitted = 0
  Total Broadcast Packets Transmitted = 0
  Total Jumbo Packets Transmitted = 0
  Total Undersize Packets Transmitted = 0
  Total Transmit Errors = 0
  Total Transmit FIFO Overruns = 0
MAC statistics from interface GigabitEthernet0/0
  Media Type = TX
  Link Status = Up
  Link Speed = Auto_100
  Link Duplex = Auto_Full
  Total Packets Received = 2211296
  Total Bytes Received = 157577635
  Total Multicast Packets Received = 20
  Total Receive Errors = 0
  Total Receive FIFO Overruns = 0
  Total Packets Transmitted = 239723
  Total Bytes Transmitted = 107213390
  Total Transmit Errors = 0
  Total Transmit FIFO Overruns = 0
sensor#

```

Displaying Interface Traffic History

Use the **show interfaces-history** [**traffic-by-hour** | **traffic-by-minute**] command in EXEC mode to display historical interfaces statistics for all system interfaces. The historical information for each interface is maintained for three days with 60 seconds granularity. Use the **show interfaces-history** {**FastEthernet** | **GigabitEthernet** | **Management** | **PortChannel**} [**traffic-by-hour** | **traffic-by-minute**] command to display statistics for specific interfaces.

**Note**

You must have health monitoring enabled to support the historic interface function.

Each record has the following details:

- Total packets received
- Total bytes received
- FIFO overruns
- Receive errors
- Received Mbps
- Missed packet percentage
- Average load
- Peak load

**Note**

Historical data for each interface for the past 72 hours is also included in the **show tech-support** command.

The following commands apply:

- **traffic-by-hour**—Displays interface traffic history by the hour.
- **traffic-by-minute**—Displays interface traffic history by the minute.
- **past**—Displays historical interface traffic information.
- **HH:MM**—Specifies the amount of time to go back in the past to begin the traffic display. The range for HH is 0 to 72. The range for MM is 0 to 59. The minimum value is 00:01 and the maximum value is 72:00.
- **FastEthernet**—Displays statistics for FastEthernet interfaces.
- **GigabitEthernet**—Displays statistics for GigabitEthernet interfaces.
- **Management**—Displays statistics for Management interfaces.

**Note**

Only platforms with external ports marked *Management* support this keyword.

- **PortChannel**—Displays statistics for PortChannel interfaces.

Displaying Historical Interface Statistics

To display interface traffic history, follow these steps:

- Step 1** Log in to the CLI.
- Step 2** Display the interface traffic history by the hour.

```

sensor# show interfaces-history traffic-by-hour past 02:15
GigabitEthernet0/0
Time                Packets Received  Bytes Received  Mbps  MPP  FIFO  Overruns  Receive Errors  Avg Load  Peak Load
11:30:31 UTC Tue Mar 05 2013  0                0                0     0    0      0          0                0          0
10:27:32 UTC Tue Mar 05 2013  0                0                0     0    0      0          0                0          0

GigabitEthernet0/1
Time                Packets Received  Bytes Received  Mbps  MPP  FIFO  Overruns  Receive Errors  Avg Load  Peak Load
11:30:31 UTC Tue Mar 05 2013  0                0                0     0    0      0          0                0          0
10:27:32 UTC Tue Mar 05 2013  0                0                0     0    0      0          0                0          0

GigabitEthernet0/2
Time                Packets Received  Bytes Received  Mbps  MPP  FIFO  Overruns  Receive Errors  Avg Load  Peak Load
11:30:31 UTC Tue Mar 05 2013  0                0                0     0    0      0          0                0          0
10:27:32 UTC Tue Mar 05 2013  0                0                0     0    0      0          0                0          0

GigabitEthernet0/3
Time                Packets Received  Bytes Received  Mbps  MPP  FIFO  Overruns  Receive Errors  Avg Load  Peak Load
11:30:31 UTC Tue Mar 05 2013  0                0                0     0    0      0          0                0          0
10:27:32 UTC Tue Mar 05 2013  0                0                0     0    0      0          0                0          0

Management0/0
Time                Packets Received  Bytes Received  Mbps  MPP  FIFO  Overruns  Receive Errors  Avg Load  Peak Load
11:30:31 UTC Tue Mar 05 2013  31071600         3240924703      0     0    0      0          0                0          0
10:27:32 UTC Tue Mar 05 2013  30859941         3216904786      0     0    0      0          0                0          0

--MORE--

```

- Step 3** Display the interface traffic history by the minute.

```

sensor# show interfaces-history traffic-by-minute past 00:45
GigabitEthernet0/0
Time                Packets Received  Bytes Received  Mbps  MPP  FIFO  Overruns  Receive Errors  Avg Load  Peak
Load
12:27:49 UTC Tue Mar 05 2013  0                0                0     0    0      0          0                0          0
12:26:45 UTC Tue Mar 05 2013  0                0                0     0    0      0          0                0          0
12:25:48 UTC Tue Mar 05 2013  0                0                0     0    0      0          0                0          0
12:24:42 UTC Tue Mar 05 2013  0                0                0     0    0      0          0                0          0
12:23:37 UTC Tue Mar 05 2013  0                0                0     0    0      0          0                0          0
12:22:30 UTC Tue Mar 05 2013  0                0                0     0    0      0          0                0          0
12:21:31 UTC Tue Mar 05 2013  0                0                0     0    0      0          0                0          0
12:20:29 UTC Tue Mar 05 2013  0                0                0     0    0      0          0                0          0
12:19:25 UTC Tue Mar 05 2013  0                0                0     0    0      0          0                0          0
12:18:18 UTC Tue Mar 05 2013  0                0                0     0    0      0          0                0          0
12:17:12 UTC Tue Mar 05 2013  0                0                0     0    0      0          0                0          0
12:16:07 UTC Tue Mar 05 2013  0                0                0     0    0      0          0                0          0
12:15:00 UTC Tue Mar 05 2013  0                0                0     0    0      0          0                0          0
12:13:54 UTC Tue Mar 05 2013  0                0                0     0    0      0          0                0          0
12:12:49 UTC Tue Mar 05 2013  0                0                0     0    0      0          0                0          0
12:11:43 UTC Tue Mar 05 2013  0                0                0     0    0      0          0                0          0
12:10:36 UTC Tue Mar 05 2013  0                0                0     0    0      0          0                0          0
12:09:30 UTC Tue Mar 05 2013  0                0                0     0    0      0          0                0          0
12:08:24 UTC Tue Mar 05 2013  0                0                0     0    0      0          0                0          0
12:07:25 UTC Tue Mar 05 2013  0                0                0     0    0      0          0                0          0
12:06:23 UTC Tue Mar 05 2013  0                0                0     0    0      0          0                0          0
12:05:25 UTC Tue Mar 05 2013  0                0                0     0    0      0          0                0          0
sensor#

```

Step 4 Display the interface traffic history for a specific interface.

```

sensor# show interfaces-history GigabitEthernet0/0 traffic-by-minute past 00:05
GigabitEthernet0/0
Time                Packets Received  Bytes Received  Mbps  MPP  FIFO  Overruns  Receive Errors  Avg Load  Peak Load
13:34:38 UTC Thu Mar 07 2013  0                0                0     0   0     0         0                0         0
13:33:35 UTC Thu Mar 07 2013  0                0                0     0   0     0         0                0         0
13:32:32 UTC Thu Mar 07 2013  0                0                0     0   0     0         0                0         0
13:31:27 UTC Thu Mar 07 2013  0                0                0     0   0     0         0                0         0
13:30:25 UTC Thu Mar 07 2013  0                0                0     0   0     0         0                0         0
sensor#

```

Events Information

You can use the **show events** command to view the alerts generated by SensorApp and errors generated by an application. This section describes the **show events** command, and contains the following topics:

- [Sensor Events, page C-98](#)
- [Understanding the show events Command, page C-98](#)
- [Displaying Events, page C-99](#)
- [Clearing Events, page C-102](#)

Sensor Events

There are five types of events:

- evAlert—Intrusion detection alerts
- evError—Application errors
- evStatus—Status changes, such as an IP log being created
- evLogTransaction—Record of control transactions processed by each sensor application
- evShunRqst—Block requests

Events remain in the Event Store until they are overwritten by newer events.

Understanding the show events Command



Note

The Event Store has a fixed size of 30 MB for all platforms.

The **show events** command is useful for troubleshooting event capture issues in which you are not seeing events in Event Viewer or Security Monitor. You can use the **show events** command to determine which events are being generated on the sensor to make sure events are being generated and that the fault lies with the monitoring side.

You can clear all events from Event Store by using the **clear events** command.

Here are the parameters for the **show events** command:

```

sensor# show events
<cr>
alert          Display local system alerts.
error          Display error events.

```

```

hh:mm[:ss]    Display start time.
log           Display log events.
nac          Display NAC shun events.
past         Display events starting in the past specified time.
status       Display status events.
|            Output modifiers.

```

Displaying Events



Note The Event Store has a fixed size of 30 MB for all platforms.



Note Events are displayed as a live feed. To cancel the request, press **Ctrl-C**.

Use the **show events** [{**alert** [informational] [low] [medium] [high] [**include-traits** *traits*] [**exclude-traits** *traits*] [**min-threat-rating** *min-rr*] [**max-threat-rating** *max-rr*] | **error** [warning] [error] [fatal] | **NAC** | **status**]}] [*hh:mm:ss* [*month day* [*year*]]] | **past** *hh:mm:ss*] command to display events from Event Store. Events are displayed beginning at the start time. If you do not specify a start time, events are displayed beginning at the current time. If you do not specify an event type, all events are displayed.

The following commands apply:

- **alert**—Displays alerts. Provides notification of some suspicious activity that may indicate an attack is in process or has been attempted. Alert events are generated by the Analysis Engine whenever a signature is triggered by network activity. If no level is selected (informational, low, medium, or high), all alert events are displayed.
- **include-traits**—Displays alerts that have the specified traits.
- **exclude-traits**—Does not display alerts that have the specified traits.
- **traits**—Specifies the trait bit position in decimal (0 to 15).
- **min-threat-rating**—Displays events with a threat rating above or equal to this value. The default is 0. The valid range is 0 to 100.
- **max-threat-rating**—Displays events with a threat rating below or equal to this value. The default is 100. The valid range is 0 to 100.
- **error**—Displays error events. Error events are generated by services when error conditions are encountered. If no level is selected (warning, error, or fatal), all error events are displayed.
- **NAC**—Displays the ARC (block) requests.



Note The ARC is formerly known as NAC. This name change has not been completely implemented throughout the IDM, the IME, and the CLI.

- **status**—Displays status events.
- **past**—Displays events starting in the past for the specified hours, minutes, and seconds.
- *hh:mm:ss*—Specifies the hours, minutes, and seconds in the past to begin the display.

**Note**

The **show events** command continues to display events until a specified event is available. To exit, press **Ctrl-C**.

Displaying Events

To display events from the Event Store, follow these steps:

Step 1 Log in to the CLI.

Step 2 Display all events starting now. The feed continues showing all events until you press **Ctrl-C**.

```
sensor# show events
evError: eventId=1041472274774840147 severity=warning vendor=Cisco
  originator:
    hostId: sensor2
    appName: cidwebserver
    appInstanceId: 12075
  time: 2011/01/07 04:41:45 2011/01/07 04:41:45 UTC
  errorMessage: name=errWarning received fatal alert: certificate_unknown

evError: eventId=1041472274774840148 severity=error vendor=Cisco
  originator:
    hostId: sensor2
    appName: cidwebserver
    appInstanceId: 351
  time: 2011/01/07 04:41:45 2011/01/07 04:41:45 UTC
  errorMessage: name=errTransport WebSession::sessionTask(6) TLS connection exception: handshake incomplete.
```

Step 3 Display the block requests beginning at 10:00 a.m. on February 9, 2011.

```
sensor# show events NAC 10:00:00 Feb 9 2011
evShunRqst: eventId=1106837332219222281 vendor=Cisco
  originator:
    deviceName: Sensor1
    appName: NetworkAccessControllerApp
    appInstance: 654
  time: 2011/02/09 10:33:31 2011/08/09 13:13:31
  shunInfo:
    host: connectionShun=false
    srcAddr: 11.0.0.1
    destAddr:
    srcPort:
    destPort:
    protocol: numericType=0 other
    timeoutMinutes: 40
  evAlertRef: hostId=esendHost 123456789012345678
sensor#
```

Step 4 Display errors with the warning level starting at 10:00 a.m. on February 9, 2011.

```
sensor# show events error warning 10:00:00 Feb 9 2011
evError: eventId=1041472274774840197 severity=warning vendor=Cisco
  originator:
    hostId: sensor
    appName: cidwebserver
    appInstanceId: 12160
  time: 2011/01/07 04:49:25 2011/01/07 04:49:25 UTC
  errorMessage: name=errWarning received fatal alert: certificate_unknown
```


Step 5 Display alerts from the past 45 seconds.

```
sensor# show events alert past 00:00:45

evIdsAlert: eventId=1109695939102805307 severity=medium vendor=Cisco
originator:
  hostId: sensor
  appName: sensorApp
  appInstanceId: 367
time: 2011/03/02 14:15:59 2011/03/02 14:15:59 UTC
signature: description=Nachi Worm ICMP Echo Request id=2156 version=S54
  subsigId: 0
  sigDetails: Nachi ICMP
interfaceGroup:
vlan: 0
participants:
  attacker:
    addr: locality=OUT 10.89.228.202
  target:
    addr: locality=OUT 10.89.150.185
riskRatingValue: 70
interface: fe0_1
protocol: icmp

evIdsAlert: eventId=1109695939102805308 severity=medium vendor=Cisco
originator:
--MORE--
```

Step 6 Display events that began 30 seconds in the past.

```
sensor# show events past 00:00:30
evStatus: eventId=1041526834774829055 vendor=Cisco
originator:
  hostId: sensor
  appName: mainApp
  appInstanceId: 2215
time: 2011/01/08 02:41:00 2011/01/08 02:41:00 UTC
controlTransaction: command=getVersion successful=true
description: Control transaction response.
requestor:
  user: cids
  application:
    hostId: 64.101.182.101
    appName: -cidcli
    appInstanceId: 2316

evStatus: eventId=1041526834774829056 vendor=Cisco
originator:
  hostId: sensor
  appName: login(pam_unix)
  appInstanceId: 2315
time: 2011/01/08 02:41:00 2011/01/08 02:41:00 UTC
syslogMessage:
  description: session opened for user cisco by cisco(uid=0)
```

Clearing Events

Use the **clear events** command to clear the Event Store.

To clear events from the Event Store, follow these steps:

Step 1 Log in to the CLI using an account with administrator privileges.

Step 2 Clear the Event Store.

```
sensor# clear events
Warning: Executing this command will remove all events currently stored in the event
store.
Continue with clear? []:
```

Step 3 Enter **yes** to clear the events.

cidDump Script

If you do not have access to the IDM, the IME, or the CLI, you can run the underlying script `cidDump` from the service account by logging in as `root` and running `/usr/cids/idsRoot/bin/cidDump`. The path of the `cidDump` file is `/usr/cids/idsRoot/htdocs/private/cidDump.html`. `cidDump` is a script that captures a large amount of information including the IPS processes list, log files, OS information, directory listings, package information, and configuration files.

To run the `cidDump` script, follow these steps:

Step 1 Log in to the sensor service account.

Step 2 `su` to `root` using the service account password.

Step 3 Enter the following command.

```
/usr/cids/idsRoot/bin/cidDump
```

Step 4 Enter the following command to compress the resulting `/usr/cids/idsRoot/log/cidDump.html` file.

