Cisco Intrusion Prevention System Sensor CLI Configuration Guide for IPS 6.1

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-15172-01

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 Sensor CLI Configuration Guide for IPS 6.1
© 2008-2012 Cisco Systems, Inc. All rights reserved.

CONTENTS

Preface

Published: April 21, 2008, OL-15172-01

Revised: October 26, 2012

Contents

This document describes how to configure the sensor using the Cisco IPS 6.1 CLI. It contains the following topics:

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.

Conventions

This guide uses the following conventions:

- Filenames, directory names, and arguments for which you supply values are in italics.
- The symbol ^ represents the key labeled Ctrl (control). To enter a control key; for example, ^z, hold down the Ctrl key while you press the Z key.
- Command names, buttons, and keywords in text are shown in boldface.
- Command statements are shown in boldface italic screen font.

- Variables in command syntax descriptions are shown in italics. Command options in square brackets [ ] can be optionally entered, and parameters separated by a vertical bar (|) require you to enter one parameter, but not the other(s).
- Examples depict screen displays and the command line in screen font.
- Information you need to enter in examples is shown in boldface screen font.
- Variables for which you must supply a value are shown in italic screen font.
- Selecting a menu item (or screen) is indicated by the following convention: Select screen > screen 2 > screen 3.

Note Means reader take note. Notes contain helpful suggestions or references to material not covered in the manual.

Caution Means reader be careful. In this situation, you might do something that could result in equipment damage or loss of data.

Warning **Identifies information that you must heed to prevent damaging yourself, the state of software, or equipment. Warnings identify definite security breaches that will result if the information presented is not followed carefully.**

Related Documentation

For more information on Cisco IPS 6.1, refer to the following documentation found at this URL:

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

- *Documentation Roadmap for Cisco Intrusion Prevention System 6.1*
- *Release Notes for Cisco Intrusion Prevention System 6.1*
- *Cisco Intrusion Prevention System Device Manager Configuration Guide for IPS 6.1*
- *Cisco Intrusion Prevention System Manager Express Configuration Guide for IPS 6.1*
- *Cisco Intrusion Prevention System Command Reference 6.1*
- *Installing Cisco Intrusion Prevention System Appliances and Modules for IPS 6.1*
- *Installing and Removing Interface Cards in Cisco IPS-4260 and IPS 4270-20*
- *Regulatory Compliance and Safety Information for the Cisco Intrusion Detection and Prevention System 4200 Series Appliance Sensor*

Obtaining Documentation and Submitting a Service Request

For information on obtaining documentation, submitting a service request, and gathering additional information, see the monthly *What's New in Cisco Product Documentation*, which also lists all new and revised Cisco technical documentation, at:

http://www.cisco.com/en/US/docs/general/whatsnew/whatsnew.html

Subscribe to the *What's New in Cisco Product Documentation* as a Really Simple Syndication (RSS) feed and set content to be delivered directly to your desktop using a reader application. The RSS feeds are a free service and Cisco currently supports RSS Version 2.0.

CHAPTER 1

Introducing the CLI Configuration Guide

This chapter describes how to use the IPS CLI, and contains the following sections:

Purpose of the CLI Configuration Guide

This guide is a task-based configuration guide for the Cisco IPS 6.1 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, such as the AIM IPS, AIP SSM, IDSM2, or NME IPS.

For an alphabetical list of all IPS commands, refer to *Command Reference for Cisco Intrusion Prevention System 6.1*. For information on locating all IPS 6.1 documents on Cisco.com, refer to *Documentation Roadmap for Cisco Intrusion Prevention System 6.1*.

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 *Documentation Roadmap for Cisco Intrusion Prevention System 6.1*.

Sensor Configuration Sequence

Perform the following tasks to configure the sensor:

1. Log in to the sensor.
2. Initialize the sensor.

   Run the **setup** command to initialize the sensor.
3. Verify the sensor initialization.

4. Create the 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. However, you can use the service account to create a new password if the administrator password is lost. Analyze your situation to decide if you want a service account existing on the system.

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 anomaly detection.

    You can run anomaly detection using the default values or you can tailor it to suit your network needs.
12. Set up any external product interfaces.