```
gzip /usr/cids/idsRoot/log/cidDump.html
```

Step 5 Send the resulting HTML file to TAC or the IPS developers in case of a problem.

For More Information

For the procedure for putting a file on the Cisco FTP site, see [Uploading and Accessing Files on the Cisco FTP Site, page C-103](#).

Uploading and Accessing Files on the Cisco FTP Site

You can upload large files, for example, cidDump.html, the **show tech-support** command output, and cores, to the ftp-sj server.

To upload and access files on the Cisco FTP site, follow these steps:

-
- Step 1** Log in to ftp-sj.cisco.com as anonymous.
 - Step 2** Change to the /incoming directory.
 - Step 3** Use the **put** command to upload the files. Make sure to use the binary transfer type.
 - Step 4** To access uploaded files, log in to an ECS-supported host.
 - Step 5** Change to the /auto/ftp/incoming directory.
-



CLI Error Messages

This appendix lists the CLI error messages and CLI validation error messages. It contains the following sections:

- [CLI Error Messages, page D-1](#)
- [CLI Validation Error Messages, page D-6](#)

CLI Error Messages

[Table D-1](#) describes CLI error messages.

Table D-1 CLI Error Messages

Error Message	Reason	Command
<code>getVirtualSensorStatistics : Analysis Engine is busy</code>	Analysis Engine is busy because the virtual sensor has not finished initializing.	show statistics virtual-sensor
<code>getVirtualSensorStatistics : Analysis Engine is busy rebuilding regex tables. This may take a while.</code>	Analysis Engine is busy building cache files immediately after the sensor has been imaged.	show statistics virtual-sensor
<code>editConfigDeltaSignatureDefinition : Analysis Engine is busy rebuilding regex tables. This may take a while.</code>	Analysis Engine is busy building cache files immediately after the sensor has been imaged.	service signature-definition
<code>Invalid command received.</code>	The .conf file and code are out of synchronization, which should never occur in the field.	All commands
<code>Invalid port number was entered.</code>	An out-of-range port number was entered in URI.	copy, upgrade, show tech-support
<code>Invalid scheme was entered.</code>	Internal tables are out of synchronization, which should never occur in the field.	copy, upgrade, show tech-support
<code>Unknown scheme was entered.</code>	An invalid scheme was entered in URI.	copy, upgrade, show tech-support

Table D-1 CLI Error Messages (continued)

Error Message	Reason	Command
The filename <file> is not a valid upgrade file type.	Attempt to install the wrong file for your platform and version.	upgrade
idsPackageMgr: digital signature of the update was not valid	The signature update or service pack is corrupt. Contact TAC.	upgrade
Cannot create a new event-action-rules configuration. "rules0" is currently the only configuration allowed.	An invalid logical instance name was entered for service event action rules. ¹	service event-action-rules
Cannot create a new signature-definition configuration. "sig0" is currently the only configuration allowed.	An invalid logical instance name was entered for service signature definition. ²	service signature-definition
Cannot create a new anomaly-detection configuration. "ad0" is currently the only configuration allowed.	An invalid logical instance name was entered for service anomaly detection. ³	service anomaly-detection
This node is Threat Profile tuned, reverting to default entirely may cause certain unexpected results. Please use 'show settings' to know the default value for a parameter and set it to its default value.	Defaulting a particular signature that is both threat-profile tuned and custom tuned is not allowed.	service signature-definition
Configuration instance "sig12" does not exist. Use the global configuration command "service signature-definition" to create a new instance.	Cannot add a threat profile for a signature instance that does not exist.	service threat-profile
User does not exist.	The administrator is attempting to change the password for a username that does not exist in the system.	password
Incorrect password for user account.	The user entered an invalid password while attempting to change the password.	password
Empty user list.	The curUserAccountList.xml file does not contain any entries, which should never occur in the field.	username
User already exists.	An attempt to create a user that already exists in the system was made.	username
Cannot communicate with system processes. Please contact your system administrator.	One or more required applications is not responding to control transactions.	All commands

Table D-1 CLI Error Messages (continued)

Error Message	Reason	Command
Source and Destination are the same.	—	copy
Backup config was missing.	The user attempted to copy or erase the backup config file but no backup config file has been generated.	copy erase
Could not load CLI configuration files, can not complete request.	The .conf files could not be located, which should never occur in the field.	copy
Error writing to <URL>.	The URL specified in the destination could not be written.	copy
Error reading from <URL>.	The URL specified in the source could not be read.	copy
Packet-file does not exist.	The user attempted to copy or erase the packet-file but no packet-file has been captured.	copy erase
No downgrade available.	The user attempted to downgrade a system that has not been upgraded.	downgrade
No packet-file available.	The user attempted to display the file-info or the packet-file but no packet-file exists.	packet
Another user is currently capturing into the packet-file. Please try again later.	—	packet capture
Another CLI client is currently displaying packets from the interface.	The user must wait for the other CLI session to terminate display before this will be available. Multiple users may display the command control interface simultaneously.	packet display
Log does not exist.	The user attempted to copy or display an iplog that does not exist.	copy iplog packet display iplog
The requested IPLOG is not complete. Please try again after the IPLOG status is 'completed.'	The user attempted to copy or display an iplog that is not complete.	copy iplog
Error: Log file exists but an error occurred during read. The log file might have been overwritten.	The user was displaying or copying an iplog file that was overwritten. The partial file contents should still be viewable.	copy iplog
Error: Iplug transfer failed. The copied log file is incomplete.	Iplug transfer has failed. There may or may not be any partial copied file at remote server.	copy iplog
Could not create pipe /usr/cids/idsRoot/tmp/pipe_cliPacket.<pid>.tmp	Could not open pipe for sending iplog file. This indicates a space or resource limitation, which should not occur in the field.	copy iplog

Table D-1 CLI Error Messages (continued)

Error Message	Reason	Command
Error: The log file might have been overwritten while the copy was in progress. The copied log file may be viewable but is incomplete.	The iplog was overwritten while it was being copied off the sensor.	copy iplog
Could not read license file.	The license file was copied but cannot be opened.	copy license-key
Could not write the temporary license file location used to copy the file off the box.	Could not open the temporary storage location /usr/cids/idsRoot/tmp/ips.lic. This indicates a space issue, which should not occur in the field.	copy license-key
Virtual sensor name does not exist.	The user attempted to start or stop an iplog on a non-existent virtual sensor.	iplog
You do not have permission to terminate the requested CLI session.	An operator or viewer user attempted to terminate a CLI session belonging to another user.	clear line
Invalid CLI ID specified, use the 'show users all' command to view the valid CLI session IDs.	The user attempted to cancel a CLI session that does not exist.	clear line
The maximum allowed CLI sessions are currently open, please try again later.	Operator or viewer user attempted to log in when the maximum number of CLI sessions were already open.	initial login
The maximum allowed CLI sessions are currently open, would you like to terminate one of the open sessions?	Administrator user attempted to log in when the maximum number of CLI sessions were already open.	initial login
Can not communicate with system processes. Please contact your system administrator.	The CLI cannot contact the applications on the sensor to retrieve start-up information. This is a fatal error that should never happen. The user has to log in to the service account and manually reboot the sensor.	initial login
The instance cannot be removed. Instance assigned to virtual sensor name.	The user attempted to remove a configuration instance that is currently assigned to a virtual sensor. Use the default service command to reset the configuration setting to default.	no service component instance
Insufficient disk space to complete request.	Not enough disk space is available to create a new instance of a configuration file.	copy instance service component instance

1. This error only occurs on platforms that do not support virtual policies.
2. This error only occurs on platforms that do not support virtual policies.

3. This error only occurs on platforms that do not support virtual policies.

CLI Validation Error Messages

Table D-2 describes the validation error messages.

Table D-2 Validation Error Messages

Error Message	Reason/Location
Interface 'name' has not been subdivided.	The physical interface or inline interface <i>name</i> subinterface type is none (service interface submode).
Interface 'name' subinterface 'num' does not exist.	The physical interface <i>name</i> has been subdivided into inline VLAN pairs, but the specified subinterface number does not exist (service interface submode).
Interface 'name' is the command-control interface.	The physical interface <i>name</i> is the command and control interface (service interface submode).
Interface 'name' has been subdivided.	The physical interface <i>name</i> subinterface type is inline VLAN pair or VLAN group. Or the inline interface <i>name</i> subinterface type is VLAN group (service interface submode).
Interface 'name' is assigned to inline-interfaces 'inlinename.'	The physical interface <i>name</i> is assigned to an inline interface entry's interface1 or interface2 (service interface submode).
Vlan 'vlannum' is assigned to subinterface 'subnum.'	The VLAN <i>vlannum</i> is already assigned to a different subinterface <i>subnum</i> entry's vlan1 or vlan2 (service interface submode).
Vlan range 'vlanrange' overlaps with vlans assigned to subinterface 'subnum.'	The VLAN range <i>vlanrange</i> contains values that are already used in a different subinterface <i>subnum</i> entry's <i>vlans range</i> (service interface submode).
Unassigned vlans already assigned to subinterface 'subnum.'	Unassigned VLANs have already been selected in a different subinterface <i>subnum</i> entry.
Inline-interface 'inlinename' does not exist.	The inline interface <i>inlinename</i> does not exist (service interface submode).
The default-vlans for the selected interfaces do not match. interface1, 'name' default-vlan is 'vlannum,' interface2, 'name' default-vlan is 'vlannum.'	The user is trying to change the subinterface type of an inline interface to VLAN group, but the default VLANs for the two interfaces assigned to the inline interface do not match (service interface submode).
interface1 and interface2 must be set before the logical interface can be divided into subinterfaces.	The user is trying to change the subinterface type of an inline interface to VLAN group, but has not set both interface1 and interface2 (service interface submode).
Interface 'name' has not been subdivided into inline-vlan-pairs.	The physical interface <i>name</i> subinterface type is not inline VLAN pair (service interface submode).

Table D-2 Validation Error Messages (continued)

Error Message	Reason/Location
Interface already assigned to virtual sensor `vsname.`	The interface and optional sub-interface being added to the virtual sensor entry physical interface set has already been assigned to another virtual sensor entry.
The instance cannot be removed. Instance assigned to virtual sensor `vsname.`	The user is trying to remove a signature definition, event action rules, or anomaly detection configuration file that is currently in use by virtual sensor <i>vsname</i> .



Revised: April 23, 2014

Numerals

- 3DES** Triple Data Encryption Standard. A stronger version of DES, which is the default encryption method for SSH version 1.5. Used when establishing an SSH session with the sensor. It can be used when the sensor is managing a device.
- 802.x** A set of IEEE standards for the definition of LAN protocols.

A

- AAA** authentication, authorization, and accounting. Pronounced “triple a.” The primary and recommended method for access control in Cisco devices.
- ACE** Access Control Entry. An entry in the ACL that describes what action should be taken for a specified address or protocol. The sensor adds/removes ACE to block hosts.
- ACK** acknowledgment. Notification sent from one network device to another to acknowledge that some event occurred (for example, the receipt of a message).
- ACL** Access Control List. A list of ACEs that control the flow of data through a router. There are two ACLs per router interface for inbound data and outbound data. Only one ACL per direction can be active at a time. ACLs are identified by number or by name. ACLs can be standard, enhanced, or extended. You can configure the sensor to manage ACLs.
- ACS server** Cisco Access Control Server. A RADIUS security server that is the centralized control point for managing network users, network administrators, and network infrastructure resources.
- action** The response of the sensor to an event. An action only happens if the event is not filtered. Examples include TCP reset, block host, block connection, IP logging, and capturing the alert trigger packet.
- active ACL** The ACL created and maintained by ARC and applied to the router block interfaces.
- AIC engine** Application Inspection and Control engine. Provides deep analysis of web traffic. It provides granular control over HTTP sessions to prevent abuse of the HTTP protocol. It allows administrative control over applications that try to tunnel over specified ports, such as instant messaging, and tunneling applications, such as gotomypc. It can also inspect FTP traffic and control the commands being issued.
- Alarm Channel** The IPS software module that processes all signature events generated by the inspectors. Its primary function is to generate alerts for each event it receives.
- alert** Specifically, an IPS event type; it is written to the Event Store as an evidsAlert. In general, an alert is an IPS message that indicates a network exploit in progress or a potential security problem occurrence. Also known as an alarm.

Analysis Engine	The IPS software module that handles sensor configuration. It maps the interfaces and also the signature and alarm channel policy to the configured interfaces. It performs packet analysis and alert detection. The Analysis Engine functionality is provided by the SensorApp process.
anomaly detection	AD. The sensor component that creates a baseline of normal network traffic and then uses this baseline to detect worm-infected hosts.
API	Application Programming Interface. The means by which an application program talks to communications software. Standardized APIs allow application programs to be developed independently of the underlying method of communication. Computer application programs run a set of standard software interrupts, calls, and data formats to initiate contact with other devices (for example, network services, mainframe communications programs, or other program-to-program communications). Typically, APIs make it easier for software developers to create links that an application needs to communicate with the operating system or with the network.
application	Any program (process) designed to run in the Cisco IPS environment.
application image	Full IPS image stored on a permanent storage device used for operating the sensor.
application instance	A specific application running on a specific piece of hardware in the IPS environment. An application instance is addressable by its name and the IP address of its host computer.
application partition	The bootable disk or compact-flash partition that contains the IPS software image.
ARC	Attack Response Controller. Formerly known as Network Access Controller (NAC). A component of the IPS. A software module that provides block and unblock functionality where applicable.
architecture	The overall structure of a computer or communication system. The architecture influences the capabilities and limitations of the system.
ARP	Address Resolution Protocol. Internet protocol used to map an IP address to a MAC address. Defined in RFC 826.
ASN.1	Abstract Syntax Notation 1. Standard for data presentation.
aspect version	Version information associated with a group of IDIOM default configuration settings. For example, Cisco Systems publishes the standard set of attack signatures as a collection of default settings with the S aspect. The S-aspect version number is displayed after the S in the signature update package file name. Other aspects include the Virus signature definitions in the V-aspect and IDIOM signing keys in the key-aspect.
atomic attack	Represents exploits contained within a single packet. For example, the “ping of death” attack is a single, abnormally large ICMP packet.
Atomic engine	There are two Atomic engines: Atomic IP inspects IP protocol packets and associated Layer-4 transport protocols, and Atomic ARP inspects Layer-2 ARP protocol.
attack	An assault on system security that derives from an intelligent threat, that is, an intelligent act that is a deliberate attempt (especially in the sense of method or technique) to evade security services and violate the security policy of a system.
attack relevance rating	ARR. A weight associated with the relevancy of the targeted OS. The attack relevance rating is a derived value (relevant, unknown, or not relevant), which is determined at alert time. The relevant OSes are configured per signature.

attack severity rating	ASR. A weight associated with the severity of a successful exploit of the vulnerability. The attack severity rating is derived from the alert severity parameter (informational, low, medium, or high) of the signature. The attack severity rating is configured per signature and indicates how dangerous the event detected is.
authentication	Process of verifying that a user has permission to use the system, usually by means of a password key or certificate.
AuthenticationApp	A component of the IPS. Authorizes and authenticates users based on IP address, password, and digital certificates.
autostate	In normal autostate mode, the Layer 3 interfaces remain up if at least one port in the VLAN remains up. If you have appliances, such as load balancers or firewall servers that are connected to the ports in the VLAN, you can configure these ports to be excluded from the autostate feature to make sure that the forwarding SVI does not go down if these ports become inactive.
AV	Anti-Virus.
<hr/>	
B	
backplane	The physical connection between an interface processor or card and the data buses and the power distribution buses inside a chassis.
base version	A software release that must be installed before a follow-up release, such as a service pack or signature update, can be installed. Major and minor updates are base version releases.
benign trigger	A situation in which a signature is fired correctly, but the source of the traffic is nonmalicious.
BIOS	Basic Input/Output System. The program that starts the sensor and communicates between the devices in the sensor and the system.
blackhole	Routing term for an area of the internetwork where packets enter, but do not emerge, due to adverse conditions or poor system configuration within a portion of the network.
block	The ability of the sensor to direct a network device to deny entry to all packets from a specified network host or network.
block interface	The interface on the network device that the sensor manages.
BO	BackOrifice. The original Windows back door Trojan that ran over UDP only.
BO2K	BackOrifice 2000. A Windows back door Trojan that runs over TCP and UDP.
Botnets	A collection of software robots, or bots, that run autonomously and automatically. The term is often associated with malicious software but it can also refer to the network of computers using distributed computing software. The term Botnet is used to refer to a collection of compromised computers (called Zombie computers) running software, usually installed through worms, Trojan horses, or back doors, under a common command-and-control infrastructure.
Bpdu	Bridge Protocol Data Unit. Spanning-Tree Protocol hello packet that is sent out at configurable intervals to exchange information among bridges in the network.

Bubble Babble	In computing, Bubble Babble is a binary data encoding designed by Antti Huima. This encoding uses alternation of consonants and vowels to encode binary data to pseudowords that can be pronounced more easily than arbitrary lists of hexadecimal digits. While Bubble Babble is technically a binary encoding, it also acts as a 65,536-digit positional number system with a one-to-one mapping from each five-character sequence to 16 bits of data.
bypass mode	Mode that lets packets continue to flow through the sensor even if the sensor fails. Bypass mode is only applicable to inline-paired interfaces.

C

CA	certification authority. Entity that issues digital certificates (especially X.509 certificates) and vouches for the binding between the data items in a certificate. Sensors use self-signed certificates.
CA certificate	Certificate for one CA issued by another CA.
CEF	Cisco Express Forwarding. CEF is advanced, Layer 3 IP switching technology. CEF optimizes network performance and scalability for networks with large and dynamic traffic patterns, such as the Internet, on networks characterized by intensive Web-based applications, or interactive sessions.
certificate	Digital representation of user or device attributes, including a public key, that is signed with an authoritative private key.
cidDump	A script that captures a large amount of information including the IPS processes list, log files, OS information, directory listings, package information, and configuration files.
CIDEE	Cisco Intrusion Detection Event Exchange. Specifies the extensions to SDEE that are used by Cisco IPS systems. The CIDEE standard specifies all possible extensions that may be supported by Cisco IPS systems.
CIDS header	The header that is attached to each packet in the IPS system. It contains packet classification, packet length, checksum results, timestamp, and the receive interface.
cipher key	The secret binary data used to convert between clear text and cipher text. When the same cipher key is used for both encryption and decryption, it is called symmetric. When it is used for either encryption or decryption (but not both), it is called asymmetric.
Cisco IOS	Cisco system software that provides common functionality, scalability, and security for all products under the CiscoFusion architecture. Cisco IOS allows centralized, integrated, and automated installation and management of internetworks while supporting a wide variety of protocols, media, services, and platforms.
CLI	command-line interface. A shell provided with the sensor used for configuring and controlling the sensor applications.
CollaborationApp	A component of the IPS. Shares information with other devices through a global correlation database to improve the combined efficacy of all the devices.
command and control interface	The interface on the sensor that communicates with the IPS manager and other network devices. This interface has an assigned IP address.
community	In SNMP, a logical group of managed devices and NMSs in the same administrative domain.

composite attack	Spans multiple packets in a single session. Examples include most conversation attacks such as FTP, Telnet, and most Regex-based attacks.
connection block	ARC blocks traffic from a given source IP address to a given destination IP address and destination port.
console	A terminal or laptop computer used to monitor and control the sensor.
console port	An RJ45 or DB9 serial port on the sensor that is used to connect to a console device.
control interface	When ARC opens a Telnet or SSH session with a network device, it uses one of the routing interfaces of the device as the remote IP address. This is the control interface.
control transaction	CT. An IPS message containing a command addressed to a specific application instance. Control transactions can be sent between a management application and an IPS sensor, or between applications on the same IPS sensor. Example control transactions include <i>start</i> , <i>stop</i> , <i>getConfig</i> .
Control Transaction Server	A component of the IPS. Accepts control transactions from a remote client, initiates a local control transaction, and returns the response to the remote client.
Control Transaction Source	A component of the IPS. Waits for control transactions directed to remote applications, forwards the control transactions to the remote node, and returns the response to the initiator.
cookie	A piece of information sent by a web server to a web browser that the browser is expected to save and send back to the web server whenever the browser makes additional requests of the web server.
CSM	Cisco Security Manager, the provisioning component of the Cisco Self-Defending Networks solution. CS-Manager is fully integrated with CS-MARS.
cut-through architecture	Cut-through architecture is one method of design for packet-switching systems. When a packet arrives at a switch, the switch starts forwarding the packet almost immediately, reading only the first few bytes in the packet to learn the destination address. This technique improves performance
CVE	Common Vulnerabilities and Exposures. A list of standardized names for vulnerabilities and other information security exposures maintained at http://cve.mitre.org/ .

D

darknets	A virtual private network where users connect only to people they trust. In its most general meaning, a darknet can be any type of closed, private group of people communicating, but the name is most often used specifically for file-sharing networks. Darknet can be used to refer collectively to all covert communication networks.
Database Processor	A processor in the IPS. Maintains the signature state and flow databases.
datagram	Logical grouping of information sent as a network layer unit over a transmission medium without prior establishment of a virtual circuit. IP datagrams are the primary information units in the Internet. The terms cell, frame, message, packet, and segment also are used to describe logical information groupings at various layers of the OSI reference model and in various technology circles.

DCE	data circuit-terminating equipment (ITU-T expansion). Devices and connections of a communications network that comprise the network end of the user-to-network interface. The DCE provides a physical connection to the network, forwards traffic, and provides a clocking signal used to synchronize data transmission between DCE and DTE devices. Modems and interface cards are examples of DCE.
DCOM	Distributed Component Object Model. Protocol that enables software components to communicate directly over a network. Developed by Microsoft and previously called Network OLE, DCOM is designed for use across multiple network transports, including such Internet protocols as HTTP.
DDoS	Distributed Denial of Service. An attack in which a multitude of compromised systems attack a single target, thereby causing denial of service for users of the targeted system. The flood of incoming messages to the target system essentially forces it to shut down, thereby denying service to the system to legitimate users.
Deny Filters Processor	A processor in the IPS. Handles the deny attacker functions. It maintains a list of denied source IP addresses.
DES	Data Encryption Standard. A strong encryption method where the strength lies in a 56-bit key rather than an algorithm.
destination address	Address of a network device that is receiving data.
DIMM	Dual In-line Memory Modules.
DMZ	demilitarized zone. A separate network located in the neutral zone between a private (inside) network and a public (outside) network.
DNS	Domain Name System. An Internet-wide hostname to IP address mapping. DNS enables you to convert human-readable names into the IP addresses needed for network packets.
DoS	Denial of Service. An attack whose goal is just to disrupt the operation of a specific system or network.
DRAM	dynamic random-access memory. RAM that stores information in capacitors that must be refreshed periodically. Delays can occur because DRAMs are inaccessible to the processor when refreshing their contents. However, DRAMs are less complex and have greater capacity than SRAMs.
DTE	Data Terminal Equipment. Refers to the role of a device on an RS-232C connection. A DTE writes data to the transmit line and reads data from the receive line.
DTP	Dynamic Trunking Protocol. A Cisco proprietary protocol in the VLAN group used for negotiating trunking on a link between two devices and for negotiating the type of trunking encapsulation (ISL or 802.1q) to be used.

E

ECLB	Ether Channel Load Balancing. Lets a Catalyst switch split traffic flows over different physical paths.
egress	Traffic leaving the network.
encryption	Application of a specific algorithm to data to alter the appearance of the data making it incomprehensible to those who are not authorized to see the information.

engine	A component of the sensor designed to support many signatures in a certain category. Each engine has parameters that can be used to create signatures or tune existing signatures.
enterprise network	Large and diverse network connecting most major points in a company or other organization. Differs from a WAN in that it is privately owned and maintained.
escaped expression	Used in regular expression. A character can be represented as its hexadecimal value, for example, \x61 equals 'a,' so \x61 is an escaped expression representing the character 'a.'
ESD	electrostatic discharge. Electrostatic discharge is the rapid movement of a charge from one object to another object, which produces several thousand volts of electrical charge that can cause severe damage to electronic components or entire circuit card assemblies.
event	An IPS message that contains an alert, a block request, a status message, or an error message.
Event Store	One of the components of the IPS. A fixed-size, indexed store used to store IPS events.
evldsAlert	The XML entity written to the Event Store that represents an alert.

F

fail closed	Blocks traffic on the device after a hardware failure.
fail open	Lets traffic pass through the device after a hardware failure.
false negative	A signature is not fired when offending traffic is detected.
false positive	Normal traffic or a benign action causes a signature to fire.
Fast Ethernet	Any of a number of 100-Mbps Ethernet specifications. Fast Ethernet offers a speed increase 10 times that of the 10BaseT Ethernet specification while preserving such qualities as frame format, MAC mechanisms, and MTU. Such similarities allow the use of existing 10BaseT applications and network management tools on Fast Ethernet networks. Based on an extension to the IEEE 802.3 specification.
Fast flux	Fast flux is a DNS technique used by Botnets to hide phishing and malware delivery sites behind an ever-changing network of compromised hosts acting as proxies. It can also refer to the combination of peer-to-peer networking, distributed command and control, web-based load balancing and proxy redirection used to make malware networks more resistant to discovery and counter-measures. The Storm Worm is one of the recent malware variants to make use of this technique.
firewall	Router or access server, or several routers or access servers, designated as a buffer between any connected public networks and a private network. A firewall router uses access lists and other methods to ensure the security of the private network.
Flood engine	Detects ICMP and UDP floods directed at hosts and networks.
flooding	Traffic passing technique used by switches and bridges in which traffic received on an interface is sent out all the interfaces of that device except the interface on which the information was received originally.
forwarding	Process of sending a frame toward its ultimate destination by way of an internetworking device.
fragment	Piece of a larger packet that has been broken down to smaller units.

fragmentation	Process of breaking a packet into smaller units when transmitting over a network medium that cannot support the original size of the packet.
Fragment Reassembly Processor	A processor in the IPS. Reassembles fragmented IP datagrams. It is also responsible for normalization of IP fragments when the sensor is in inline mode.
FTP	File Transfer Protocol. Application protocol, part of the TCP/IP protocol stack, used for transferring files between network nodes. FTP is defined in RFC 959.
FTP server	File Transfer Protocol server. A server that uses the FTP protocol for transferring files between network nodes.
full duplex	Capability for simultaneous data transmission between a sending station and a receiving station.
FQDN	Fully Qualified Domain Name. A domain name that specifies its exact location in the tree hierarchy of the DNS. It specifies all domain levels, including the top-level domain, relative to the root domain. A fully qualified domain name is distinguished by this absoluteness in the name space.

G

GBIC	GigaBit Interface Converter. Often refers to a fiber optic transceiver that adapts optical cabling to fiber interfaces. Fiber-ready switches and NICs generally provide GBIC and/or SFP slots. For more information, refer to the <i>Catalyst Switch Cable, Connector, and AC Power Cord Guide</i> .
Gigabit Ethernet	Standard for a high-speed Ethernet, approved by the IEEE (Institute of Electrical and Electronics Engineers) 802.3z standards committee in 1996.
global correlation	The IPS sensor shares information with other devices through a global correlation database to improve the combined efficacy of all devices.
global correlation client	The software component of CollaborationApp that obtains and installs updates to the local global correlation databases.
global correlation database	The collective information obtained from and shared with collaborative devices such as IPS sensors.
GMT	Greenwich Mean Time. Time zone at zero degrees longitude. Now called Coordinated Universal Time (UTC).
GRUB	Grand Unified Bootloader. Boot loader is the first software program that runs when a computer starts. It is responsible for loading and transferring control to the operating system kernel software. The kernel, in turn, initializes the rest of the operating system.

H

H.225.0	An ITU standard that governs H.225.0 session establishment and packetization. H.225.0 actually describes several different protocols: RAS, use of Q.931, and use of RTP.
H.245	An ITU standard that governs H.245 endpoint control.

H.323	Allows dissimilar communication devices to communicate with each other by using a standardized communication protocol. H.323 defines a common set of CODECs, call setup and negotiating procedures, and basic data transport methods.
half duplex	Capability for data transmission in only one direction at a time between a sending station and a receiving station. BSC is an example of a half-duplex protocol.
handshake	Sequence of messages exchanged between two or more network devices to ensure transmission synchronization.
hardware bypass	A specialized interface card that pairs physical interfaces so that when a software error is detected, a bypass mechanism is engaged that directly connects the physical interfaces and allows traffic to flow through the pair. Hardware bypass passes traffic at the network interface, does not pass it to the IPS system.
host block	ARC blocks all traffic from a given IP address.
HTTP	Hypertext Transfer Protocol. The stateless request/response media transfer protocol used in the IPS architecture for remote data exchange.
HTTPS	An extension to the standard HTTP protocol that provides confidentiality by encrypting the traffic from the website. By default this protocol uses TCP port 443.

I

ICMP	Internet Control Message Protocol. Network layer Internet protocol that reports errors and provides other information relevant to IP packet processing. Documented in RFC 792.
ICMP flood	Denial of Service attack that sends a host more ICMP echo request (“ping”) packets than the protocol implementation can handle.
IDAPI	Intrusion Detection Application Programming Interface. Provides a simple interface between IPS architecture applications. IDAPI reads and writes event data and provides a mechanism for control transactions.
IDCONF	Intrusion Detection Configuration. A data format standard that defines operational messages that are used to configure intrusion detection and prevention systems.
IDENT	Ident protocol, specified in RFC 1413, is an Internet protocol that helps identify the user of a particular TCP connection.
IDIOM	Intrusion Detection Interchange and Operations Messages. A data format standard that defines the event messages that are reported by intrusion detection systems and the operational messages that are used to configure and control intrusion detection systems.
IDM	IPS Device Manager. A web-based application that lets you configure and manage your sensor. The web server for IDM resides on the sensor. You can access it through Internet Explorer or Firefox web browsers.
IDMEF	Intrusion Detection Message Exchange Format. The IETF Intrusion Detection Working Group draft standard.

IME	IPS Manager Express. A network management application that provides system health monitoring, events monitoring, reporting, and configuration for up to ten sensors.
inline mode	All packets entering or leaving the network must pass through the sensor.
inline interface	A pair of physical interfaces configured so that the sensor forwards all traffic received on one interface out to the other interface in the pair.
InterfaceApp	A component of the IPS. Handles bypass and physical settings and defines paired interfaces. Physical settings are speed, duplex, and administrative state.
intrusion detection system	IDS. A security service that monitors and analyzes system events to find and provide real-time or near real-time warning of attempts to access system resources in an unauthorized manner.
IP address	32-bit address assigned to hosts using TCP/IP. An IP address belongs to one of five classes (A, B, C, D, or E) and is written as 4 octets separated by periods (dotted decimal format). Each address consists of a network number, an optional subnetwork number, and a host number. The network and subnetwork numbers together are used for routing, and the host number is used to address an individual host within the network or subnetwork. A subnet mask is used to extract network and subnetwork information from the IP address.
IPS	Intrusion Prevention System. A system that alerts the user to the presence of an intrusion on the network through network traffic analysis techniques.
IPS data or message	Describes the messages transferred over the command and control interface between IPS applications.
iplog	A log of the binary packets to and from a designated address. Iplogs are created when the log Event Action is selected for a signature. Iplogs are stored in a libpcap format, which can be read by WireShark and TCPDUMP.
IP spoofing	IP spoofing attack occurs when an attacker outside your network pretends to be a trusted user either by using an IP address that is within the range of IP addresses for your network or by using an authorized external IP address that you trust and to which you want to provide access to specified resources on your network. Should an attacker get access to your IPSec security parameters, that attacker can masquerade as the remote user authorized to connect to the corporate network.
IPv6	IP version 6. Replacement for the current version of IP (version 4). IPv6 includes support for flow ID in the packet header, which can be used to identify flows. Formerly called IPng (next generation).
ISL	Inter-Switch Link. Cisco-proprietary protocol that maintains VLAN information as traffic flows between switches and routers.

J

Java Web Start	Java Web Start provides a platform-independent, secure, and robust deployment technology. It enables developers to deploy full-featured applications to you by making the applications available on a standard web server. With any web browser, you can launch the applications and be confident you always have the most-recent version.
JNLP	Java Network Launching Protocol. Defined in an XML file format specifying how Java Web Start applications are launched. JNLP consists of a set of rules defining how exactly the launching mechanism should be implemented.

K

KB	Knowledge Base. The sets of thresholds learned by Anomaly Detection and used for worm virus detection.
Knowledge Base	See KB.

L

LACP	Link Aggregation Control Protocol. LACP aids in the automatic creation of EtherChannel links by exchanging LACP packets between LAN ports. This protocol is defined in IEEE 802.3ad.
LAN	Local Area Network. Refers to the Layer 2 network domain local to a given host. Packets exchanged between two hosts on the same LAN do not require Layer 3 routing.
Layer 2 Processor	A processor in the IPS. Processes layer 2-related events. It also identifies malformed packets and removes them from the processing path.
Logger	A component of the IPS. Writes all the log messages of the application to the log file and the error messages of the application to the Event Store.
logging	Gathers actions that have occurred in a log file. Logging of security information is performed on two levels: logging of events (such as IPS commands, errors, and alerts), and logging of individual IP session information.
LOKI	Remote access, back door Trojan, ICMP tunneling software. When the computer is infected, the malicious code creates an ICMP tunnel that can be used to send small payload ICMP replies.

M

MainApp	The main application in the IPS. The first application to start on the sensor after the operating system has booted. Reads the configuration and starts applications, handles starting and stopping of applications and node reboots, handles software upgrades.
major update	A base version that contains major new functionality or a major architectural change in the product.
Malware	Malicious software that is installed on an unknowing host.
manufacturing image	Full IPS system image used by manufacturing to image sensors.
master blocking sensor	A remote sensor that controls one or more devices. Blocking forwarding sensors send blocking requests to the master blocking sensor and the master blocking sensor executes the blocking requests.
MD5	Message Digest 5. A one-way hashing algorithm that produces a 128-bit hash. Both MD5 and Secure Hash Algorithm (SHA) are variations on MD4 and strengthen the security of the MD4 hashing algorithm. Cisco uses hashes for authentication within the IPSec framework. Also used for message authentication in SNMP v.2. MD5 verifies the integrity of the communication, authenticates the origin, and checks for timeliness.