    CSA MC is the only external product supported by Cisco IPS 6.1.
13. Configure IP Logging.
14. Configure blocking.
15. Configure SNMP if you are going to use it.
16. Perform miscellaneous tasks to keep your sensor running smoothly.
17. Upgrade the IPS software with new signature updates and service packs.
18. Reimage the application partition and the maintenance 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 3-27.
- For the procedure for obtaining and installing the license key, see Installing the License Key, page 4-49.
- 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 4-14.
- 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 7, "Configuring Event Action Rules."
- For the procedures for configuring signatures for intrusion prevention, see Chapter 8, "Defining Signatures."
- 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 10, "Configuring External Product Interfaces."
- For the procedures for configuring IP logging, see Chapter 11, "Configuring IP Logging."
- For the procedures for configuring blocking on your sensor, see Chapter 13, "Configuring Attack Response Controller for Blocking and Rate Limiting."
- For the procedures for configuring SNMP on your sensor, see Chapter 14, "Configuring SNMP."
- For the administrative procedures, see Chapter 16, "Administrative Tasks for the Sensor."
- For more information on how to obtain Cisco IPS software, see Chapter 21, "Obtaining Software."
- For the procedures for working with images, see Chapter 22, "Upgrading, Downgrading, and Installing System Images."
- For procedures specific to the modules, see the following chapters:
  - Chapter 17, "Configuring the AIM IPS"
  - Chapter 18, "Configuring the AIP SSM"
  - Chapter 19, "Configuring the IDSM2"
  - Chapter 20, "Configuring the NME IPS"

User Roles

Note All IPS platforms allow ten concurrent log in sessions.

The CLI for Cisco IPS 6.1 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.

- **Administrators**—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.

  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 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.

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
  ```

  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**.

  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
  ```

  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 1-7.

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 `---More---` 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 `---More---` 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.
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 redisplays the system prompt and command line
Ctrl-T	Transposes the character to the left of the cursor with the character located at the cursor.

Table 1-1 Command Line Editing (continued)

Keys	Description
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, *not* 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

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 "a|b" 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.
[]	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 Cisco IPS 6.1.

- **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, host, interface, logger, network-access, notification, signature-definition, ssh-known-hosts, trusted-certificates, and web-server.

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

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. The AIM IPS and NME IPS support only the predefined instances. All other sensors support the creation of additional instances.

- **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.

CHAPTER 2

Logging In to the Sensor

Note All IPS platforms allow ten concurrent log in sessions.

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

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.

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.

For More Information

For the procedure for creating the service account, see Creating the Service Account, page 4-14.

Logging In to the Appliance

You can log in to the appliance from a console port.

Note You must initialize the appliance (run the **setup** command) from the console. After networking is configured, SSH and Telnet are available.