Meta engine	Defines events that occur in a related manner within a sliding time interval. This engine processes events rather than packets.
MIB	Management Information Base. Database of network management information that is used and maintained by a network management protocol, such as SNMP or CMIP. The value of a MIB object can be changed or retrieved using SNMP or CMIP commands, usually through a GUI network management system. MIB objects are organized in a tree structure that includes public (standard) and private (proprietary) branches.
MIME	Multipurpose Internet Mail Extension. Standard for transmitting nontext data (or data that cannot be represented in plain ASCII code) in Internet mail, such as binary, foreign language text (such as Russian or Chinese), audio, or video data. MIME is defined in RFC 2045.
minor update	A minor version that contains minor enhancements to the product line. Minor updates are incremental to the major version, and are also base versions for service packs.
monitoring interface	See sensing interface.
MPF	Modular Policy Framework. A means of configuring security appliance features in a manner similar to Cisco IOS software Modular QoS CLI.
MSFC, MSFC2	Multilayer Switch Feature Card. An optional card on a Catalyst 6000 supervisor engine that performs L3 routing for the switch.
MSRPC	Microsoft Remote Procedure Call. MSRPC is the Microsoft implementation of the DCE RPC mechanism. Microsoft added support for Unicode strings, implicit handles, inheritance of interfaces (which are extensively used in DCOM), and complex calculations in the variable-length string and structure paradigms already present in DCE/RPC.
MySDN	My Self-Defending Network. A part of the signature definition section of IDM and IME. It provides detailed information about signatures.

N

NAC	Network Access Controller. See ARC.
NAS-ID	Network Access ID. An identifier that clients send to servers to communicate the type of service they are attempting to authenticate.
NAT	Native Address Translation. A network device can present an IP address to the outside networks that is different from the actual IP address of a host.
NBD	Next Business Day. The arrival of replacement hardware according to Cisco service contracts.
Neighborhood Discovery	Protocol for IPv6. IPv6 nodes on the same link use Neighbor Discovery to discover each other's presence, to determine each other's link-layer addresses, to find routers, and to maintain reachability information about the paths to active neighbors.
Network Access ID	See NAS-ID.

network device	A device that controls IP traffic on a network and can block an attacking host. An example of a network device is a Cisco router or PIX Firewall.
network participation	Networks contributing learned information to the global correlation database.
network participation client	The software component of CollaborationApp that sends data to the SensorBase Network.
never block address	Hosts and networks you have identified that should never be blocked.
never shun address	See never block address.
NIC	Network Interface Card. Board that provides network communication capabilities to and from a computer system.
NMS	network management system. System responsible for managing at least part of a network. An NMS is generally a reasonably powerful and well-equipped computer, such as an engineering workstation. NMSs communicate with agents to help keep track of network statistics and resources.
node	A physical communicating element on the command and control network. For example, an appliance or a router.
Normalizer engine	Configures how the IP and TCP normalizer functions and provides configuration for signature events related to the IP and TCP normalizer.
NOS	network operating system. Generic term used to refer to distributed file systems. Examples include LAN Manager, NetWare, NFS, and VINES.
NotificationApp	A component of the IPS. Sends SNMP traps when triggered by alert, status, and error events. NotificationApp uses the public domain SNMP agent. SNMP GETs provide information about the general health of the sensor.
NTP	Network Timing Protocol. Protocol built on top of TCP that ensures accurate local time-keeping with reference to radio and atomic clocks located on the Internet. This protocol is capable of synchronizing distributed clocks within milliseconds over long time periods.
NTP server	Network Timing Protocol server. A server that uses NTP. NTP is a protocol built on top of TCP that ensures accurate local time-keeping with reference to radio and atomic clocks located on the Internet. This protocol is capable of synchronizing distributed clocks within milliseconds over long time periods.
NVRAM	Non-Volatile Read/Write Memory. RAM that retains its contents when a unit is powered off.

O

OIR	online insertion and removal. Feature that permits you to add, replace, or remove cards without interrupting the system power, entering console commands, or causing other software or interfaces to shutdown.
OPS	Outbreak Prevention Service.

P

P2P	Peer-to-Peer. P2P networks use nodes that can simultaneously function as both client and server for the purpose of file sharing.
packet	Logical grouping of information that includes a header containing control information and (usually) user data. Packets most often are used to refer to network layer units of data. The terms datagram, frame, message, and segment also are used to describe logical information groupings at various layers of the OSI reference model and in various technology circles.
PAgP	Port Aggregation Control Protocol. PAgP aids in the automatic creation of EtherChannel links by exchanging PAgP packets between LAN ports. It is a Cisco-proprietary protocol.
PAM	Software module that provides AAA functionality to applications.
PAP	Password Authentication Protocol. Most commonly used RADIUS messaging protocol.
passive fingerprinting	Act of determining the OS or services available on a system from passive observation of network interactions.
Passive OS Fingerprinting	The sensor determines host operating systems by inspecting characteristics of the packets exchanged on the network.
PASV Port Spoof	An attempt to open connections through a firewall to a protected FTP server to a non-FTP port. This happens when the firewall incorrectly interprets an FTP 227 passive command by opening an unauthorized connection.
PAT	Port Address Translation. A more restricted translation scheme than NAT in which a single IP address and different ports are used to represent the hosts of a network.
patch release	Release that addresses defects identified in the update (minor, major, or service pack) binaries after a software release (service pack, minor, or major update) has been released.
PAWS	Protection Against Wrapped Sequence. Protection against wrapped sequence numbers in high performance TCP networks. See RFC 1323 .
PCI	Peripheral Component Interface. The most common peripheral expansion bus used on Intel-based computers.
PDU	protocol data unit. OSI term for packet. See also BPDU and packet.
PEP	Cisco Product Evolution Program. PEP is the UDI information that consists of the PID, the VID, and the SN of your sensor. PEP provides hardware version and serial number visibility through electronic query, product labels, and shipping items.
PER	packed encoding rules. Instead of using a generic style of encoding that encodes all types in a uniform way, PER specializes the encoding based on the data type to generate much more compact representations.
PFC	Policy Feature Card. An optional card on a Catalyst 6000 supervisor engine that supports VACL packet filtering.
PID	Product Identifier. The orderable product identifier that is one of the three parts of the UDI. The UDI is part of the PEP policy.

ping	packet internet groper. Often used in IP networks to test the reachability of a network device. It works by sending ICMP echo request packets to the target host and listening for echo response replies.
PKI	Public Key Infrastructure. Authentication of HTTP clients using the clients X.509 certificates.
Pluggable Authentication Modules	See PAM.
POST	Power-On Self Test. Set of hardware diagnostics that runs on a hardware device when that device is powered up.
Post-ACL	Designates an ACL from which ARC should read the ACL entries, and where it places entries after all deny entries for the addresses being blocked.
Pre-ACL	Designates an ACL from which ARC should read the ACL entries, and where it places entries before any deny entries for the addresses being blocked.
promiscuous delta	PD. A weight in the range of 0 to 30 configured per signature. This weight can be subtracted from the overall risk rating in promiscuous mode.
promiscuous mode	A passive interface for monitoring packets of the network segment. The sensing interface does not have an IP address assigned to it and is therefore invisible to attackers.

Q

Q.931	ITU-T specification for signaling to establish, maintain, and clear ISDN network connections.
QoS	quality of service. Measure of performance for a transmission system that reflects its transmission quality and service availability.

R

rack mounting	Refers to mounting a sensor in an equipment rack.
RADIUS	Remote Authentication Dial In User Service. A networking protocol that provides centralized AAA functionality for systems to connect and use a network service.
RAM	random-access memory. Volatile memory that can be read and written by a microprocessor.
RAS	Registration, Admission, and Status Protocol. Protocol that is used between endpoints and the gatekeeper to perform management functions. RAS signalling function performs registration, admissions, bandwidth changes, status, and disengage procedures between the VoIP gateway and the gatekeeper.
RBCP	Router Blade Control Protocol. RBCP is based on SCP, but modified specifically for the router application. It is designed to run over Ethernet interfaces and uses 802.2 SNAP encapsulation for messages.

reassemble	The putting back together of an IP datagram at the destination after it has been fragmented either at the source or at an intermediate node.
recovery package	An IPS package file that includes the full application image and installer used for recovery on sensors.
Regex	See regular expression.
regular expression	A mechanism by which you can define how to search for a specified sequence of characters in a data stream or file. Regular expressions are a powerful and flexible notation almost like a mini-programming language that allow you to describe text. In the context of pattern matching, regular expressions allow a succinct description of any arbitrary pattern.
Remote Authentication Dial In User Service	See RADIUS.
repackage release	A release that addresses defects in the packaging or the installer.
reputation	Similar to human social interaction, reputation is an opinion toward a device on the Internet. It enables the installed base of IPS sensors in the field to collaborate using the existing network infrastructure. A network device with reputation is most probably malicious or infected.
risk rating	RR. A risk rating is a value between 0 and 100 that represents a numerical quantification of the risk associated with a particular event on the network. The risk of the attack accounts for the severity, fidelity, relevance, and asset value of the attack, but not any response or mitigation actions. This risk is higher when more damage could be inflicted on your network.
RMA	Return Materials Authorization. The Cisco program for returning faulty hardware and obtaining a replacement.
ROMMON	Read-Only-Memory Monitor. ROMMON lets you TFTP system images onto the sensor for recovery purposes.
round-trip time	See RTT.
RPC	remote-procedure call. Technological foundation of client/server computing. RPCs are procedure calls that are built or specified by clients and are executed on servers, with the results returned over the network to the clients.
RTP	Real-Time Transport Protocol. Commonly used with IP networks. RTP is designed to provide end-to-end network transport functions for applications transmitting real-time data, such as audio, video, or simulation data, over multicast or unicast network services. RTP provides such services as payload type identification, sequence numbering, timestamping, and delivery monitoring to real-time applications.
RTT	round-trip time. A measure of the time delay imposed by a network on a host from the sending of a packet until acknowledgment of the receipt.
RU	rack unit. A rack is measured in rack units. An RU is equal to 44 mm or 1.75 inches.

S

SCADA	Supervisory Control and Data Acquisition. A type of ICS (Industrial Control System), which is essentially a network of computer-controlled devices.
SCEP	Simple Certificate Enrollment Protocol. The Cisco Systems PKI communication protocol that leverages existing technology by using PKCS#7 and PKCS#10. SCEP is the evolution of the enrollment protocol.
SCP	Switch Configuration Protocol. Cisco control protocol that runs directly over the Ethernet.
SDEE	Security Device Event Exchange. A product-independent standard for communicating security device events. It adds extensibility features that are needed for communicating events generated by various types of security devices.
SDEE Server	Accepts requests for events from remote clients.
Secure Shell Protocol	Protocol that provides a secure remote connection to a router through a Transmission Control Protocol (TCP) application.
security context	You can partition a single adaptive security appliance into multiple virtual devices, known as security contexts. Each context is an independent device, with its own security policy, interfaces, and administrators. Multiple contexts are similar to having multiple standalone devices. Many features are supported in multiple context mode, including routing tables, firewall features, IPS, and management.
sensing interface	The interface on the sensor that monitors the desired network segment. The sensing interface is in promiscuous mode; it has no IP address and is not visible on the monitored segment.
sensor	The sensor is the intrusion detection engine. It analyzes network traffic searching for signs of unauthorized activity.
SensorApp	A component of the IPS. Performs packet capture and analysis. SensorApp analyzes network traffic for malicious content. Packets flow through a pipeline of processors fed by a producer designed to collect packets from the network interfaces on the sensor. SensorApp is the standalone executable that runs Analysis Engine.
Service engine	Deals with specific protocols, such as DNS, FTP, H255, HTTP, IDENT, MS RPC, MS SQL, NTP, P2P, RPC, SMB, SNMP, SSH, and TNS.
service pack	Used for the release of defect fixes and for the support of new signature engines. Service packs contain all of the defect fixes since the last base version (minor or major) and any new defects fixes.
session command	Command used on routers and switches to provide either Telnet or console access to a module in the router or switch.
SFP	Small Form-factor Pluggable. Often refers to a fiber optic transceiver that adapts optical cabling to fiber interfaces. See GBIC for more information.
shared secret	A piece of data known only to the parties involved in a secure communication. The shared secret can be a password, a passphrase, a big number, or an array of randomly chosen bytes.
shun command	Enables a dynamic response to an attacking host by preventing new connections and disallowing packets from any existing connection. It is used by ARC when blocking with a PIX Firewall.

Signature Analysis Processor	A processor in the IPS. Dispatches packets to the inspectors that are not stream-based and that are configured for interest in the packet in process.
signature	A signature distills network information and compares it against a rule set that indicates typical intrusion activity.
signature engine	A component of the sensor that supports many signatures in a certain category. An engine is composed of a parser and an inspector. Each engine has a set of legal parameters that have allowable ranges or sets of values.
signature engine update	Executable file with its own versioning scheme that contains binary code to support new signature updates.
Signature Event Action Filter	Subtracts actions based on the signature event signature ID, addresses, and risk rating. The input to the Signature Event Action Filter is the signature event with actions possibly added by the Signature Event Action Override.
Signature Event Action Handler	Performs the requested actions. The output from Signature Event Action Handler is the actions being performed and possibly an evIdsAlert written to the Event Store.
Signature Event Action Override	Adds actions based on the risk rating value. Signature Event Action Override applies to all signatures that fall into the range of the configured risk rating threshold. Each Signature Event Action Override is independent and has a separate configuration value for each action type.
Signature Event Action Processor	Processes event actions. Event actions can be associated with an event risk rating threshold that must be surpassed for the actions to take place.
signature fidelity rating	SFR. A weight associated with how well a signature might perform in the absence of specific knowledge of the target. The signature fidelity rating is configured per signature and indicates how accurately the signature detects the event or condition it describes.
signature update	Executable file that contains a set of rules designed to recognize malicious network activities, such as worms, DDOS, viruses, and so forth. Signature updates are released independently, are dependent on a required signature engine version, and have their own versioning scheme.
Slave Dispatch Processor	A processor in the IPS. Process found on dual CPU systems.
SMB	Server Message Block. File-system protocol used in LAN manager and similar NOSs to package data and exchange information with other systems.
SMTP	Simple Mail Transfer Protocol. Internet protocol providing e-mail services.
SN	Serial Number. Part of the UDI. The SN is the serial number of your Cisco product.
SNAP	Subnetwork Access Protocol. Internet protocol that operates between a network entity in the subnetwork and a network entity in the end system. SNAP specifies a standard method of encapsulating IP datagrams and ARP messages on IEEE networks. The SNAP entity in the end system makes use of the services of the subnetwork and performs three key functions: data transfer, connection management, and QoS selection.
sniffing interface	See sensing interface.

SNMP	Simple Network Management Protocol. Network management protocol used almost exclusively in TCP/IP networks. SNMP provides a means to monitor and control network devices, and to manage configurations, statistics collection, performance, and security.
SNMP2	SNMP Version 2. Version 2 of the network management protocol. SNMP2 supports centralized and distributed network management strategies, and includes improvements in the SMI, protocol operations, management architecture, and security.
SNMP3	SNMP Version 3. Version 3 of the network management protocol. SNMPv3 adds security and remote configuration enhancements to SNMP, such as encryption of packets, message integrity, and authentication.
software bypass	Passes traffic through the IPS system without inspection.
source address	Address of a network device that is sending data.
SPAN	Switched Port Analyzer. Feature of the Catalyst 5000 switch that extends the monitoring abilities of existing network analyzers into a switched Ethernet environment. SPAN mirrors the traffic at one switched segment onto a predefined SPAN port. A network analyzer attached to the SPAN port can monitor traffic from any other Catalyst switched port.
spanning tree	Loop-free subset of a network topology.
SQL	Structured Query Language. International standard language for defining and accessing relational databases.
SRAM	Type of RAM that retains its contents for as long as power is supplied. SRAM does not require constant refreshing, like DRAM.
SSH	Secure Shell. A utility that uses strong authentication and secure communications to log in to another computer over a network.
SSL	Secure Socket Layer. Encryption technology for the Internet used to provide secure transactions, such as the transmission of credit card numbers for e-commerce.
Stacheldraht	A DDoS tool that relies on the ICMP protocol.
State engine	Stateful searches of HTTP strings.
Statistics Processor	A processor in the IPS. Keeps track of system statistics such as packet counts and packet arrival rates.
STP	Spanning Tree Protocol. A network protocol that ensures a loop-free topology for any bridged Ethernet local area network. STP prevents bridge loops and the broadcast radiation that results from them. Spanning tree also allows a network design to include spare (redundant) links to provide automatic backup paths if an active link fails without the danger of bridge loops or the need for manual enabling/disabling of these backup links.
Stream Reassembly Processor	A processor in the IPS. Reorders TCP streams to ensure the arrival order of the packets at the various stream-based inspectors. It is also responsible for normalization of the TCP stream. The normalizer engine lets you enable or disable alert and deny actions.
String engine	A signature engine that provides regular expression-based pattern inspection and alert functionality for multiple transport protocols, including TCP, UDP, and ICMP.

subsignature	A more granular representation of a general signature. It typically further defines a broad scope signature.