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/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-4240#
```

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 3-3.

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 180 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 AIM IPS

This section describes how to use the **session** command to log in to the AIM IPS, and contains the following topics:

- The AIM IPS and the session Command, page 2-4
- Sessioning In to the AIM IPS, page 2-5

The AIM IPS and the session Command

Because the AIM IPS does not have an external console port, console access to the AIM IPS is enabled when you issue the **service-module ids-sensor** *slot/port* **session** command on the router, or when you initiate a Telnet connection into the router with the slot number corresponding to the AIM IPS port number. The lack of an external console port means that the initial bootup configuration is possible only through the router.

When you issue the **service-module ids-sensor** *slot/port* **session** command, you create a console session with the AIM IPS, in which you can issue any IPS configuration commands. After completing work in the session and exiting the IPS CLI, you are returned to the Cisco IOS CLI.

The **session** command starts a reverse Telnet connection using the IP address of the IDS-Sensor interface. The IDS-Sensor interface is an interface between the AIM IPS and the router. You must assign an IP address to the IDS-Sensor interface before invoking the **session** command. Assigning a routable IP address can make the IDS-Sensor interface itself vulnerable to attacks, because the AIM IPS is visible on the network through that routable IP address, meaning you can communicate with the AIM IPS outside the router. To counter this vulnerability, assign an unnumbered IP address to the IDS-Sensor interface. Then the AIM IPS IP address is only used locally between the router and the AIM IPS, and is isolated for the purposes of sessioning in to the AIM IPS.

Note Before you install your application software or reimage the module, opening a session brings up the bootloader. After you install the software, opening a session brings up the application.

Caution If you session to the module and perform large console transfers, character traffic may be lost unless the host console interface speed is set to 115200/bps or higher. Use the **show running config** command to check that the speed is set to 115200/bps.

For More Information

For the procedure for configuring an unnumbered IP address interface for the AIM IPS, see Using an Unnumbered IP Address Interface, page 17-5.

Sessioning In to the AIM IPS

Because the AIM IPS does not have an external console port, console access to the AIM IPS is enabled when you issue the **service-module ids-sensor** *slot/port* **session** command on the router, or when you initiate a Telnet connection into the router with the slot number corresponding to the AIM IPS port number. The lack of an external console port means that the initial bootup configuration is possible only through the router.

When you issue the **service-module ids-sensor** *slot/port* **session** command, you create a console session with the AIM IPS, in which you can issue any IPS configuration commands. After completing work in the session and exiting the IPS CLI, you are returned to the Cisco IOS CLI.

The **session** command starts a reverse Telnet connection using the IP address of the IDS-Sensor interface. The IDS-Sensor interface is an interface between the AIM IPS and the router. You must assign an IP address to the IDS-Sensor interface before invoking the **session** command. Assigning a routable IP address can make the IDS-Sensor interface itself vulnerable to attacks, because the AIM IPS is visible on the network through that routable IP address, meaning you can communicate with the AIM IPS outside the router. To counter this vulnerability, assign an unnumbered IP address to the IDS-Sensor interface. Then the AIM IPS IP address is only used locally between the router and the AIM IPS, and is isolated for the purposes of sessioning in to the AIM IPS.

Note Before you install your application software or reimage the module, opening a session brings up the bootloader. After you install the software, opening a session brings up the application.

Caution If you session to the module and perform large console transfers, character traffic may be lost unless the host console interface speed is set to 115200/bps or higher. Use the **show running config** command to check that the speed is set to 115200/bps.

For More Information

For the procedure for using the **setup** command to initialize the AIM IPS, see Advanced Setup for the AIM IPS, page 3-12.

Logging In to AIP SSM

You log in to the AIP SSM from the ASA 5500 series adaptive security appliance.

Note You must initialize the AIP SSM (run the **setup** command) from the ASA 5500 series adaptive security appliance. After networking is configured, SSH and Telnet are available.

To session in to the AIP SSM from the ASA 5500 series adaptive security appliance, follow these steps:

Step 1 Log in to the ASA 5500 series adaptive security appliance.

Note If the ASA 5500 series 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 AIP SSM.

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

You have 60 seconds to log in before the session times out.

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 AIP SSM. 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.
aip-ssm#
```

Step 4 To escape from a session and return to the ASA 5500 series 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 AIP SSM, see Advanced Setup for the AIP SSM, page 3-15.

Logging In to the IDSM2

You log in to the IDSM2 from the switch. You must initialize the IDSM2 (run the **setup** command) from the switch. After networking is configured, SSH and Telnet are available.

To session in to the IDSM2, follow these steps:

Step 1 Session to the IDSM2 from the switch:

- For Catalyst Software:

  console> (enable) **session** *slot_number*

- For Cisco IOS software:

  router# **session** *slot_number* **processor 1**

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 IDSM2.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.
idsm-2#
```

For More Information

For the procedure for using the **setup** command to initialize the IDSM2, see Advanced Setup for the IDSM2, page 3-20.

Logging In to the NME IPS

This section describes how to use the **session** command to log in to the NME IPS, and contains the following topics:

The NME IPS and the session Command

Because the NME IPS does not have an external console port, console access to the NME IPS is enabled when you issue the **service-module ids-sensor** *slot/port* **session** command on the router, or when you initiate a Telnet connection into the router with the slot number corresponding to the NME IPS port number. The lack of an external console port means that the initial bootup configuration is possible only through the router.

When you issue the **service-module ids-sensor** *slot/port* **session** command, you create a console session with the NME IPS, in which you can issue any IPS configuration commands. After completing work in the session and exiting the IPS CLI, you are returned to the Cisco IOS CLI.

The **session** command starts a reverse Telnet connection using the IP address of the IDS-Sensor interface. The IDS-Sensor interface is an interface between the NME IPS and the router. You must assign an IP address to the IDS-Sensor interface before invoking the **session** command. Assigning a routable IP address can make the IDS-Sensor interface itself vulnerable to attacks, because the NME IPS is visible on the network through that routable IP address, meaning you can communicate with the NME IPS outside the router. To counter this vulnerability, assign an unnumbered IP address to the IDS-Sensor interface. Then the NME IPS IP address is only used locally between the router and the NME IPS, and is isolated for the purposes of sessioning in to the NME IPS.

Note Before you install your application software or reimage the module, opening a session brings up the bootloader. After you install the software, opening a session brings up the application.

Caution If you session to the module and perform large console transfers, character traffic may be lost unless the host console interface speed is set to 115200/bps or higher. Use the **show running config** command to check that the speed is set to 115200/bps.

For More Information

For the procedure for configuring the monitoring interface for the NME IPS, see Setting Up Interfaces on the NME IPS and the Router, page 20-4.

Sessioning In to the NME IPS

Note You must initialize the NME IPS (run the **setup** command) from the router. After networking is configured, SSH and Telnet are available.

Use the **service-module ids-sensor** *slot/port* **session** command to establish a session from the NME IPS to the module. Press **Ctrl-Shift-6**, then **x**, to return a session prompt to a router prompt, that is, to go from the NME IPS prompt back to the router prompt. Press **Enter** on a blank line to go back to the session prompt, which is also the router prompt. You should only suspend a session to the router if you will be returning to the session after executing router commands. If you do not plan on returning to the NME IPS session, you should close the session rather than suspend it.

When you close a session, you are logged completely out of the NME IPS CLI and a new session connection requires a username and password to log in. A suspended session leaves you logged in to the CLI. When you connect with the **session** command, you can go back to the same CLI without having to provide your username and password.

Note Telnet clients vary. In some cases, you may have to press **Ctrl-6 + x**. The control character is specified as ^^, Ctrl-^, or ASCII value 30 (hex 1E).

Caution If you use the **disconnect** command to leave the session, the session remains running. The open session can be exploited by someone wanting to take advantage of a connection that is still in place.

To open and close sessions to the NME IPS, follow these steps:

Step 1 Log in to the router.

Step 2 Check the status of the NME IPS to make sure it is running.

```
router# service-module ids-sensor 1/0 status
Service Module is Cisco IDS-Sensor1/0
Service Module supports session via TTY line 130
Service Module is in Steady state
Service Module heartbeat-reset is disabled
Getting status from the Service Module, please wait..

Cisco Systems Intrusion Prevention System Network Module
  Software version:  6.1(1)E2
  Model:             NME-IPS
  Memory:            443508 KB
  Mgmt IP addr:      10.89.148.195
  Mgmt web ports:    443
  Mgmt TLS enabled:  true


router#
```

Step 3 Open a session from the router to the NME IPS.

```
router# service-module ids-sensor 1/0 session
Trying 10.89.148.195, 2322 ... Open
```

Step 4 Exit, or suspend and close the module session:

- `sensor# exit`

Note If you are in submodes of the IPS CLI, you must exit all submodes. Enter **`exit`** until the sensor login prompt appears.

Caution Failing to close a session properly makes it possible for others to exploit a connection that is still in place. Remember to enter **exit** at the `router#` prompt to close the Cisco IOS session completely.

- To suspend and close the session to the NME IPS, press **Ctrl-Shift** and press **6**. Release all keys, and then press **x**.

Note When you are finished with a session, you need to return to the router to establish the association between a session (the IPS application) and the router interfaces you want to monitor.

Step 5 Disconnect from the router.

```
router# disconnect
```

Step 6 Press **Enter** to confirm the disconnection.

```
router# Closing connection to 10.89.148.196 [confirm] <Enter>
```

For More Information

For the procedure for using the **setup** command to initialize the NME IPS, see Advanced Setup for the NME IPS, page 3-24.

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, 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 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.
sensor#
```

CHAPTER 3

Initializing the Sensor

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

Understanding Initialization

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

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. With the **setup** command, you configure basic sensor settings, including the hostname, IP interfaces, access control lists, and time settings. You can continue using Advanced Setup in the CLI to enable Telnet, configure the Web server, and assign and enable virtual sensors and interfaces, or you can use the Startup Wizard in IDM or IME.