surface mounting	Refers to attaching rubber feet to the bottom of a sensor when it is installed on a flat surface. The rubber feet allow proper airflow around the sensor and they also absorb vibration so that the hard-disk drive is less impacted.
switch	Network device that filters, forwards, and floods frames based on the destination address of each frame. The switch operates at the data link layer of the OSI model.
SwitchApp	A component of the IPS. The IPS 4500 series sensors have a built in switch that provides external monitoring interfaces. The SwitchApp enables the InterfaceApp and sensor initialization scripts to communicate with and control the switch.
SYN flood	Denial of Service attack that sends a host more TCP SYN packets (request to synchronize sequence numbers, used when opening a connection) than the protocol implementation can handle.
system image	The full IPS application and recovery image used for reimaging an entire sensor.

T

TAC	A Cisco Technical Assistance Center. There are four TACs worldwide.
TACACS+	Terminal Access Controller Access Control System Plus. Proprietary Cisco enhancement to Terminal Access Controller Access Control System (TACACS). Provides additional support for authentication, authorization, and accounting.
target value rating	TVR. A weight associated with the perceived value of the target. Target value rating is a user-configurable value (zero, low, medium, high, or mission critical) that identifies the importance of a network asset (through its IP address).
TCP	Transmission Control Protocol. Connection-oriented transport layer protocol that provides reliable full-duplex data transmission. TCP is part of the TCP/IP protocol stack.
TCPDUMP	The TCPDUMP utility is a free network protocol analyzer for UNIX and Windows. It lets you examine data from a live network or from a capture file on disk. You can use different options for viewing summary and detail information for each packet. For more information, see http://www.tcpdump.org/ .
Telnet	Standard terminal emulation protocol in the TCP/IP protocol stack. Telnet is used for remote terminal connection, enabling users to log in to remote systems and use resources as if they were connected to a local system. Telnet is defined in RFC 854.
terminal server	A router with multiple, low speed, asynchronous ports that are connected to other serial devices. Terminal servers can be used to remotely manage network equipment, including sensors.
TFN	Tribe Flood Network. A common type of DoS attack that can take advantage of forged or rapidly changing source IP addresses to allow attackers to thwart efforts to locate or filter the attacks.
TFN2K	Tribe Flood Network 2000. A common type of DoS attack that can take advantage of forged or rapidly changing source IP addresses to allow attackers to thwart efforts to locate or filter the attacks.

TFTP	Trivial File Transfer Protocol. Simplified version of FTP that lets files be transferred from one computer to another over a network, usually without the use of client authentication (for example, username and password).
threat rating	TR. A threat rating is a value between 0 and 100 that represents a numerical decrease of the risk rating of an attack based on the response action that depicts the threat of an alert on the monitored network.
three-way handshake	Process whereby two protocol entities synchronize during connection establishment.
threshold	A value, either upper- or lower-bound that defines the maximum/minimum allowable condition before an alarm is sent.
Time Processor	A processor in the IPS. Processes events stored in a time-slice calendar. Its primary task is to make stale database entries expire and to calculate time-dependent statistics.
TLS	Transport Layer Security. The protocol used over stream transports to negotiate the identity of peers and establish encrypted communications.
TNS	Transparent Network Substrate. Provides database applications with a single common interface to all industry-standard network protocols. With TNS, database applications can connect to other database applications across networks with different protocols.
topology	Physical arrangement of network nodes and media within an enterprise networking structure.
TPKT	Transport Packet. RFC 1006-defined method of demarking messages in a packet. The protocol uses ISO transport services on top of TCP.
traceroute	Program available on many systems that traces the path a packet takes to a destination. It is used mostly to debug routing problems between hosts. A traceroute protocol is also defined in RFC 1393.
traffic analysis	Inference of information from observable characteristics of data flow(s), even when the data is encrypted or otherwise not directly available. Such characteristics include the identities and locations of the source(s) and destination(s), and the presence, amount, frequency, and duration of occurrence.
Traffic ICMP engine	Analyzes traffic from nonstandard protocols, such as TFN2K, LOKI, and DDOS.
trap	Message sent by an SNMP agent to an NMS, a console, or a terminal to indicate the occurrence of a significant event, such as a specifically defined condition or a threshold that was reached.
Trojan engine	Analyzes traffic from nonstandard protocols, such as BO2K and TFN2K.
trunk	Physical and logical connection between two switches across which network traffic travels. A backbone is composed of a number of trunks.
trusted certificate	Certificate upon which a certificate user relies as being valid without the need for validation testing; especially a public-key certificate that is used to provide the first public key in a certification path.
trusted key	Public key upon which a user relies; especially a public key that can be used as the first public key in a certification path.
tune	Adjusting signature parameters to modify an existing signature.

U

UDI	Unique Device Identifier. Provides a unique identity for every Cisco product. The UDI is composed of the PID, VID, and SN. The UDI is stored in the Cisco IPS ID PROM.
UDLD	UniDirectional Link Detection. Cisco proprietary protocol that allows devices connected through fiber-optic or copper Ethernet cables connected to LAN ports to monitor the physical configuration of the cables and detect when a unidirectional link exists. When a unidirectional link is detected, UDLD shuts down the affected LAN port and sends an alert, since unidirectional links can cause a variety of problems, such as, spanning tree topology loops.
UDP	User Datagram Protocol. Connectionless transport layer protocol in the TCP/IP protocol stack. UDP is a simple protocol that exchanges datagrams without acknowledgments or guaranteed delivery, requiring that error processing and retransmission be handled by other protocols. UDP is defined in RFC 768.
unblock	To direct a router to remove a previously applied block.
UniDirectional Link Detection	See UDLD.
unvirtualized sensing interface	An unvirtualized sensing interface has not been divided into subinterfaces and the entire interfaces can be associated with at most one virtual sensor.
UPS	Uninterruptable Power Source.
UTC	Coordinated Universal Time. Time zone at zero degrees longitude. Formerly called Greenwich Mean Time (GMT) and Zulu time.
UTF-8	8-bit Unicode Transformation Format. A variable-length character encoding for Unicode. UTF-8 can represent every character in the Unicode character set and is backwards-compatible with ASCII.

V

VACL	VLAN ACL. An ACL that filters all packets (both within a VLAN and between VLANs) that pass through a switch. Also known as security ACLs.
VID	Version identifier. Part of the UDI.
VIP	Versatile Interface Processor. Interface card used in Cisco 7000 and Cisco 7500 series routers. The VIP provides multilayer switching and runs Cisco IOS. The most recent version of the VIP is VIP2.
virtual sensor	A logical grouping of sensing interfaces and the configuration policy for the signature engines and alarm filters to apply to them. In other words, multiple virtual sensors running on the same appliance, each configured with different signature behavior and traffic feeds.
virtualized sensing interface	A virtualized interface has been divided into subinterfaces each of which consists of a group of VLANs. You can associate a virtual sensor with one or more subinterfaces so that different intrusion prevention policies can be assigned to those subinterfaces. You can virtualize both physical and inline interfaces.

virus	Hidden, self-replicating section of computer software, usually malicious logic, that propagates by infecting—that is, inserting a copy of itself into and becoming part of—another program. A virus cannot run by itself; it requires that its host program be run to make the virus active.
VLAN	Virtual Local Area Network. Group of devices on one or more LANs that are configured (using management software) so that they can communicate as if they were attached to the same wire, when in fact they are located on a number of different LAN segments. Because VLANs are based on logical instead of physical connections, they are extremely flexible.
VTP	VLAN Trunking Protocol. Cisco Layer 2 messaging protocol that manages the addition, deletion, and renaming of VLANs on a network-wide basis.
VoIP	Voice over IP. The capability to carry normal telephony-style voice over an IP-based internet with POTS-like functionality, reliability, and voice quality. VoIP enables a router to carry voice traffic (for example, telephone calls and faxes) over an IP network. In VoIP, the DSP segments the voice signal into frames, which then are coupled in groups of two and stored in voice packets. These voice packets are transported using IP in compliance with ITU-T specification H.323.
VPN	Virtual Private Network(ing). Enables IP traffic to travel securely over a public TCP/IP network by encrypting all traffic from one network to another. A VPN uses “tunneling” to encrypt all information at the IP level.
VTP	VLAN Trunking Protocol. A Cisco Layer 2 messaging protocol that manages the addition, deletion, and renaming of VLANs on a network-wide basis.
vulnerability	One or more attributes of a computer or a network that permit a subject to initiate patterns of misuse on that computer or network.

W

WAN	wide-area network. Data communications network that serves users across a broad geographic area and often uses transmission devices provided by common carriers. Frame Relay, SMDS, and X.25 are examples of WANs.
watch list rating	WLR. A weight associated with the CSA MC watch list in the range of 0 to 100 (CSA MC only uses the range 0 to 35).
Web Server	A component of the IPS. Waits for remote HTTP client requests and calls the appropriate servlet application.
WHOIS	A TCP-based query/response protocol used for querying an official database to determine the owner of a domain name or an IP address.
Wireshark	Wireshark is a free network protocol analyzer for UNIX and Windows. It lets you examine data from a live network or from a capture file on disk. You can interactively browse the capture data, viewing summary and detail information for each packet. Wireshark has several powerful features, including a rich display filter language and the ability to view the reconstructed stream of a TCP session. For more information, see http://www.wireshark.org .
worm	A computer program that can run independently, can propagate a complete working version of itself onto other hosts on a network, and can consume computer resources destructively.

X

- X.509** Standard that defines information contained in a certificate.
- XML** eXtensible Markup Language. Textual file format used for data interchange between heterogeneous hosts.
- XPI** Cross Packet Inspection. Technology used by TCP that allows searches across packets to achieve packet and payload reassembly.

Z

- zone** A set of destination IP addresses sorted into an internal, illegal, or external zone used by anomaly detection.



Numerics

802.1q encapsulation for VLAN groups [5-27](#)

A

AAA authentication

 configuring [4-23](#)

AAA RADIUS

 functionality [4-30](#)

 limitations [4-30](#)

accessing

 IPS software [20-2](#)

 service account [4-28, C-5](#)

access-list command [4-6](#)

access list misconfiguration [C-26](#)

access lists

 changing [4-7](#)

 configuring [4-7](#)

account locking

 configuring [4-34](#)

 security [4-34](#)

account unlocking configuring [4-35](#)

ACLs

 described [14-3](#)

 Post-Block [14-22, 14-23](#)

 Pre-Block [14-22, 14-23](#)

adding

 denied attackers [8-36](#)

 event action overrides [8-18](#)

 external product interfaces [11-5](#)

 global parameters [6-12](#)

 hosts to the SSH known hosts list [4-47, 4-48](#)

 login banners [4-9](#)

 signature variables [7-10](#)

 target value ratings [8-16](#)

 trusted hosts [4-53, 4-55](#)

 users [4-18, 4-19, 4-31, 4-32](#)

 virtual sensors [6-6, 6-9](#)

 virtual sensors (ASA 5500-X IPS SSP) [18-5](#)

 virtual sensors (ASA 5585-X IPS SSP) [19-5](#)

Address Resolution Protocol. See ARP.

administrative tasks notes and caveats [17-2](#)

administrator role privileges [1-3](#)

aggregation

 alert frequency [8-34](#)

 operating modes [8-34](#)

AIC engine

 AIC FTP [B-11](#)

 AIC FTP engine parameters (table) [B-13](#)

 AIC HTTP [B-11](#)

 AIC HTTP engine parameters (table) [B-12](#)

 described [B-11](#)

 features [B-11](#)

 signature categories [7-23](#)

AIC policy enforcement

 default configuration [7-23, B-11](#)

 described [7-23, B-11](#)

 sensor oversubscription [7-23, B-11](#)

Alarm Channel

 described [8-3, A-26](#)

 risk rating [10-6](#)

alert and log actions (list) [8-5](#)

alert-frequency command [7-12](#)

alert frequency modes [B-7](#)

alert-severity command [7-14](#)

- alert severity configuring [7-14](#)
- allocate-ips command [18-4, 19-4](#)
 - ASA 5500-X IPS SSP [18-23](#)
- allow-sensor-block command [14-8](#)
- alternate TCP reset interface
 - configuration restrictions [5-9](#)
 - designating [5-5](#)
 - restrictions [5-3](#)
- Analysis Engine
 - described [6-2](#)
 - error messages [C-23](#)
 - IDM exits [C-54](#)
 - sensing interfaces [5-4](#)
 - verify it is running [C-20](#)
 - virtual sensors [6-2](#)
- anomaly detection [9-1](#)
 - asymmetric traffic [9-1, 9-2](#)
 - caution [9-1, 9-2](#)
 - configuration sequence [9-5](#)
 - default anomaly detection configuration [9-4](#)
 - default configuration (example) [9-4](#)
 - described [9-2](#)
 - detect mode [9-4](#)
 - enabling [9-8](#)
 - event actions [9-6, B-70](#)
 - inactive mode [9-4](#)
 - learning accept mode [9-3](#)
 - learning process [9-3](#)
 - limiting false positives [9-37](#)
 - protocols [9-3](#)
 - signatures (table) [9-6, B-71](#)
 - signatures described [9-6](#)
 - worms
 - attacks [9-37](#)
 - described [9-3](#)
 - zones [9-4](#)
- anomaly detection disabling [9-49, C-19](#)
- anomaly-detection load command [9-41](#)
- anomaly detection operational settings
 - configuring [9-11, 9-38](#)
 - described [9-10](#)
- anomaly detection policies
 - copying [9-9](#)
 - creating [9-9](#)
 - deleting [9-9](#)
 - displaying [9-9](#)
 - editing [9-9](#)
 - lists [17-29](#)
- anomaly-detection save command [9-41](#)
- anomaly detection statistics
 - clearing [9-47](#)
 - displaying [9-47](#)
- Anomaly Detection zones
 - illegal [9-20](#)
 - internal [9-12](#)
- appliances
 - GRUB menu [17-3, C-8](#)
 - initializing [3-9](#)
 - logging in [2-2](#)
 - password recovery [17-3, C-8](#)
 - resetting [17-46](#)
 - setting system clock [4-39, 17-27](#)
 - terminal servers
 - described [2-3, 21-16](#)
 - setting up [2-3, 21-16](#)
 - time sources [4-36, C-15](#)
 - upgrading recovery partition [21-7](#)
- Application Inspection and Control. See AIC.
- application partition
 - described [A-4](#)
 - image recovery [21-14](#)
- application-policy command [7-23](#)
- application policy configuring [7-24](#)
- application policy enforcement described [7-23, B-11](#)
- applications in XML format [A-4](#)
- applying
 - software updates [C-51](#)

- threat profiles [7-6](#)
- ARC
 - ACLs [14-22, A-14](#)
 - authentication [A-15](#)
 - blocking
 - connection-based [A-17](#)
 - response [A-13](#)
 - unconditional blocking [A-17](#)
 - blocking application [14-2](#)
 - blocking not occurring for signature [C-41](#)
 - Catalyst switches
 - VACL commands [A-19](#)
 - VACLs [A-16, A-19](#)
 - VLANs [A-16](#)
 - checking status [14-4, 14-5](#)
 - described [A-4](#)
 - design [14-2](#)
 - device access issues [C-39](#)
 - enabling SSH [C-41](#)
 - features [A-14](#)
 - firewalls
 - AAA [A-18](#)
 - connection blocking [A-18](#)
 - NAT [A-18](#)
 - network blocking [A-18](#)
 - postblock ACL [A-16](#)
 - preblock ACL [A-16](#)
 - shun command [A-18](#)
 - TACACS+ [A-18](#)
 - formerly Network Access Controller [14-1](#)
 - functions [14-2, A-12](#)
 - illustration [A-13](#)
 - inactive state [C-37](#)
 - interfaces [A-14](#)
 - maintaining states [A-16](#)
 - master blocking sensors [A-14](#)
 - maximum blocks [14-2](#)
 - misconfigured master blocking sensor [C-42](#)
 - nac.shun.txt file [A-16](#)
 - NAT addressing [A-15](#)
 - number of blocks [A-15](#)
 - postblock ACL [A-16](#)
 - preblock ACL [A-16](#)
 - prerequisites [14-6](#)
 - rate limiting [14-4](#)
 - responsibilities [A-13](#)
 - single point of control [A-15](#)
 - SSH [A-14](#)
 - supported devices [14-6, A-15](#)
 - Telnet [A-14](#)
 - troubleshooting [C-35](#)
 - VACLs [A-14](#)
 - verifying device interfaces [C-40](#)
 - verifying status [C-36](#)
- ARP
 - Layer 2 signatures [B-14](#)
 - protocol [B-14](#)
- ARP spoof tools
 - dsniff [B-14](#)
 - ettercap [B-14](#)
- ASA 5500 AIP SSC-5
 - time sources [4-36](#)
- ASA 5500 AIP SSM
 - IPS reloading messages [C-67, C-73](#)
 - time sources [4-36](#)
- ASA 5500-X IPS SSP
 - assigning virtual sensors [18-7](#)
 - bypass mode [18-9, 19-11](#)
 - creating virtual sensors [18-5](#)
 - Deny Connection Inline [18-11](#)
 - Deny Packet Inline [18-11](#)
 - IPS reloading messages [18-11, C-67, C-73](#)
 - logging in [2-4, 3-14](#)
 - memory usage [17-16, 18-11, C-66](#)
 - memory usage values (table) [17-16, 18-11, C-66](#)
 - no CDP mode support [5-45](#)
 - Normalizer engine [18-9, C-65](#)
 - notes and caveats [18-1](#)

- password recovery [17-4, C-10](#)
 - Reset TCP Connection [18-11](#)
 - resetting the password [17-5, C-10](#)
 - sensing interface [18-4](#)
 - session command [2-4, 3-14](#)
 - sessioning in [2-4, 3-14](#)
 - show module command [18-3](#)
 - sw-module module 1 recover configure [18-12](#)
 - sw-module module slot_number password-reset [18-12](#)
 - sw-module module slot_number reload [18-12](#)
 - sw-module module slot_number reset [18-12](#)
 - sw-module module slot_number shutdown [18-12](#)
 - task sequence [18-2](#)
 - TCP reset differences [8-7](#)
 - TCP reset packets [18-11](#)
 - time sources [4-36, C-15](#)
 - verifying initialization [18-3](#)
 - virtual sensors
 - assigning policies [18-4, 19-5](#)
 - assigning the interface [18-4, 19-5](#)
 - virtual sensor sequence [18-4, 19-5](#)
- ASA 5585-X IPS SSP
- assigning virtual sensors [19-8](#)
 - bypass mode [19-10](#)
 - creating virtual sensors [19-5](#)
 - Deny Connection Inline [19-11](#)
 - Deny Packet Inline [19-11](#)
 - hw-module module 1 recover configure [19-13](#)
 - hw-module module slot_number password-reset [19-12](#)
 - hw-module module slot_number recover boot [19-12](#)
 - hw-module module slot_number recover stop [19-12](#)
 - hw-module module slot_number reload [19-12](#)
 - hw-module module slot_number reset [19-12](#)
 - hw-module module slot_number shutdown [19-12](#)
 - installing system image [21-24](#)
 - interfaces
 - command and control [19-4](#)
 - described [19-4](#)
 - port numbers [19-4](#)
 - sensing [19-4](#)
 - slot numbers [19-4](#)
 - IPS reloading messages [19-12, C-67, C-73](#)
 - logging in [2-5, 3-15](#)
 - no CDP mode support [5-45, 19-1](#)
 - Normalizer engine [19-10, C-72](#)
 - notes and caveats [19-1](#)
 - password recovery [17-6, C-12](#)
 - Reset TCP Connection [19-11](#)
 - resetting the password [17-7, C-12](#)
 - session command [2-5, 3-15](#)
 - sessioning in [2-5, 3-15](#)
 - task sequence [18-2, 19-2](#)
 - TCP reset differences [8-7](#)
 - TCP reset packets [19-11](#)
 - time sources [4-36, C-15](#)
 - virtual sensors
 - assigning policies [18-4, 19-5](#)
 - assigning the interface [18-4, 19-5](#)
 - sequence [19-5](#)
- ASA IPS modules
- jumbo packet count [5-46, 18-10, 19-11, C-66, C-73](#)
- ASDM
- resetting passwords [17-6, 17-8, C-11, C-13](#)
- assigning
- interfaces to virtual sensors (ASA 5500-X IPS SSP) [18-4, 19-5](#)
 - interfaces to virtual sensors (ASA 5585-X IPS SSP) [18-4, 19-5](#)
 - policies to virtual sensors (ASA 5500-X IPS SSP) [18-4, 19-5](#)
 - policies to virtual sensors (ASA 5585-X IPS SSP) [18-4, 19-5](#)
- assigning interfaces to virtual sensors [6-5](#)
- assigning policies to virtual sensors [6-5](#)
- asymmetric mode
- described [6-4](#)
 - normalization [6-4](#)

- asymmetric traffic
 - anomaly detection [9-1, 9-2](#)
 - caution [9-1, 9-2](#)
 - asymmetric traffic and disabling anomaly detection [9-48, C-18](#)
 - Atomic ARP engine
 - described [B-14](#)
 - parameters (table) [B-14](#)
 - Atomic IP Advanced engine
 - described [B-15](#)
 - parameters (table) [B-17](#)
 - restrictions [B-16](#)
 - Atomic IP engine
 - described [B-25](#)
 - parameters (table) [B-25](#)
 - Atomic IPv6 engine
 - described [B-28](#)
 - Neighborhood Discovery protocol [B-29](#)
 - signatures [B-29](#)
 - attack relevance rating
 - calculating risk rating [8-14](#)
 - described [8-14, 8-27](#)
 - Attack Response Controller
 - described [A-4](#)
 - formerly known as Network Access Controller [A-4](#)
 - See ARC
 - attack severity rating
 - calculating risk rating [8-14](#)
 - described [8-14](#)
 - attempt limit
 - RADIUS [C-20](#)
 - attemptLimit command [4-34](#)