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: Thu Mar  6 21:19:51 2008

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:
  No entries
Permit:
Permit:
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]: yes
    NTP Server IP Address[]:
    Use NTP Authentication?[no]: yes
      NTP Key ID[]: 1
      NTP Key Value[]: 8675309
```

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/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, *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.

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

a. 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.

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). 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 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** 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.

The following completed configuration appears:

```
The following configuration was entered.
service host
network-settings
host-ip 10.89.143.126/24,10.89.143.254
host-name sensor126
telnet-option disabled
access-list 10.0.0.0/8
ftp-timeout 300
no login-banner-text
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 10.89.143.92 key-id 1
exit
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 8 Enter **2** to save the configuration (or **3** to continue with advanced setup using the CLI, IDM, or IME).

```
Enter your selection[2]: 2
Configuration Saved.
```

Step 9 Enter **yes** to reboot the sensor.

Step 10 After reboot, log in to the sensor, and display the self-signed X.509 certificate (needed by TLS):

```
sensor# show tls fingerprint
MD5: C4:BC:F2:92:C2:E2:4D:EB:92:0F:E4:86:53:6A:C6:01
SHA1: 64:9B:AC:DE:21:62:0C:D3:57:2E:9B:E5:3D:04:8F:A7:FD:CD:6F:27
```

Step 11 Write down the certificate fingerprints.

You need the fingerprints to check the authenticity of the certificate when using HTTPS to connect to this appliance with a web browser.

Step 12 Apply the most recent service pack and signature update.

You are now ready to configure your sensor for intrusion prevention.

For More Information

For the procedure for obtaining the most recent IPS software, see Obtaining Cisco IPS Software, page 21-1.

Advanced Setup

This section describes how to continue with Advanced Setup in the CLI for the various Cisco IPS platforms. It contains the following sections:

Advanced Setup for the Appliance

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

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.

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.

Step 3 Enter `3` to access advanced setup.

Step 4 Specify the Telnet server status. The default is disabled.

Step 5 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 If you change the web server port, you must specify the port in the URL address of your browser when you connect to the IDM in the format https://*appliance_ip_address:port* (for example, https://10.1.9.201:1040).

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

Step 6 Enter `yes` to modify the interface and virtual sensor configuration.

The current interface configuration appears:

```
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 7 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.

The following options appear:

```
  [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 8 Enter **2** to add inline VLAN pairs.

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

The list of available interfaces is displayed:

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

Step 9 Enter **1** to add an inline VLAN pair to GigabitEthernet0/0, for example:

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

Step 10 Enter a subinterface number and description:

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

Step 11 Enter numbers for VLAN 1 and 2:

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

Step 12 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, GigabitEthernet0/1, for inline VLAN pair.

Step 13 Press **Enter** to return to the top-level interface editing menu.

The following options appear:

```
  [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 14 Enter **4** to add an inline interface pair.

The following options appear:

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

Step 15 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 16 Press **Enter** to return to the top-level interface editing menu.

The following options appear:

```
  [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 17 Press **Enter** to return to the top-level editing menu.

The following options appear:

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

Step 18 Enter **2** to edit the virtual sensor configuration.

The following options appear:

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

Step 19 Enter **2** to modify the virtual sensor configuration, vs0.

The following options appear:

```
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 20 Enter **3** to add inline VLAN pair GigabitEthernet0/0:1.

Step 21 Enter **4** to add inline interface pair NewPair.

Step 22 Press **Enter** to return to the top-level virtual sensor menu.

The following options appear:

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

The following options appear:

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

Step 24 Enter **yes** 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.

The following appears:

```
   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 25 Enter **yes** to disable automatic threat prevention on all virtual sensors.

Step 26 Press **Enter** to exit the interface and virtual sensor configuration.

The following completed configuration appears:

```
The following configuration was entered.
service host
network-settings
host-ip 10.1.9.201/24,10.1.9.1
host-name sensor
telnet-option disabled
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 27 Enter **2** to save the configuration.