 - audit mode
 - described [10-9](#)
 - testing global correlation [10-9](#)
 - authenticated NTP [4-2, 4-36, 4-45, C-15](#)
 - authentication
 - local [4-20](#)
 - RADIUS [4-20](#)
 - AuthenticationApp
 - authenticating users [A-20](#)
 - described [A-4](#)
 - login attempt limit [A-20](#)
 - method [A-20](#)
 - responsibilities [A-20](#)
 - secure communications [A-21](#)
 - sensor configuration [A-20](#)
 - Authentication pane
 - user roles [A-30](#)
 - authorized keys
 - defining [4-49](#)
 - RSA authentication [4-49](#)
 - automatic setup [3-3](#)
 - automatic update
 - DNS servers [4-11](#)
 - immediate [21-11](#)
 - proxy server [4-11](#)
 - automatic upgrade
 - information required [21-8](#)
 - troubleshooting [C-51](#)
 - autoupdatenow command [21-11](#)
 - auto-upgrade-option command [21-8](#)
-
- B**
- backing up
 - configuration [16-24, C-2](#)
 - current configuration [16-23, C-4](#)
 - BackOrifice. See BO.
 - BackOrifice 2000. See BO2K.
 - backup-config command [16-20](#)
 - banner login command [17-20](#)
 - basic setup [3-5](#)
 - block connection command [14-33](#)
 - block-enable command [14-9](#)
 - block hosts command [14-31](#)
 - blocking
 - addresses never to block [14-19](#)

- block time [14-13](#)
- connection [14-33](#)
- described [14-2](#)
- disabling [14-10](#)
- hosts [14-31](#)
- list of blocked hosts [14-34](#)
- managing firewalls [14-27](#)
- managing routers [14-23](#)
- managing switches [14-26](#)
- master blocking sensor [14-28](#)
- maximum entries [14-11](#)
- necessary information [14-3](#)
- notes and caveats [14-1](#)
- prerequisites [14-6](#)
- properties [14-7](#)
- sensor block itself [14-8](#)
- show statistics [14-34](#)
- supported devices [14-6](#)
- types [14-3](#)
- user profiles [14-20](#)
- blocking not occurring for signature [C-41](#)
- block network command [14-32](#)
- BO
 - described [B-73](#)
 - Trojans [B-73](#)
- BO2K
 - described [B-73](#)
 - Trojans [B-73](#)
- BST
 - described [C-1](#)
 - URL [C-1](#)
- Bug Search Tool. See BST.
- bypass mode
 - ASA 5500-X IPS SSP [18-9, 19-11](#)
 - ASA 5585-X IPS SSP [19-10](#)
 - configuring [5-43](#)
 - described [5-42](#)
- bypass-option command [5-43](#)

C

- calculating risk rating
 - attack relevance rating [8-14](#)
 - attack severity rating [8-14](#)
 - promiscuous delta [8-14](#)
 - signature fidelity rating [8-14](#)
 - target value rating [8-14](#)
 - watch list rating [8-14](#)
- cannot access sensor [C-24](#)
- capture packet files
 - notes and caveats [13-1](#)
- capturing live traffic [13-5](#)
- caution for clearing databases [17-10](#)
- CDP mode
 - ASA 5500-X IPS SSP [5-45](#)
 - ASA 5585-X IPS SSP [5-45, 19-1](#)
 - configuring [5-45](#)
 - described [5-45](#)
 - interfaces [5-45](#)
- certificates (IDM) [4-52](#)
- changing [4-30](#)
 - access lists [4-7](#)
 - CLI inactivity timeout [4-14](#)
 - FTP timeout [4-8](#)
 - host IP address [4-4](#)
 - hostname [4-3](#)
 - passwords [4-30](#)
 - privilege [4-31](#)
 - web server settings [4-16](#)
- cidDump obtaining information [C-102](#)
- CIDEE
 - defined [A-34](#)
 - example [A-34](#)
 - IPS extensions [A-34](#)
 - protocol [A-34](#)
 - supported IPS events [A-34](#)
- cisco
 - default password [2-2](#)

- default username [2-2](#)
- Cisco.com
 - accessing software [20-2](#)
 - downloading software [20-1](#)
 - software downloads [20-1](#)
- Cisco Bug Search Tool
 - described [C-1](#)
- Cisco Discovery Protocol. See CDP.
- Cisco IOS rate limiting [14-4](#)
- cisco-security-agents-mc-settings command [11-4](#)
- Cisco Security Intelligence Operations
 - described [20-7](#)
 - URL [20-7](#)
- Cisco Services for IPS
 - service contract [4-59](#)
 - supported products [4-59](#)
- clear database command [17-10](#)
- clear denied-attackers command [8-37, 17-27](#)
- clear events command [4-37, 8-42, 17-25, C-17, C-102](#)
- clearing
 - anomaly detection statistics [9-47](#)
 - denied attackers statistics [8-37, 17-27](#)
 - events [8-42, 17-25, C-102](#)
 - global correlation statistics [10-14](#)
 - OS IDs [8-32](#)
 - sensor databases [17-10](#)
 - statistics [17-30, C-83](#)
- clearing databases caution [17-10](#)
- clear line command [17-20](#)
- clear os-identification command [8-32](#)
- CLI
 - command line editing [1-6](#)
 - command modes [1-7](#)
 - default keywords [1-10](#)
 - described [A-4, A-30](#)
 - error messages [D-1](#)
 - generic commands [1-10](#)
 - password recovery [17-8, C-14](#)
 - regular expression syntax [1-8](#)
 - CLI behavior [1-5](#)
 - case sensitivity [1-6](#)
 - display options [1-6](#)
 - help [1-5](#)
 - prompts [1-5](#)
 - recall [1-5](#)
 - tab completion [1-5](#)
 - client manifest described [A-28](#)
 - CLI guide introduction [1-1](#)
 - CLI inactivity timeout
 - configuring [4-14](#)
 - described [4-14](#)
 - cli-inactivity-timeout command [4-14](#)
 - CLI session termination [17-21](#)
 - clock set command [4-38, 17-26](#)
 - CollaborationApp described [A-4, A-28](#)
 - command and control interface
 - described [5-3](#)
 - list [5-3](#)
 - command and control interface described (ASA 5585-X IPS SSP) [19-4](#)
 - command line editing (table) [1-6](#)
 - command modes [1-7](#)
 - anomaly detection configuration [1-7](#)
 - event action rules configuration [1-7](#)
 - EXEC [1-7](#)
 - global configuration [1-7](#)
 - privileged EXEC [1-7](#)
 - service mode configuration [1-7](#)
 - signature definition configuration [1-7](#)
 - commands [15-8, 18-3, 19-3](#)
 - access-list [4-6](#)
 - alert-frequency [7-12](#)
 - alert-severity [7-14](#)
 - allocate-ips [18-4, 19-4](#)
 - allow-sensor-block [14-8](#)
 - anomaly-detection load [9-41](#)
 - anomaly-detection save [9-41](#)
 - application-policy [7-23](#)

- attemptLimit [4-34](#)
- autoupdatenow [21-11](#)
- auto-upgrade-option [21-8](#)
- backup-config [16-20](#)
- banner login [17-20](#)
- block connection [14-33](#)
- block-enable [14-9](#)
- block hosts [14-31](#)
- block network [14-32](#)
- bypass-option [5-43](#)
- cisco-security-agents-mc-settings [11-4](#)
- clear database [17-10](#)
- clear denied-attackers [8-37, 17-27](#)
- clear events [4-37, 8-42, 17-25, C-17, C-102](#)
- clear line [17-20](#)
- clear os-identification [8-32](#)
- cli-inactivity-timeout [4-14](#)
- clock set [4-38, 17-26](#)
- copy ad-knowledge-base [9-42](#)
- copy anomaly-detection [9-8](#)
- copy backup-config [16-21, C-3](#)
- copy current-config [16-21, C-3](#)
- copy event-action-rules [8-8](#)
- copy iplog [12-7](#)
- copy license-key [4-60](#)
- copy packet-file [13-6](#)
- copy signature-definition [7-2](#)
- current-config [16-20](#)
- default service anomaly-detection [9-9](#)
- default service event-action-rules [8-8](#)
- default service signature-definition [7-2](#)
- deny attacker [8-36](#)
- downgrade [21-12](#)
- enable-acl-logging [14-14](#)
- enable-detail-traps [15-8](#)
- enable-nvram-write [14-15](#)
- erase [16-24](#)
- erase ad-knowledge-base [9-42](#)
- erase license-key [4-62](#)
- erase packet-file [13-7](#)
- event-action [7-20](#)
- event-action-rules-configurations [17-29](#)
- event-counter [7-15](#)
- external-zone [9-28](#)
- filters [8-22](#)
- fragment-reassembly [7-36](#)
- ftp-timeout [4-8](#)
- global-block-timeout [8-34, 14-13](#)
- global-deny-timeout [8-34](#)
- global-filters-status [8-34](#)
- global-metaevent-status [8-34](#)
- global-overrides-status [8-34](#)
- global-parameters [6-12](#)
- global-summarization [8-35](#)
- health-monitor [10-8](#)
- host-ip [4-4](#)
- host-name [4-3](#)
- hw-module module 1 recover configure [19-13](#)
- hw-module module slot_number password-reset [17-6, 19-12, C-12](#)
- hw-module module slot_number recover boot [19-12](#)
- hw-module module slot_number recover stop [19-12](#)
- hw-module module slot_number reload [19-12](#)
- hw-module module slot_number reset [19-12](#)
- hw-module module slot_number shutdown [19-12](#)
- ignore [9-10](#)
- illegal-zone [9-20](#)
- inline-interfaces [5-17](#)
- interface-notifications [5-44](#)
- internal-zone [9-12](#)
- ip-log [7-45](#)
- iplog [12-3](#)
- ip-log-bytes [12-2](#)
- ip-log-packets [12-2](#)
- iplog-status [12-5](#)
- ip-log-time [12-2](#)
- ipv6-target-value [8-15](#)
- learning-accept-mode [9-38](#)

- list anomaly-detection-configurations [9-9, 17-29](#)
- list event-action-rules-configurations [8-8](#)
- list signature-definition-configurations [7-2](#)
- log-all-block-events-and-errors [14-16](#)
- login-banner-text [4-9](#)
- max-block-entries [14-11](#)
- max-denied-attackers [8-35](#)
- max-interfaces [14-17](#)
- more [16-20](#)
- more current-config [16-1](#)
- never-block-hosts [14-19](#)
- never-block-networks [14-19](#)
- no iplog [12-6](#)
- no ipv6-target-value [8-15](#)
- no service anomaly-detection [9-9](#)
- no service event-action-rules [8-8](#)
- no service signature-definition [7-2](#)
- no target-value [8-15](#)
- no variables [8-12](#)
- os-identifications [8-28](#)
- other [9-18, 9-26, 9-34](#)
- overrides [8-18](#)
- packet capture [13-4](#)
- packet-display [13-2](#)
- password [4-18, 4-30](#)
- permit-packet-logging [4-26](#)
- physical-interfaces [5-11, 5-22, 5-28](#)
- ping [17-45](#)
- privilege [4-18, 4-31](#)
- rename ad-knowledge-base [9-42](#)
- reset [17-46](#)
- service anomaly-detection [9-8](#)
- service event-action-rules [8-8](#)
- service signature-definition [7-2](#)
- setup [3-1, 3-5, 3-9](#)
- show ad-knowledge-base diff [9-44, 9-45](#)
- show ad-knowledge-base files [9-40, 9-41](#)
- show clock [4-37, 17-25](#)
- show configuration [16-1](#)
- show context [18-7, 19-8](#)
- show events [8-39, 17-22, C-99](#)
- show health [10-9, 17-19, C-74](#)
- show history [17-47](#)
- show inspection-load [17-12](#)
- show interfaces [5-46](#)
- show interfaces-history [5-48, C-96](#)
- show inventory [17-47](#)
- show lacp [5-38](#)
- show module 1 details [19-13, C-57, C-67](#)
- show os-identification [8-32](#)
- show settings [16-3, 16-18, 17-9, 17-49, C-14](#)
- show statistics [14-34, 17-30, C-83](#)
- show statistics anomaly-detection [9-47](#)
- show statistics denied-attackers [8-37, 17-27](#)
- show statistics virtual-sensor [17-30, C-23, C-83](#)
- show tech-support [17-42, C-75](#)
- show tls trusted-root-certificates [4-57](#)
- show users [4-32](#)
- show version [17-43, C-79](#)
- sig-fidelity-rating [7-17, 7-19](#)
- signature-definition-configurations [17-29](#)
- snmp-agent-port [15-3](#)
- snmp-agent-protocol [15-3](#)
- ssh authorized-key [4-49](#)
- ssh-generate-key [4-50](#)
- ssh host-key [4-47](#)
- sshv1-fallback [4-13](#)
- status [7-18](#)
- stream-reassembly [7-44](#)
- strict-tls-server-validation [4-54](#)
- subinterface-type [5-22, 5-28](#)
- summertime-option non-recurring [4-41](#)
- summertime-option recurring [4-39](#)
- sw-module module 1 recover configure [18-12](#)
- sw-module module slot_number password-reset [17-5, 18-12, C-10](#)
- sw-module module slot_number reload [18-12](#)
- sw-module module slot_number reset [18-12](#)

- sw-module module slot_number shutdown [18-12](#)
- target-value [8-15](#)
- tcp [9-13, 9-21, 9-29](#)
- telnet-option [4-5](#)
- terminal [17-21](#)
- threat-profile [7-5](#)
- time-zone-settings [4-43](#)
- tls generate-key [4-58](#)
- tls trusted-host [4-53, 4-55](#)
- trace [17-48](#)
- trap-community-name [15-8](#)
- trap-destinations [15-8](#)
- udp [9-15, 9-24, 9-32](#)
- unlock user username [4-35](#)
- upgrade [21-4, 21-6](#)
- username [4-18](#)
- user-profile [14-20](#)
- variables [7-10, 8-12](#)
- virtual-sensor name [6-5, 18-4, 19-5](#)
- worm-timeout [9-10](#)
- comparing KBs [9-44](#)
- configuration files
 - backing up [16-24, C-2](#)
 - merging [16-24, C-2](#)
- configuration restrictions
 - alternate TCP reset interface [5-9](#)
 - inline interface pairs [5-9](#)
 - inline VLAN pairs [5-9](#)
 - interfaces [5-8](#)
 - physical interfaces [5-8](#)
 - VLAN groups [5-10](#)
- configuration sequence
 - ASA 5500-X IPS SSP [18-2](#)
 - ASA 5585-X IPS SSP [18-2, 19-2](#)
- configured OS mapping (example) [8-28](#)
- configuring
 - AAA authentication [4-23](#)
 - access lists [4-7](#)
 - account locking [4-34](#)
 - account unlocking [4-35](#)
 - ACL logging [14-14](#)
 - alert frequency parameters [7-13](#)
 - alert severity [7-14](#)
 - anomaly detection operational settings [9-11, 9-38](#)
 - application policy [7-24, 7-33](#)
 - automatic IP logging [12-3](#)
 - automatic upgrades [21-9](#)
 - blocking
 - firewalls [14-27](#)
 - routers [14-23](#)
 - switches [14-26](#)
 - time [14-13](#)
 - bypass mode [5-43](#)
 - CDP mode [5-45](#)
 - cli-inactivity-timeout [4-14](#)
 - connection blocking [14-33](#)
 - CSA MC IPS interfaces [11-4](#)
 - DNS servers [4-11](#)
 - event action filters [8-23](#)
 - event actions [7-21](#)
 - event counter [7-16](#)
 - external zone [9-29](#)
 - ftp-timeout [4-8](#)
 - global correlation [10-10, 10-12](#)
 - health statistics [17-16](#)
 - host blocks [14-31](#)
 - host IP address [4-4](#)
 - hostname [4-3](#)
 - hosts never to block [14-19](#)
 - illegal zone [9-20](#)
 - inline interface pairs [5-17](#)
 - inline VLAN groups [5-29](#)
 - inline VLAN pairs [5-23](#)
 - internal zone [9-12](#)
 - IP fragment reassembly [7-37](#)
 - IP fragment reassembly parameters [7-36, 7-43](#)
 - IP logging [7-45](#)
 - LACP [5-37](#)

- logging all blocking events and errors [14-16](#)
- logical devices [14-20](#)
- login-banner-text [4-9](#)
- manual IP logging [12-4](#)
- master blocking sensor [14-29](#)
- maximum block entries [14-12](#)
- maximum blocking interfaces [14-18](#)
- maximum denied attackers [8-35](#)
- Meta Event Generator [8-35](#)
- network blocks [14-32](#)
- networks never to block [14-19](#)
- NTP servers [4-44](#)
- NVRAM write [14-15](#)
- OS maps [8-30](#)
- other protocols
 - external zone [9-35](#)
 - illegal zone [9-26](#)
 - internal zone [9-18](#)
- packet command restrictions [4-27](#)
- password policy [4-33](#)
- passwords [4-30](#)
- physical interfaces [5-12](#)
- privilege [4-31](#)
- proxy servers [4-11](#)
- sensor sequence [1-2](#)
- sensor to block itself [14-8](#)
- sensor to use NTP [4-45](#)
- signature fidelity rating [7-17, 7-19](#)
- sshv1-fallback [4-13](#)
- status [7-18](#)
- summarizer [8-35](#)
- summertime
 - non-recurring [4-41](#)
 - recurring [4-39](#)
- TCP
 - external zone [9-30](#)
 - illegal zone [9-22](#)
 - internal zone [9-13](#)
- TCP stream reassembly [7-44](#)
- telnet-option [4-5](#)
- threat profiles [7-6](#)
- time zone settings [4-43](#)
- traffic flow notifications [5-44](#)
- UDP
 - external zone [9-32](#)
 - illegal zone [9-24](#)
 - internal zone [9-16](#)
- upgrades [21-5](#)
- user profiles [14-20](#)
- users (SNMPv3) [15-4](#)
- web server settings [4-15](#)
- configuring interfaces
 - notes and caveats [5-1](#)
 - sequence [5-10](#)
- control transactions
 - characteristics [A-9](#)
 - request types [A-9](#)
- copy ad-knowledge-base command [9-42](#)
- copy anomaly-detection command [9-8](#)
- copy backup-config command [16-21, C-3](#)
- copy command syntax [9-42](#)
- copy current-config command [16-21, C-3](#)
- copy event-action-rules command [8-8](#)
- copying
 - anomaly detection policies [9-9](#)
 - event action rules policies [8-9](#)
 - IP log files [12-7](#)
 - KBs [9-42, 9-43](#)
 - packet files [13-7](#)
 - signature definition policies [7-2](#)
- copy iplog command [12-7](#)
- copy license-key command [4-60](#)
- copy packet-file command [13-6](#)
- copy signature-definition command [7-2](#)
- correcting time on the sensor [4-37, C-17](#)
- creating
 - anomaly detection policies [9-9](#)
 - Atomic IP Advanced signatures [7-57](#)

- banner logins [17-20](#)
- custom signatures [7-46](#)
- event action rules policies [8-9](#)
- event action variables [8-12](#)
- global parameters [6-12](#)
- Meta signatures [7-55](#)
- OS maps [8-30](#)
- Post-Block VACLs [14-26](#)
- Pre-Block VACLs [14-26](#)
- service HTTP signatures [7-51](#)
- signature definition policies [7-2](#)
- string TCP signatures [7-48](#)
- string TCP XL signatures [7-58, 7-62](#)
- user profiles [14-20](#)
- virtual sensors [6-6, 6-9](#)
- creating the service account [4-29, C-6](#)
- cryptographic account
 - Encryption Software Export Distribution Authorization from [20-2](#)
 - obtaining [20-2](#)
- CSA MC
 - configuring IPS interfaces [11-4](#)
 - host posture events [11-2, 11-4](#)
 - quarantined IP address events [11-2](#)
 - supported IPS interfaces [11-4](#)
- CtrlTransSource
 - described [A-4, A-11](#)
 - illustration [A-12](#)
- Ctrl-N [1-5](#)
- Ctrl-P [1-5](#)
- current-config command [16-20](#)
- current configuration back up [16-24, C-2](#)
- custom signatures
 - AIC MIME-type [7-33](#)
 - Atomic IP Advanced signature [7-57](#)
 - configuration sequence [7-46](#)
 - described [7-4](#)
 - Meta signature [7-55](#)
 - service HTTP example [7-51](#)

- String TCP [7-48](#)
- String TCP XL [7-58, 7-62](#)

D

- data nodes [B-68](#)
- data structures (examples) [A-8](#)
- DDoS
 - protocols [B-72](#)
 - Stacheldraht [B-72](#)
 - TFN [B-72](#)
- debug logging enable [C-44](#)
- default blocking time [14-13](#)
- default keywords [1-10](#)
- defaults
 - password [2-2](#)
 - username [2-2](#)
 - virtual sensor vs0 [6-2](#)
- default service anomaly-detection command [9-9](#)
- default service event-action-rules command [8-8](#)
- default service signature-definition command [7-2](#)
- defining authorized keys [4-49](#)
- defining signatures [7-1](#)
- deleting
 - anomaly detection policies [9-9](#)
 - denied attackers list [8-37, 17-27](#)
 - event action rules policies [8-9](#)
 - event action variables [8-12](#)
 - inline interface pairs [5-20](#)
 - inline VLAN pairs [5-26](#)
 - OS maps [8-31](#)
 - signature definition policies [7-2](#)
 - signature variables [7-10](#)
 - target value ratings [8-16](#)
 - VLAN groups [5-32](#)
- Denial of Service. See DoS.
- denied attackers add [8-36](#)
- deny actions (list) [8-5](#)
- deny attacker command [8-36](#)

- deny-packet-inline described [8-7](#)
 - detect mode (anomaly detection) [9-4](#)
 - device access issues [C-39](#)
 - diagnosing network connectivity [17-45](#)
 - disabling
 - anomaly detection [9-49, C-19](#)
 - blocking [14-10](#)
 - global correlation [10-14](#)
 - password recovery [17-8, C-14](#)
 - signatures [7-18](#)
 - SSHv1 fallback [4-13](#)
 - Telnet [4-5](#)
 - disaster recovery [C-6](#)
 - displaying
 - anomaly detection policies [9-9](#)
 - anomaly detection policy lists [17-29](#)
 - anomaly detection statistics [9-47](#)
 - contents of logical file [16-20](#)
 - current configuration [16-1](#)
 - current submode configuration [16-3](#)
 - event action rules policies [8-9](#)
 - event actions rules lists [17-29](#)
 - events [8-40, 17-23, C-100](#)
 - global correlation statistics [10-14](#)
 - health status [17-19, C-74](#)
 - inspection load [17-13](#)
 - interface statistics [5-40, 5-46](#)
 - interface traffic history [5-49, C-97](#)
 - IP log contents [12-5](#)
 - KB files [9-40](#)
 - KB thresholds [9-46](#)
 - LACP information [5-40](#)
 - live traffic [13-3](#)
 - OS IDs [8-32](#)
 - password recovery setting [17-9, C-14](#)
 - PEP information [17-47](#)
 - policy lists [17-29](#)
 - signature definition lists [17-29](#)
 - statistics [17-30, C-83](#)
 - submode settings [17-49](#)
 - system clock [4-38, 17-26](#)
 - tech support information [17-42, C-76](#)
 - threat profiles [7-6](#)
 - TLS trusted root certificates [4-57](#)
 - version [17-43, C-80](#)
 - Distributed Denial of Service. See [DDoS](#).
 - DNS servers
 - configuring [4-11](#)
 - DoS tools
 - Stacheldraht [B-72](#)
 - stick [B-7](#)
 - TFN [B-72](#)
 - downgrade command [21-12](#)
 - downgrading sensors [21-12](#)
 - downloading Cisco software [20-1](#)
 - duplicate IP addresses [C-27](#)
-
- ## E
- editing
 - anomaly detection policies [9-9](#)
 - event action rules policies [8-9](#)
 - event action variables [8-12](#)
 - signature definition policies [7-2](#)
 - signature variables [7-10](#)
 - target value ratings [8-16](#)
 - efficacy
 - described [10-5](#)
 - measurements [10-5](#)
 - enable-acl-logging command [14-14](#)
 - enable-detail-traps command [15-8](#)
 - enable-nvram-write command [14-15](#)
 - enabling
 - anomaly detection [9-8](#)
 - signatures [7-18](#)
 - SSHv1 fallback [4-13](#)
 - strict TLS server validation [4-54](#)
 - Telnet [4-5](#)

- enabling debug logging [C-44](#)
- Encryption Software Export Distribution Authorization form
 - cryptographic account [20-2](#)
 - described [20-2](#)
- engines
 - AIC [7-22, B-11](#)
 - AIC FTP [B-11](#)
 - AIC HTTP [B-11](#)
 - Atomic ARP [B-14](#)
 - Atomic IP [B-25](#)
 - Atomic IP Advanced [B-15](#)
 - Atomic IPv6 [B-28](#)
 - Fixed [B-29](#)
 - Fixed ICMP [B-29](#)
 - Fixed TCP [B-29](#)
 - Fixed UDP [B-29](#)
 - Flood [B-32](#)
 - Flood Host [B-32](#)
 - Flood Net [B-32](#)
 - Master [B-4](#)
 - Meta [7-53, B-33](#)
 - Multi String [B-35](#)
 - Normalizer [B-36](#)
 - Service [B-39](#)
 - Service DNS [B-39](#)
 - Service FTP [B-40](#)
 - Service Generic [B-41](#)
 - Service H225 [B-43](#)
 - Service HTTP [7-50, B-45](#)
 - Service IDENT [B-47](#)
 - Service MSRPC [B-48](#)
 - Service MSSQL [B-50](#)
 - Service NTP [B-51](#)
 - Service P2P [B-52](#)
 - Service RPC [B-52](#)
 - Service SMB Advanced [B-54](#)
 - Service SNMP [B-56](#)
 - Service SSH [B-57](#)
 - Service TNS [B-58](#)
 - State [B-59](#)
 - String [7-47, B-61](#)
 - String ICMP [7-47, B-61](#)
 - String TCP [7-47, B-61](#)
 - String UDP [7-47, B-61](#)
 - Sweep [B-67](#)
 - Sweep Other TCP [B-69](#)
 - Traffic Anomaly [B-70](#)
 - Traffic ICMP [B-72](#)
 - Trojan [B-73](#)
- erase ad-knowledge-base command [9-42](#)
- erase command [16-24](#)
- erase license-key command [4-62](#)
- erase packet-file command [13-7](#)
- erasing
 - current configuration [16-24](#)
 - KBs [9-42, 9-43](#)
 - packet files [13-7](#)
- error messages
 - described [D-1](#)
 - validation [D-6](#)
- evAlert [A-9](#)
- event-action command [7-20](#)
- event action filters
 - described [8-21](#)
 - using variables [8-21](#)
- event action overrides
 - described [8-17](#)
 - risk rating range [8-17](#)
- event action rules
 - described [8-2](#)
 - functions [8-2](#)
 - notes and caveats [8-1](#)
 - task list [8-8](#)
- event action rules lists display [17-29](#)
- event action rules policies
 - copying [8-9](#)
 - creating [8-9](#)

- deleting [8-9](#)
 - displaying [8-9](#)
 - editing [8-9](#)
 - event actions
 - risk ratings [8-15](#)
 - threat ratings [8-15](#)
 - event actions configure [7-21](#)
 - event-counter command [7-15](#)
 - event counter configure [7-16](#)
 - events
 - clearing [8-42, 17-25, C-102](#)
 - displaying [8-40, 17-23, C-100](#)
 - host posture [11-2](#)
 - quarantined IP address [11-2](#)
 - Event Store
 - clearing [8-42, 17-25, C-102](#)
 - clearing events [4-37, C-17](#)
 - data structures [A-8](#)
 - described [A-4](#)
 - examples [A-8](#)
 - no alerts [C-31](#)
 - responsibilities [A-7](#)
 - time stamp [4-37, C-17](#)
 - timestamp [A-7](#)
 - event types [C-98](#)
 - event variables
 - described [8-10](#)
 - example [8-11](#)
 - evError [A-9](#)
 - evLogTransaction [A-9](#)
 - evShunRqst [A-9](#)
 - evStatus [A-9](#)
 - examples
 - ASA failover configuration [18-21, 19-17, C-65, C-71](#)
 - default anomaly detection configuration [9-4](#)
 - KB histogram [9-37](#)
 - password [4-19](#)
 - password policy [4-33](#)
 - privilege [4-19](#)
 - SPAN configuration for IPv6 support [5-16](#)
 - System Configuration Dialog [3-4](#)
 - username [4-19](#)
 - external product interfaces
 - adding [11-5](#)
 - described [11-1](#)
 - issues [11-3, C-21](#)
 - notes and caveats [11-1](#)
 - troubleshooting [11-8, C-22](#)
 - external zone
 - configuring [9-29](#)
 - configuring other protocols [9-35](#)
 - configuring TCP [9-30](#)
 - configuring UDP [9-32](#)
 - described [9-28](#)
 - external-zone command [9-28](#)
-
- ## F
- failover
 - TCP support [5-36](#)
 - fallback
 - TCP support [5-36](#)
 - false positives described [7-4](#)
 - files
 - Cisco IPS (list) [20-1](#)
 - filtering
 - more command [16-17](#)
 - submode configuration [16-18](#)
 - filters command [8-22](#)
 - Fixed engine described [B-29](#)
 - Fixed ICMP engine parameters (table) [B-30](#)
 - Fixed TCP engine parameters (table) [B-30](#)
 - Fixed UDP engine parameters (table) [B-31](#)
 - Flood engine described [B-32](#)
 - Flood Host engine parameters (table) [B-32](#)
 - Flood Net engine parameters (table) [B-33](#)
 - fragment-reassembly command [7-36](#)
 - FTP servers and software updates [21-3](#)

FTP timeout

- configuring [4-8](#)
- described [4-8](#)

ftp-timeout command [4-8](#)

G

generating

- SSH server host key [4-51](#)
- TLS certificate [4-58](#)

generic commands [1-10](#)

global-block-timeout command [8-34, 14-13](#)

global correlation [10-1](#)

- described [3-2, 10-2](#)
- disabling about [10-13](#)

DNS server [10-7](#)

DNS servers [4-11](#)

error messages [A-29](#)

features [10-6](#)

goals [10-6](#)

health metrics [10-8](#)

health status [10-8](#)

HTTP proxy server [10-7](#)

IPS reloading messages [18-11, 19-12, C-67, C-73](#)

license [3-1, 3-6, 10-1, 10-7, 10-9](#)

no IPv6 support [8-1, 8-10, 8-11, 8-15, 8-21, 10-2, 10-7](#)

notes and caveats [10-1](#)

options [10-10, 10-11, 10-13](#)

Produce Alert [8-5](#)

proxy servers [4-11](#)

requirements [10-7](#)

risk rating [10-6](#)

troubleshooting [10-13, C-20](#)

update client (illustration) [10-9](#)

Global Correlation Update

- client described [A-28](#)
- server described [A-28](#)

global-deny-timeout command [8-34](#)

global-filters-status command [8-34](#)

global-metaevent-status command [8-34](#)

global-overrides-status command [8-34](#)

global parameters

- adding [6-12](#)
- creating [6-12](#)
- maximum open IP logs [6-12](#)
- options [6-12](#)

global-parameters command [6-12](#)

global-summarization command [8-35](#)

GRUB menu password recovery [17-3, C-8](#)

H

H.225.0 protocol [B-43](#)

H.323 protocol [B-43](#)

health-monitor command [10-8](#)

health statistics configuration [17-16](#)

health status

- display [17-19, C-74](#)
- global correlation [10-8](#)

help

- question mark [1-5](#)
- using [1-5](#)

host blocks configure [14-31](#)

host IP address

- changing [4-4](#)
- configuring [4-4](#)

host-ip command [4-4](#)

hostname

- changing [4-3](#)
- configuring [4-3](#)

host-name command [4-3](#)

host posture events

- CSA MC [11-4](#)
- described [11-2](#)

HTTP/HTTPS servers supported [21-3](#)

HTTP advanced decoding

- described [6-4](#)
- platform support [6-4](#)

- restrictions [6-4](#)
 - HTTP deobfuscation
 - ASCII normalization [7-50, B-45](#)
 - described [7-50, B-45](#)
 - hw-module module 1 recover configure command [19-13](#)
 - hw-module module slot_number password-reset command [17-6, 19-12, C-12](#)
 - hw-module module slot_number recover boot command [19-12](#)
 - hw-module module slot_number recover stop command [19-12](#)
 - hw-module module slot_number reload command [19-12](#)
 - hw-module module slot_number reset command [19-12](#)
 - hw-module module slot_number shutdown command [19-12](#)
-
- I**
- IDAPI
 - communications [A-4, A-32](#)
 - described [A-4](#)
 - functions [A-32](#)
 - illustration [A-32](#)
 - responsibilities [A-32](#)
 - IDCONF
 - described [A-33](#)
 - example [A-33](#)
 - RDEP2 [A-33](#)
 - XML [A-33](#)
 - IDIOM
 - defined [A-32](#)
 - messages [A-32](#)
 - IDM
 - Analysis Engine is busy [C-54](#)
 - certificates [4-52](#)
 - TLS [4-52](#)
 - will not load [C-53](#)
 - ignore command [9-10](#)
 - illegal zone
 - configuring [9-20](#)
 - configuring other protocols [9-26](#)
 - configuring TCP [9-22](#)
 - configuring UDP [9-24](#)
 - described [9-20](#)
 - protocols [9-20](#)
 - illegal-zone command [9-20](#)
 - IME time synchronization problems [C-56](#)
 - inactive mode (anomaly detection) [9-4](#)
 - initializing
 - appliances [3-9](#)
 - sensors [3-1, 3-5](#)
 - user roles [3-1, 3-2](#)
 - verifying [3-16](#)
 - verifying (ASA 5500-X IPS SSP) [18-3](#)
 - verifying (IPS SSP) [18-3, 19-3](#)
 - initializing the sensor (notes and caveats) [3-1](#)
 - inline interface pair mode
 - configuration restrictions [5-9](#)
 - described [5-16](#)
 - illustration [5-17](#)
 - inline interface pairs
 - configuring [5-17](#)
 - deleting [5-20](#)
 - inline-interfaces command [5-17](#)
 - inline mode
 - interface cards [5-4](#)
 - normalization [6-4](#)
 - pairing interfaces [5-4](#)
 - inline TCP session tracking modes described [6-3](#)
 - inline VLAN groups
 - configuring [5-29](#)
 - deleting [5-32](#)
 - inline VLAN pair mode
 - configuration restrictions [5-9](#)
 - described [5-21](#)
 - illustration [5-21](#)
 - supported sensors [5-21](#)
 - inline VLAN pairs
 - configuring [5-23](#)

- deleting [5-26](#)
- inspection load
 - description [17-12](#)
 - displaying [17-13](#)
- installer major version [20-5](#)
- installer minor version [20-5](#)
- installing
 - license key [4-61](#)
 - system image
 - ASA 5500-X IPS SSP [21-22](#)
 - ASA 5585-X IPS SSP [21-24](#)
 - IPS 4345 [21-17](#)
 - IPS 4360 [21-17](#)
 - IPS 4510 [21-20](#)
 - IPS 4520 [21-20](#)
- InterfaceApp described [A-4](#)
- interface configuration sequence [5-10](#)
- interface-notifications command [5-44](#)
- interfaces
 - alternate TCP reset [5-3](#)
 - command and control [5-3](#)
 - configuration restrictions [5-8](#)
 - described [5-2](#)
 - displaying live traffic [13-3](#)
 - port numbers [5-2](#)
 - sensing [5-3, 5-4](#)
 - slot numbers [5-2](#)
 - support (table) [5-6](#)
 - TCP reset [5-4](#)
- interface statistics displaying [5-40, 5-46](#)
- interface traffic history displaying [5-49, C-97](#)
- internal zone
 - configuring [9-12](#)
 - configuring other protocols [9-18](#)
 - configuring TCP [9-13](#)
 - configuring UDP [9-16](#)
 - described [9-12](#)
 - protocols [9-12](#)
- internal-zone command [9-12](#)
- introducing the CLI guide [1-1](#)
- IP fragmentation described [B-36](#)
- IP fragment reassembly
 - described [7-34](#)
 - parameters (table) [7-34](#)
 - signatures (table) [7-34](#)
- ip-log-bytes command [12-2](#)
- ip-log command [7-45](#)
- iplog command [12-3](#)
- IP log contents
 - displaying [12-5](#)
 - viewing [12-5](#)
- IP log files copying [12-7](#)
- IP logging
 - automatic [12-2](#)
 - configuring [12-2](#)
 - copying files [12-7](#)
 - described [7-45, 12-2](#)
 - manual [12-4](#)
 - notes and caveats [12-1](#)
- ip-log-packets command [12-2](#)
- ip logs
 - TCPDUMP [12-2](#)
 - Wireshark [12-2](#)
- iplog-status command [12-5](#)
- ip-log-time command [12-2](#)
- IP packet trace [17-48](#)
- IPS 4345
 - installing system image [21-17](#)
 - password recovery [17-3, 17-4, C-8, C-9](#)
 - reimaging [21-16](#)
- IPS 4360
 - installing system image [21-17](#)
 - password recovery [17-3, 17-4, C-8, C-9](#)
 - reimaging [21-16](#)
- IPS 4510
 - installing system image [21-20](#)
 - LACP grouping [5-33](#)
 - password recovery [17-3, 17-4, C-8, C-9](#)

- reimaging [21-20](#)
 - SwitchApp [A-29](#)
 - IPS 4520
 - installing system image [21-20](#)
 - LACP grouping [5-33](#)
 - password recovery [17-3, 17-4, C-8, C-9](#)
 - reimaging [21-20](#)
 - SwitchApp [A-29](#)
 - IPS appliances
 - Deny Connection Inline [8-7, 18-11, 19-11](#)
 - Deny Packet Inline [8-7, 18-11, 19-11](#)
 - Reset TCP Connection [8-7, 18-11, 19-11](#)
 - TCP reset packets [8-7, 18-11, 19-11](#)
 - IPS applications
 - summary [A-35](#)
 - table [A-35](#)
 - XML format [A-4](#)
 - IPS clock synchronization [4-37, C-16](#)
 - IPS data
 - types [A-8](#)
 - XML document [A-9](#)
 - IPS events
 - evAlert [A-9](#)
 - evError [A-9](#)
 - evLogTransaction [A-9](#)
 - evShunRqst [A-9](#)
 - evStatus [A-9](#)
 - list [A-9](#)
 - types [A-9](#)
 - IPS internal communications [A-32](#)
 - IPS port channel
 - illustration [5-34](#)
 - IPS software
 - application list [A-4](#)
 - available files [20-1](#)
 - configuring device parameters [A-5](#)
 - directory structure [A-34](#)
 - Linux OS [A-1](#)
 - obtaining [20-1](#)
 - retrieving data [A-5](#)
 - security features [A-5](#)
 - tuning signatures [A-5](#)
 - updating [A-5](#)
 - user interaction [A-5](#)
 - versioning scheme [20-3](#)
 - IPS software file names
 - major updates (illustration) [20-4](#)
 - minor updates (illustration) [20-4](#)
 - patch releases (illustration) [20-4](#)
 - service packs (illustration) [20-4](#)
 - IPS SSP
 - show module command [18-3, 19-3](#)
 - verifying initialization [18-3, 19-3](#)
 - virtual sensors
 - assigning to security context [19-7](#)
 - IPv4
 - address format [8-11](#)
 - event variables [8-11](#)
 - IPv6
 - address format [8-11](#)
 - described [B-28](#)
 - event variables [8-11](#)
 - SPAN ports [5-15](#)
 - switches [5-15](#)
 - ipv6-target-value command [8-15](#)
-
- ## K
- KB files
 - displaying [9-40](#)
 - KBs
 - comparing [9-44](#)
 - copying [9-42, 9-43](#)
 - described [9-3](#)
 - erasing [9-42, 9-43](#)
 - histogram [9-36](#)
 - initial baseline [9-3](#)
 - manually loading [9-41](#)

- manually saving [9-41](#)
- renaming [9-42, 9-43](#)
- scanner threshold [9-36](#)
- tree structure [9-36](#)

KB thresholds display [9-46](#)

keywords

- default [1-10](#)
- no [1-10](#)

Knowledge Base. See KB.

L

LACP

- configuring [5-37](#)
- displaying information [5-40](#)
- link states [5-36](#)
- restrictions [5-35](#)

learning accept mode

- anomaly detection [9-3](#)

learning-accept-mode command [9-38](#)

license key

- installing [4-61](#)
- obtaining [4-59](#)
- trial [4-59](#)
- uninstalling [4-62](#)
- viewing status of [4-59](#)

licensing

- described [4-59](#)
- IPS device serial number [4-59](#)

Licensing pane

- described [4-59](#)

limitations for concurrent CLI sessions [18-1, 19-1](#)

Link Aggregation Control Protocol. See LACP.

Link Aggregation Control Protocol Data Unit. See LACPDU.

list anomaly-detection-configurations command [9-9, 17-29](#)

list event-action-rules-configurations command [8-8, 17-29](#)

list of blocked hosts [14-34](#)

list signature-definition-configurations command [7-2, 17-29](#)

loading

- KBs [9-41](#)

log-all-block-events-and-errors command [14-16](#)

Logger

- described [A-4, A-19](#)
- functions [A-19](#)
- syslog messages [A-19](#)

logging in

- appliances [2-2](#)
- ASA 5500-X IPS SSP [2-4, 3-14](#)
- ASA 5585-X IPS SSP [2-5, 3-15](#)
- notes and caveats [2-1](#)
- sensors
 - SSH [2-6](#)
 - Telnet [2-6](#)
- service role [2-2](#)
- terminal servers [2-3, 21-16](#)
- user role [2-1](#)

login banners

- adding [4-9](#)

login-banner-text

- command [4-9](#)
- configuring [4-9](#)

LOKI

- described [B-72](#)
- protocol [B-72](#)

loose connections on sensors [C-23](#)

M

MainApp

- components [A-6](#)
- described [A-4, A-6](#)
- host statistics [A-6](#)
- responsibilities [A-6](#)
- show version command [A-6](#)

major updates described [20-3](#)

- managing
 - firewalls [14-27](#)
 - routers [14-23](#)
 - switches [14-26](#)
- manifests
 - client [A-28](#)
 - server [A-28](#)
- manual blocking [14-31, 14-33](#)
- manual block to bogus host [C-41](#)
- manually loading KBs [9-41](#)
- manually saving
 - KBs [9-41](#)
- master blocking sensor
 - described [14-28](#)
 - not set up properly [C-42](#)
 - verifying configuration [C-42](#)
- Master engine
 - alert frequency [B-7](#)
 - alert frequency parameters (table) [B-7](#)
 - described [B-4](#)
 - event actions [B-8](#)
 - general parameters (table) [B-4](#)
 - universal parameters [B-4](#)
- master engine parameters
 - obsoletes [B-6](#)
 - promiscuous delta [B-6](#)
 - vulnerable OSes [B-6](#)
- max-block-entries command [14-11](#)
- max-denied-attackers command [8-35](#)
- maximum open IP logs [6-12](#)
- max-interfaces command [14-17](#)
- merging configuration files [16-24, C-2](#)
- Meta engine
 - described [7-53, B-33](#)
 - parameters (table) [B-34](#)
 - Signature Event Action Processor [7-53, B-33](#)
- MIBs supported [15-11, C-18](#)
- minor updates described [20-3](#)
- modes
 - anomaly detection detect [9-4](#)
 - anomaly detection learning accept [9-3](#)
 - asymmetric [6-4](#)
 - bypass [5-42](#)
 - inactive (anomaly detection) [9-4](#)
 - inline interface pair [5-16](#)
 - inline TCP tracking [6-3](#)
 - inline VLAN pair [5-21](#)
 - Normalizer [6-4](#)
 - promiscuous [5-14](#)
 - VLAN groups [5-27](#)
- modifying
 - terminal properties [17-22](#)
- monitoring
 - viewer privileges [1-4](#)
- more command [16-20](#)
 - filtering [16-17](#)
- more current-config command [16-1](#)
- moving
 - OS maps [8-30](#)
- Multi String engine
 - described [B-35](#)
 - parameters (table) [B-35](#)
 - Regex [B-35](#)

N

- Neighborhood Discovery
 - options [B-29](#)
 - types [B-29](#)
- network blocks
 - configuring [14-32](#)
- network connectivity diagnosis [17-45](#)
- network participation
 - data gathered [10-4](#)
 - data use (table) [3-2, 10-3](#)
 - described [10-4](#)
 - health metrics [10-8](#)

- modes [10-5](#)
 - requirements [10-4](#)
 - SensorBase Network [10-5](#)
 - statistics [10-5](#)
 - network participation data
 - improving signature fidelity [10-5](#)
 - understanding sensor deployment [10-5](#)
 - never-block-hosts command [14-19](#)
 - never-block-networks command [14-19](#)
 - no iplog command [12-6](#)
 - no ipv6-target-value command [8-15](#)
 - normalization described [6-4](#)
 - Normalizer engine
 - described [B-36](#)
 - IPv6 fragments [B-37](#)
 - modify packets inline [6-3](#)
 - parameters (table) [B-37](#)
 - no service anomaly-detection command [9-9](#)
 - no service event-action-rules command [8-8](#)
 - no service signature-definition command [7-2](#)
 - no target-value command [8-15](#)
 - notes and caveats [7-1, 9-1, 10-1](#)
 - administrative tasks [17-2](#)
 - anomaly detection [9-1](#)
 - ASA 5500-X IPS SSP [18-1](#)
 - ASA 5585-X IPS SSP [19-1](#)
 - blocking [14-1](#)
 - capture packet files [13-1](#)
 - configuring interfaces [5-1](#)
 - event action rules [8-1](#)
 - external product interfaces [11-1](#)
 - initializing the sensor [3-1](#)
 - IP logging [12-1](#)
 - logging in [2-1](#)
 - setting up the sensor [4-1](#)
 - SNMP [15-1](#)
 - virtual sensors [6-1](#)
 - NotificationApp
 - alert information [A-9](#)
 - described [A-4](#)
 - functions [A-9](#)
 - SNMP gets [A-9](#)
 - SNMP traps [A-9](#)
 - statistics [A-11](#)
 - system health information [A-10](#)
 - no variables command [8-12](#)
 - NTP
 - authenticated [4-2, 4-36, 4-45, C-15](#)
 - configuring servers [4-44](#)
 - described [4-36, C-15](#)
 - incorrect configuration [C-16](#)
 - sensor time source [4-43, 4-45](#)
 - time synchronization [4-36, C-15](#)
 - unauthenticated [4-2, 4-36, 4-45, C-15](#)
-
- O**
- obsoletes field described [B-6](#)
 - obtaining
 - command history [17-47](#)
 - cryptographic account [20-2](#)
 - IPS software [20-1](#)
 - license key [4-59](#)
 - list of blocked hosts and connections [14-34](#)
 - used commands list [17-47](#)
 - operator role privileges [1-4](#)
 - options
 - global correlation [10-10, 10-11, 10-13](#)
 - os-identifications command [8-28](#)
 - OS IDs
 - clearing [8-32](#)
 - displaying [8-32](#)
 - OS information sources [8-27](#)
 - OS maps
 - creating [8-30](#)
 - deleting [8-31](#)
 - moving [8-30](#)
 - other actions (list) [8-6](#)

- other command [9-18, 9-26, 9-34](#)
 - output
 - clearing current line [1-6](#)
 - displaying [1-6](#)
 - overrides command [8-18](#)
- ## P
- P2P networks described [B-52](#)
 - packet capture command [13-4](#)
 - packet command restrictions
 - configuring [4-27](#)
 - packet display command [13-2](#)
 - packet files
 - viewing
 - TCPDUMP [13-7](#)
 - Wireshark [13-7](#)
 - partitions
 - application [A-4](#)
 - recovery [A-4](#)
 - passive OS fingerprinting
 - components [8-27](#)
 - configuring [8-28](#)
 - described [8-26](#)
 - enabled (default) [8-28](#)
 - password command [4-18, 4-30](#)
 - password policy
 - caution [4-32](#)
 - configuring [4-33](#)
 - password recovery
 - appliances [17-3, C-8](#)
 - ASA 5500-X IPS SSP [17-4, C-10](#)
 - ASA 5585-X IPS SSP [17-6, C-12](#)
 - CLI [17-8, C-14](#)
 - described [17-2, C-8](#)
 - disabling [17-8, C-14](#)
 - displaying setting [17-9, C-14](#)
 - GRUB menu [17-3, C-8](#)
 - IPS 4345 [17-3, 17-4, C-8, C-9](#)
 - IPS 4360 [17-3, 17-4, C-8, C-9](#)
 - IPS 4510 [17-3, 17-4, C-8, C-9](#)
 - IPS 4520 [17-3, 17-4, C-8, C-9](#)
 - platforms [17-2, C-8](#)
 - ROMMON [17-4, C-9](#)
 - troubleshooting [17-9, C-15](#)
 - verifying [17-9, C-14](#)
 - passwords [4-30](#)
 - changing [4-30](#)
 - configuring [4-30](#)
 - policy [4-32](#)
 - patch releases described [20-3](#)
 - peacetime learning (anomaly detection) [9-3](#)
 - Peer-to-Peer. See P2P.
 - PEP information
 - PID [17-47](#)
 - SN [17-47](#)
 - VID [17-47](#)
 - permit-packet-logging command [4-26](#)
 - physical connectivity issues [C-30](#)
 - physical interfaces
 - configuration restrictions [5-8](#)
 - configuring [5-12](#)
 - physical-interfaces command [5-11, 5-22, 5-28](#)
 - ping command [17-45](#)
 - platforms concurrent CLI sessions [18-1, 19-1](#)
 - policies
 - passwords [4-32](#)
 - policy lists
 - displaying [17-29](#)
 - port channel configuration
 - illustration [5-34](#)
 - switch [5-34](#)
 - Post-Block ACLs [14-22, 14-23](#)
 - Pre-Block ACLs [14-22, 14-23](#)
 - prerequisites for blocking [14-6](#)
 - privilege
 - changing [4-31](#)
 - configuring [4-31](#)

privilege command [4-18, 4-31](#)

privilege levels

administrator [1-3](#)

operators [1-3](#)

service [1-3](#)

viewers [1-3](#)

promiscuous delta

calculating risk rating [8-14](#)

described [7-12, 8-14, B-6](#)

promiscuous mode

atomic attacks [5-15](#)

configuring [5-15](#)

described [5-14](#)

illustration [5-15](#)

packet flow [5-14](#)

SPAN ports [5-15](#)

TCP reset interfaces [5-4](#)

VACL capture [5-15](#)

prompts

default input [1-5](#)

protocols

ARP [B-14](#)

CDP [5-45](#)

CIDEE [A-34](#)

DCE [B-48](#)

DDoS [B-72](#)

H.323 [B-43](#)

H225.0 [B-43](#)

HTTP [4-15](#)

ICMPv6 [B-15](#)

IDAPI [A-32](#)

IDCONF [A-33](#)

IDIOM [A-32](#)

IPv6 [B-28](#)

LOKI [B-72](#)

MSSQL [B-50](#)

Neighborhood Discovery [B-29](#)

Q.931 [B-43](#)

RPC [B-48](#)

SDEE [A-33](#)

proxy servers

configuring [4-11](#)

Q

Q.931 protocol

described [B-43](#)

SETUP messages [B-43](#)

quarantined IP address events described [11-2](#)

R

RADIUS

attempt limit [C-20](#)

multiple cisco av-pairs [4-22, 4-24](#)

RADIUS authentication

described [4-20](#)

service account [4-29](#)

shared secret [4-24, 4-25](#)

rate limiting

ACLs [14-5](#)

described [14-4](#)

routers [14-4](#)

service policies [14-5](#)

supported signatures [14-4](#)

raw expression syntax

described [B-64](#)

expert mode [B-64](#)

Raw Regex

described [7-59, 7-62, B-64](#)

expert mode [7-59, 7-62, B-64](#)

recall

help and tab completion [1-5](#)

using [1-5](#)

recover command [21-14](#)

recovering the application partition image [21-14](#)

recovery partition

described [A-4](#)

- upgrade [21-7](#)
 - Regex
 - described [1-8](#)
 - Multi String engine [B-35](#)
 - standardized [B-1](#)
 - Regular Expression. See also [Regex](#).
 - regular expression syntax
 - described [1-8](#)
 - raw Regex [7-59, 7-62, B-64](#)
 - signatures [B-9](#)
 - table [1-8](#)
 - reimaging
 - ASA 5500-X IPS SSP [21-22](#)
 - described [21-2](#)
 - IPS 4345 [21-16](#)
 - IPS 4360 [21-16](#)
 - IPS 4510 [21-20](#)
 - IPS 4520 [21-20](#)
 - sensors [21-2, 21-14](#)
 - removing
 - last applied
 - service pack [21-12](#)
 - signature update [21-12](#)
 - threat profiles [7-6](#)
 - users [4-19](#)
 - rename ad-knowledge-base command [9-42](#)
 - renaming
 - KBs [9-42, 9-43](#)
 - reputation
 - described [10-3](#)
 - illustration [10-4](#)
 - servers [10-3](#)
 - reset command [17-46](#)
 - reset not occurring for a signature [C-49](#)
 - resetting
 - passwords
 - ASDM [17-6, 17-8, C-11, C-13](#)
 - hw-module command [17-6, C-12](#)
 - sw-module command [17-5, C-10](#)
 - resetting appliances [17-46](#)
 - resetting the password
 - ASA 5500-X IPS SSP [17-5, C-10](#)
 - ASA 5585-X IPS SSP [17-7, C-12](#)
 - restoring the current configuration [16-23, C-5](#)
 - retiring signatures [7-18](#)
 - risk rating
 - Alarm Channel [10-6](#)
 - calculating [8-13](#)
 - described [8-27](#)
 - global correlation [10-6](#)
 - reputation score [10-6](#)
 - ROMMON
 - ASA 5585-X IPS SSP [21-26](#)
 - described [21-15](#)
 - IPS 4345 [17-4, 21-17, C-9](#)
 - IPS 4360 [17-4, 21-17, C-9](#)
 - IPS 4510 [17-4, 21-20, C-9](#)
 - IPS 4520 [17-4, 21-20, C-9](#)
 - password recovery [17-4, C-9](#)
 - remote sensors [21-15](#)
 - serial console port [21-15](#)
 - TFTP [21-15](#)
 - round-trip time. See [RTT](#).
 - RPC portmapper [B-52](#)
 - RSA authentication
 - authorized keys [4-49](#)
 - RTT
 - described [21-15](#)
 - TFTP limitation [21-15](#)
- ## S
-
- saving
 - KBs [9-41](#)
 - scheduling automatic upgrades [21-9](#)
 - SDEE
 - described [A-33](#)
 - HTTP [A-33](#)

- protocol [A-33](#)
- server requests [A-34](#)
- searching
 - submode configuration [16-18](#)
- security
 - account locking [4-34](#)
 - information on Cisco Security Intelligence Operations [20-7](#)
 - SSH [4-47](#)
- security policies described [7-2, 8-2, 9-2](#)
- sensing interface
 - ASA 5500-X IPS SSP [18-4](#)
- sensing interfaces
 - Analysis Engine [5-4](#)
 - described [5-4](#)
 - interface cards [5-4](#)
 - modes [5-4](#)
- sensing interfaces described (ASA 5585-X IPS SSP) [19-4](#)
- SensorApp
 - Alarm Channel [A-24](#)
 - Analysis Engine [A-24](#)
 - described [A-4](#)
 - event action filtering [A-25](#)
 - inline packet processing [A-24](#)
 - IP normalization [A-24](#)
 - packet flow [A-25](#)
 - processors [A-23](#)
 - responsibilities [A-23](#)
 - risk rating [A-25](#)
 - Signature Event Action Processor [A-23](#)
 - TCP normalization [A-24](#)
- SensorBase Network
 - described [3-2, 10-2](#)
 - network participation [10-5](#)
 - participation [3-2, 10-2](#)
 - servers [3-2, 10-2](#)
- sensor databases
 - clearing [17-10](#)
- Sensor Key pane
 - described [4-50](#)
- sensors
 - access problems [C-24](#)
 - application partition image [21-14](#)
 - asymmetric traffic and disabling anomaly detection [9-48, C-18](#)
 - command and control interfaces (list) [5-3](#)
 - configuration sequence [1-2](#)
 - configuring to use NTP [4-45](#)
 - corrupted SensorApp configuration [C-34](#)
 - disaster recovery [C-6](#)
 - downgrading [21-12](#)
 - incorrect NTP configuration [C-16](#)
 - initializing [3-1, 3-5](#)
 - interface support [5-6](#)
 - IP address conflicts [C-27](#)
 - logging in
 - SSH [2-6](#)
 - Telnet [2-6](#)
 - loose connections [C-23](#)
 - managing
 - firewalls [14-27](#)
 - routers [14-23](#)
 - switches [14-26](#)
 - misconfigured access lists [C-26](#)
 - no alerts [C-31, C-55](#)
 - not seeing packets [C-33](#)
 - NTP time source [4-45](#)
 - NTP time synchronization [4-36, C-15](#)
 - partitions [A-4](#)
 - physical connectivity [C-30](#)
 - preventive maintenance [C-2](#)
 - reimaging [21-2](#)
 - sensing process not running [C-28](#)
 - setup command [3-1, 3-5, 3-9](#)
 - time sources [4-36, C-15](#)
 - troubleshooting software upgrades [C-52](#)
 - upgrading [21-5](#)

- using NTP time source [4-43](#)
- server manifest described [A-28](#)
- service account
 - accessing [4-28, C-5](#)
 - cautions [4-2, 4-28, C-5](#)
 - creating [4-29, C-6](#)
 - described [4-28, A-31, C-5](#)
 - RADIUS authentication [4-29](#)
 - TAC [A-31](#)
 - troubleshooting [A-31](#)
- service anomaly-detection command [9-8](#)
- Service DNS engine
 - described [B-39](#)
 - parameters (table) [B-39](#)
- Service engine
 - described [B-39](#)
 - Layer 5 traffic [B-39](#)
- service event-action-rules command [8-8](#)
- Service FTP engine
 - described [B-40](#)
 - parameters (table) [B-41](#)
 - PASV port spoof [B-40](#)
- Service Generic engine
 - described [B-41](#)
 - no custom signatures [B-41](#)
 - parameters (table) [B-42](#)
- Service H225 engine
 - ASN.1PER validation [B-43](#)
 - described [B-43](#)
 - features [B-43](#)
 - parameters (table) [B-44](#)
 - TPKT validation [B-43](#)
- Service HTTP engine
 - described [7-50, B-45](#)
 - parameters (table) [B-46](#)
- service HTTP engine
 - signature [7-51](#)
- Service IDENT engine
 - described [B-47](#)
- parameters (table) [B-48](#)
- Service MSRPC engine
 - DCS/RPC protocol [B-48](#)
 - described [B-48](#)
 - parameters (table) [B-49](#)
- Service MSSQL engine
 - described [B-50](#)
 - MSSQL protocol [B-50](#)
 - parameters (table) [B-51](#)
- Service NTP engine
 - described [B-51](#)
 - parameters (table) [B-51](#)
- Service P2P engine described [B-52](#)
- service packs described [20-3](#)
- service role
 - described [1-4, 2-2, A-30](#)
 - privileges [1-4](#)
- Service RPC engine
 - described [B-52](#)
 - parameters (table) [B-52](#)
 - RPC portmapper [B-52](#)
- service signature-definition command [7-2](#)
- Service SMB Advanced engine
 - described [B-54](#)
 - parameters (table) [B-54](#)
- Service SNMP engine
 - described [B-56](#)
 - parameters (table) [B-56](#)
- Service SSH engine
 - described [B-57](#)
 - parameters (table) [B-57](#)
- Service TNS engine
 - described [B-58](#)
 - parameters (table) [B-58](#)
- session command
 - ASA 5500-X IPS SSP [2-4, 3-14](#)
 - ASA 5585-X IPS SSP [2-5, 3-15](#)
- sessioning in
 - ASA 5500-X IPS SSP [2-4, 3-14](#)

- ASA 5585-X IPS SSP [2-5, 3-15](#)
- setting
 - system clock [4-39, 17-27](#)
- setting up
 - notes and caveats [4-1](#)
 - terminal servers [2-3, 21-16](#)
- setup
 - automatic [3-3](#)
 - command [3-1, 3-5, 3-9](#)
 - simplified mode [3-3](#)
- setup command
 - user roles [3-1, 3-2](#)
- shared secret
 - described [4-24, 4-25](#)
 - RADIUS authentication [4-24, 4-25](#)
- show ad-knowledge-base diff command [9-44, 9-45](#)
- show ad-knowledge-base files command [9-40, 9-41](#)
- show clock command [4-37, 17-25](#)
- show configuration command [16-1](#)
- show context command [18-7, 19-8](#)
- show events command [8-39, 17-22, C-98, C-99](#)
- show health command [10-9, 17-19, C-74](#)
- show history command [17-47](#)
- showing user information [4-32](#)
- show inspection-load command [17-12](#)
- show interfaces command [5-46, C-94](#)
- show interfaces-history command [5-48, C-96](#)
- show inventory command [17-47](#)
- show lacp command [5-38](#)
- show module [18-3](#)
- show module 1 details command [19-13, C-57, C-67](#)
- show module command [18-3, 19-3](#)
- show os-identification command [8-32](#)
- show settings command [7-8, 16-3, 16-18, 17-9, 17-49, C-14](#)
- show statistics anomaly-detection command [9-47](#)
- show statistics command [14-34, 17-30, C-82, C-83](#)
- show statistics denied-attackers command [8-37, 17-27](#)
- show statistics virtual-sensor command [17-30, C-23, C-83](#)
- show tech-support command [17-42, C-75](#)
- show tls trusted-root-certificates command [4-57](#)
- show users command [4-32](#)
- show version command [7-8, 17-43, C-79](#)
- sig-fidelity-rating command [7-17, 7-19](#)
- signature definition lists
 - displaying [17-29](#)
- signature definition policies
 - copying [7-2](#)
 - creating [7-2](#)
 - deleting [7-2](#)
 - editing [7-2](#)
- signature engines
 - AIC [7-22, B-11](#)
 - Atomic [B-14](#)
 - Atomic ARP [B-14](#)
 - Atomic IP [B-25](#)
 - Atomic IP Advanced [B-15](#)
 - Atomic IPv6 [B-28](#)
 - described [B-1](#)
 - Fixed [B-29](#)
 - Flood [B-32](#)
 - Flood Host [B-32](#)
 - Flood Net [B-33](#)
 - list [B-2](#)
 - Master [B-4](#)
 - Meta [7-53, B-33](#)
 - Multi String [B-35](#)
 - Normalizer [B-36](#)
 - Regex
 - patterns [B-10](#)
 - syntax [B-9](#)
 - Service [B-39](#)
 - Service DNS [B-39](#)
 - Service FTP [B-40](#)
 - Service Generic [B-41](#)
 - Service H225 [B-43](#)
 - Service HTTP [7-50, B-45](#)
 - Service IDENT [B-47](#)
 - Service MSRPC [B-48](#)

- Service MSSQL [B-50](#)
- Service NTP [B-51](#)
- Service P2P [B-52](#)
- Service RPC [B-52](#)
- Service SMB Advanced [B-54](#)
- Service SNMP [B-56](#)
- Service SSH engine [B-57](#)
- Service TNS [B-58](#)
- State [B-59](#)
- String [7-47](#), [B-61](#)
- Sweep [B-67](#)
- Sweep Other TCP [B-69](#)
- Traffic Anomaly [B-70](#)
- Traffic ICMP [B-72](#)
- Trojan [B-73](#)
- Signature Event Action Filter
 - described [8-3](#), [A-26](#)
 - parameters [8-3](#), [A-26](#)
- Signature Event Action Handler described [8-4](#), [A-26](#)
- Signature Event Action Override described [8-3](#), [A-26](#)
- Signature Event Action Processor
 - Alarm Channel [8-3](#), [A-26](#)
 - components [8-3](#), [A-26](#)
 - described [8-3](#), [A-23](#), [A-26](#)
- signature fidelity rating
 - calculating risk rating [8-14](#)
 - configuring [7-17](#), [7-19](#)
 - described [8-14](#)
- signatures
 - custom [7-4](#)
 - default [7-4](#)
 - described [7-3](#)
 - false positives [7-4](#)
 - general parameters [7-11](#)
 - rate limits [14-4](#)
 - service HTTP [7-51](#)
 - string TCP [7-48](#)
 - string TCP XL [7-58](#), [7-62](#)
 - subsignatures [7-4](#)
- TCP reset [C-49](#)
- threat profile tuned [7-8](#)
- tuned [7-4](#)
- signature threat profiles
 - platform support [7-4](#)
- signature update
 - files [20-4](#)
 - IPS reloading messages [18-11](#), [19-12](#), [C-67](#), [C-73](#)
- signature variables
 - adding [7-10](#)
 - deleting [7-10](#)
 - described [7-9](#)
 - editing [7-10](#)
- SNMP
 - configuring
 - agent parameters [15-3](#)
 - traps [15-9](#)
 - described [15-1](#)
 - general parameters [15-2](#)
 - Get [15-1](#)
 - GetNext [15-1](#)
 - notes and caveats [15-1](#)
 - Set [15-1](#)
 - supported MIBs [15-11](#), [C-18](#)
 - Trap [15-1](#)
- snmp-agent-port command [15-3](#)
- snmp-agent-protocol command [15-3](#)
- SNMP traps
 - described [15-2](#)
- SNMPv3
 - configuring
 - users [15-5](#)
 - configuring users [15-4](#)
- software architecture
 - ARC (illustration) [A-13](#)
 - IDAPI (illustration) [A-32](#)
- software downloads Cisco.com [20-1](#)
- software file names
 - recovery (illustration) [20-5](#)

- signature/virus updates (illustration) [20-4](#)
- system image (illustration) [20-5](#)
- software release examples
 - platform identifiers [20-6](#)
 - platform-independent [20-5](#)
- software updates
 - supported FTP servers [21-3](#)
 - supported HTTP/HTTPS servers [21-3](#)
- SPAN port issues [C-30](#)
- specifying
 - worm timeout [9-11, 9-38](#)
- SSH
 - adding hosts [4-48](#)
 - described [4-47](#)
 - security [4-47](#)
- ssh authorized-key command [4-49](#)
- ssh generate-key command [4-50](#)
- ssh host-key command [4-47](#)
- SSH known hosts list
 - adding hosts [4-47](#)
- SSH Server
 - private keys [A-21](#)
 - public keys [A-21](#)
- SSH server host key
 - generating [4-51](#)
- SSHv1 fallback
 - disabling [4-13](#)
 - enabling [4-13](#)
- sshv1-fallback
 - command [4-13](#)
 - configuring [4-13](#)
- standards
 - CIDEE [A-34](#)
 - IDCONF [A-33](#)
 - IDIOM [A-32](#)
 - SDEE [A-33](#)
- State engine
 - Cisco Login [B-59](#)
 - described [B-59](#)
 - LPR Format String [B-59](#)
 - parameters (table) [B-60](#)
 - SMTP [B-59](#)
- statistic display [17-30, C-83](#)
- status command [7-18](#)
- stopping
 - IP logging [12-6](#)
- stream-reassembly command [7-44](#)
- strict-tls-server-validation command [4-54](#)
- String engine described [7-47, B-61](#)
- String ICMP engine parameters (table) [B-62](#)
- String TCP engine
 - parameters [7-47](#)
 - parameters (table) [B-62](#)
 - signature (example) [7-48](#)
- String TCP XL signature
 - example [7-58, 7-62](#)
- String UDP engine parameters (table) [B-63](#)
- String XL engine
 - description [B-64](#)
 - hardware support [6-12, B-3, B-64](#)
 - parameters (table) [B-64](#)
 - unsupported parameters [B-67](#)
- subinterface 0 described [5-27](#)
- subinterface-type command [5-22, 5-28](#)
- submode configuration
 - filtering output [16-18](#)
 - searching output [16-18](#)
- submode settings display [17-49](#)
- subsignatures described [7-4](#)
- summarization
 - described [8-33](#)
 - fire-all [8-34](#)
 - fire-once [8-34](#)
 - global-summarization [8-34](#)
 - Meta engine [8-34](#)
 - summary [8-34](#)

summertime

- configuring
 - non-recurring [4-41](#)
 - recurring [4-39](#)

summertime-option non-recurring command [4-41](#)

summertime-option recurring command [4-39](#)

supported

- FTP servers [21-3](#)
- HTTP/HTTPS servers [21-3](#)
- IPS interfaces for CSA MC [11-4](#)

supported sensors

- signature threat profiles [7-4](#)

Sweep engine [B-68](#)

- described [B-67](#)
- parameters (table) [B-68](#)

Sweep Other TCP engine

- described [B-69](#)
- parameters (table) [B-70](#)

SwitchApp

- described [A-29](#)

switches

- TCP reset interfaces [5-5](#)

sw-module module 1 recover configure command [18-12](#)

sw-module module slot_number password-reset command [17-5, 18-12, C-10](#)

sw-module module slot_number reload command [18-12](#)

sw-module module slot_number reset command [18-12](#)

sw-module module slot_number shutdown command [18-12](#)

syntax

- case sensitivity [1-6](#)

system architecture

- directory structure [A-34](#)
- supported platforms [A-1](#)

system clock

- displaying [4-38, 17-26](#)
- setting [4-39, 17-27](#)

system components IDAPI [A-32](#)

System Configuration Dialog

- described [3-3](#)
- example [3-4](#)

system design (illustration) [A-2, A-3](#)

system images

- installing
 - ASA 5500-X IPS SSP [21-22](#)
 - IPS 4345 [21-17](#)
 - IPS 4360 [21-17](#)
 - IPS 4510 [21-20](#)
 - IPS 4520 [21-20](#)

T

tab completion

- using [1-5](#)

TAC

- PEP information [17-47](#)
- service account [4-28, A-31, C-5](#)
- show tech-support command [17-42, C-75](#)
- troubleshooting [A-31](#)

target-value command [8-15](#)

- IPv4 [8-15](#)
- IPv6 [8-15](#)

target value rating

- calculating risk rating [8-14](#)
- described [8-14, 8-15](#)

tasks

- configuring the sensor [1-2](#)

tcp command [9-13, 9-21, 9-29](#)

TCPDUMP

- copy packet-file command [13-6](#)
- expression syntax [13-2](#)
- ip logs [12-2](#)
- packet capture command [13-5](#)
- packet display command [13-2](#)

TCP fragmentation described [B-37](#)

TCP reset interfaces

- conditions [5-5](#)

- described [5-4](#)
- list [5-5](#)
- promiscuous mode [5-4](#)
- switches [5-5](#)
- TCP resets
 - not occurring [C-49](#)
- TCP stream reassembly
 - described [7-37](#)
 - parameters (table) [7-38, 7-43](#)
 - signatures (table) [7-38, 7-43](#)
- tech support information display [17-42, C-76](#)
- Telnet
 - disabling [4-5](#)
 - enabling [4-5](#)
- telnet-option
 - command [4-5](#)
 - configuring [4-5](#)
- terminal
 - command [17-21](#)
 - modifying length [17-22](#)
- terminal server setup [2-3, 21-16](#)
- terminating
 - CLI sessions [17-21](#)
- TFN2K
 - described [B-72](#)
 - Trojans [B-73](#)
- TFTP servers
 - recommended
 - UNIX [21-15](#)
 - Windows [21-15](#)
 - RTT [21-15](#)
- threat-profile command [7-5](#)
- threat profiles
 - applying [7-6](#)
 - displaying [7-6](#)
 - removing [7-6](#)
- threat rating
 - described [8-15](#)
 - risk rating [8-15](#)
- time
 - correction on the sensor [4-37, C-17](#)
 - sensors [4-36, C-15](#)
 - synchronizing IPS clocks [4-37, C-16](#)
- time sources
 - appliances [4-36, C-15](#)
 - ASA 5500 AIP SSC-5 [4-36](#)
 - ASA 5500 AIP SSM [4-36](#)
 - ASA 5500-X IPS SSP [4-36, C-15](#)
 - ASA 5585-X IPS SSP [4-36, C-15](#)
- time zone settings
 - configuring [4-43](#)
- time-zone-settings command [4-43](#)
- TLS
 - handshaking [4-52](#)
 - IDM [4-52](#)
 - web server [4-52](#)
- TLS certificates
 - generating [4-58](#)
- tls generate-key command [4-58](#)
- tls trusted-host command [4-53, 4-55](#)
- TLS trusted root certificates
 - displaying [4-57](#)
- trace command [17-48](#)
- tracing IP packet route [17-48](#)
- Traffic Anomaly engine
 - described [B-70](#)
 - protocols [B-70](#)
 - signatures [B-70](#)
- traffic flow notifications
 - configuring [5-44](#)
 - described [5-44](#)
- Traffic ICMP engine
 - DDoS [B-72](#)
 - described [B-72](#)
 - LOKI [B-72](#)
 - parameters (table) [B-73](#)
 - TFN2K [B-72](#)
- trap-community-name [15-8](#)

- trap-destinations command [15-8](#)
- trial license key [4-59](#)
- Tribe Flood Network. See TFN.
- Tribe Flood Network 2000. See TFN2K.
- Trojan engine
 - BO2K [B-73](#)
 - described [B-73](#)
 - TFN2K [B-73](#)
- Trojans
 - BO [B-73](#)
 - BO2K [B-73](#)
 - LOKI [B-72](#)
 - TFN2K [B-73](#)
- troubleshooting [C-1](#)
 - Analysis Engine busy [C-54](#)
 - applying software updates [C-51](#)
 - ARC
 - blocking not occurring for signature [C-41](#)
 - device access issues [C-39](#)
 - enabling SSH [C-41](#)
 - inactive state [C-37](#)
 - misconfigured master blocking sensor [C-42](#)
 - verifying device interfaces [C-40](#)
 - ASA 5500-X IPS SSP
 - commands [C-57](#)
 - failover scenarios [18-20, C-64](#)
 - ASA 5585-X IPS SSP
 - commands [19-13, C-67](#)
 - failover scenarios [19-16, C-70](#)
 - traffic flow stopped [19-16, C-72](#)
 - automatic updates [C-51](#)
 - cannot access sensor [C-24](#)
 - cidDump [C-102](#)
 - cidLog messages to syslog [C-48](#)
 - communication [C-24](#)
 - corrupted SensorApp configuration [C-34](#)
 - debug logger zone names (table) [C-48](#)
 - debug logging [C-44](#)
 - disaster recovery [C-6](#)
 - duplicate sensor IP addresses [C-27](#)
 - enabling debug logging [C-44](#)
 - external product interfaces [11-8, C-22](#)
 - gathering information [C-73](#)
 - global correlation [10-13, C-20](#)
 - IDM
 - cannot access sensor [C-54](#)
 - will not load [C-53](#)
 - IME time synchronization [C-56](#)
 - IPS clock time drift [4-37, C-16](#)
 - manual block to bogus host [C-41](#)
 - misconfigured access list [C-26](#)
 - no alerts [C-31, C-55](#)
 - NTP [C-49](#)
 - password recovery [17-9, C-15](#)
 - physical connectivity issues [C-30](#)
 - preventive maintenance [C-2](#)
 - RADIUS
 - attempt limit [C-20](#)
 - reset not occurring for a signature [C-49](#)
 - sensing process not running [C-28](#)
 - sensor events [C-98](#)
 - sensor loose connections [C-23](#)
 - sensor not seeing packets [C-33](#)
 - sensor software upgrade [C-52](#)
 - service account [4-28, C-5](#)
 - show events command [C-98](#)
 - show interfaces command [C-94](#)
 - show statistics command [C-82](#)
 - show tech-support command [C-74, C-75, C-76](#)
 - show version command [C-79](#)
 - software upgrades [C-51](#)
 - SPAN
 - port issue [C-30](#)
 - verifying Analysis Engine is running [C-20](#)
 - verifying ARC status [C-36](#)
- trusted hosts add [4-53, 4-55](#)
- tuned signatures described [7-4](#)

U

- udp command [9-15, 9-24, 9-32](#)
- unassigned VLAN groups described [5-27](#)
- unauthenticated NTP [4-2, 4-36, 4-45, C-15](#)
- uninstalling
 - license key [4-62](#)
- unlocking accounts [4-35](#)
- unlock user username command [4-35](#)
- updating the sensor immediately [21-11](#)
- upgrade command [21-4, 21-6](#)
- upgrade notes and caveats
 - upgrading IPS software [21-1](#)
- upgrading
 - application partition [21-14](#)
 - recovery partition [21-7](#)
 - sensors [21-5](#)
- upgrading IPS software
 - upgrade notes and caveats [21-1](#)
- URLs for Cisco Security Intelligence Operations [20-7](#)
- username command [4-18](#)
- user-profile command [14-20](#)
- user profiles [14-20](#)
- user roles
 - administrator [1-3](#)
 - authentication [4-20](#)
 - operator [1-3](#)
 - service [1-3](#)
 - viewer [1-3](#)
- users
 - adding [4-18, 4-19](#)
 - removing [4-18, 4-19](#)
- using
 - debug logging [C-44](#)
 - TCP reset interfaces [5-5](#)

V

- VACLs
 - described [14-3](#)
 - Post-Block [14-26](#)
 - Pre-Block [14-26](#)
- validation error messages described [D-6](#)
- variables command [7-10, 8-12](#)
 - IPv4 [8-12](#)
 - IPv6 [8-12](#)
- verifying
 - password recovery [17-9, C-14](#)
 - sensor initialization [3-16](#)
 - sensor setup [3-16](#)
- version display [17-43, C-80](#)
- viewer role privileges [1-4](#)
- viewing
 - IP log contents [12-5](#)
 - license key status [4-59](#)
 - threat profile tuned signatures [7-8](#)
 - user information [4-32](#)
- virtualization
 - advantages [6-2, C-17](#)
 - restrictions [6-3, C-17](#)
 - supported sensors [6-3, C-18](#)
 - traffic capture requirements [6-3, C-17](#)
- virtual-sensor name command [6-5, 18-4, 19-5](#)
- virtual sensors
 - adding [6-6, 6-9](#)
 - adding (ASA 5500-X IPS SSP) [18-5](#)
 - adding (ASA 5585-X IPS SSP) [19-5](#)
 - ASA 5500-X IPS SSP [18-7](#)
 - ASA 5585-X IPS SSP [19-8](#)
 - assigning interfaces [6-5](#)
 - assigning policies [6-5](#)
 - creating [6-6, 6-9](#)
 - creating (ASA 5500-X IPS SSP) [18-5](#)
 - creating (ASA 5585-X IPS SSP) [19-5](#)
 - default virtual sensor [6-2](#)

described [6-2](#)
 displaying KB files [9-40](#)
 notes and caveats [6-1](#)
 options [6-5, 18-4, 19-5](#)

VLAN groups

802.1q encapsulation [5-27](#)
 configuration restrictions [5-10](#)
 deploying [5-27](#)
 switches [5-27](#)

VLAN groups mode

described [5-27](#)

vulnerable OSes field described [B-6](#)

Sasser [9-2](#)
 scanners [9-3](#)
 Slammer [9-2](#)
 SQL Slammer [9-2](#)

worm-timeout command [9-10](#)

worm timeout specify [9-11, 9-38](#)

Z

zones

external [9-4](#)
 illegal [9-4](#)
 internal [9-4](#)

W

watch list rating

calculating risk rating [8-14](#)
 described [8-14](#)

web server

described [A-4, A-22](#)
 HTTP 1.0 and 1.1 support [A-22](#)
 HTTP protocol [4-15](#)
 port (default) [4-1, 4-15](#)
 private keys [A-21](#)
 public keys [A-21](#)
 SDEE support [A-22](#)
 TLS [4-52](#)

web server settings

changing [4-16](#)
 configuring [4-15](#)

Wireshark

copy packet-file command [13-6](#)
 ip logs [12-2](#)

worms

Blaster [9-2](#)
 Code Red [9-2](#)
 histograms [9-37](#)
 Nimbda [9-2](#)
 protocols [9-3](#)