```
Enter your selection[2]: 2
Configuration Saved.
```

Step 28 Reboot the appliance:

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

Step 29 Enter **yes** to continue the reboot.

Step 30 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 21-1.

Advanced Setup for the AIM IPS

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

Step 1 Session in to the AIM IPS using an account with administrator privileges:

```
router# service-module ids-sensor 0/0 session
Trying 10.1.9.1, 2322 ... Open


sensor login: cisco
Password: ********
```

Step 2 Enter the **setup** command.

The System Configuration Dialog is displayed.

Step 3 Enter **3** to access advanced setup.

Step 4 Specify the Telnet server status.

You can disable or enable Telnet services. The default is disabled.

Step 5 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 6 Enter **yes** to modify the interface and virtual sensor configuration.

You may receive a warning that Analysis Engine is initializing and you cannot modify the virtual sensor configuration at this time. Press the space bar to receive the following menu:

```
[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.

Enter your selection[2]:
```

If you receive the warning that Analysis Engine is initializing, enter **2** to save your configuration thus far and exit setup. You can then reenter setup and press **Enter** until you are back to the interface and virtual sensor menu.

Step 7 Enter **2** to modify the virtual sensor configuration.

```
Modify interface/virtual sensor configuration?[no]: yes
Current interface configuration
 Command control: Management0/0
 Unassigned:
  Monitored:
   GigabitEthernet0/1

 Virtual Sensor: vs0
  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 **2** to edit the virtual sensor vs0 configuration.

The following appears:

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

No Interfaces to remove.

Unassigned:
  Monitored:
   [1] GigabitEthernet0/1
Add Interface:
```

Step 9 Enter **1** to add GigabitEthernet0/1 to virtual sensor vs0.

```
Add Interface: 1

 Virtual Sensor: vs0
  Anomaly Detection: ad0
```

```
  Event Action Rules: rules0
  Signature Definitions: sig0
  Monitored:
   GigabitEthernet0/1

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

Step 10 Press **Enter** to exit the interface and virtual sensor configuration menu.

The following option appears:

```
Modify default threat prevention settings?[no]:
```

Step 11 Enter **yes** 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.

The following appears:

```
   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 12 Enter **yes** to disable automatic threat prevention on all virtual sensors.

The following completed configuration appears:

```
The following configuration was entered.

service host
network-settings
host-ip 10.1.9.201/24,10.1.9.1
host-name aim-ips
telnet-option disabled
access-list 10.0.0.0/8
access-list 64.0.0.0/8
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
service analysis-engine
virtual-sensor vs0
physical-interface GigabitEthernet0/1
exit
exit
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 13 Enter **2** to save the configuration.

```
Enter your selection[2]: 2
Configuration Saved.
```

Step 14 Reboot the AIM IPS.

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

Step 15 Enter **yes** to continue the reboot.

Step 16 Apply the most recent service pack and signature update.

You are now ready to configure your AIM IPS for intrusion prevention.

For More Information

For the procedure for obtaining the most recent IPS software, see Obtaining Cisco IPS Software, page 21-1.

Advanced Setup for the AIP SSM

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

Step 1 Session in to the AIP SSM using an account with administrator privileges:

```
asa# session 1
```

Step 2 Enter the **setup** command.

The System Configuration Dialog is displayed.

Step 3 Enter **3** to access advanced setup.

Step 4 Specify the Telnet server status.

You can disable or enable Telnet services. The default is disabled.

Step 5 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 6 Enter **yes** to modify the interface and virtual sensor configuration.

The current interface configuration appears:

```
Current interface configuration
```

```
 Command control: GigabitEthernet0/0
 Unassigned:
  Monitored:
   GigabitEthernet0/1

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

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

Step 7 Enter **1** to edit the interface configuration.

Note You do not need to configure interfaces on the AIP SSM. You should ignore the Modify interface default-vlan setting. The separation of traffic across virtual sensors is configured differently for the AIP SSM than for other sensors.

The following option appears:

```
  [1] Modify interface default-vlan.
Option:
```

Step 8 Press **Enter** to return to the top-level interface and virtual sensor configuration menu.

The following options appear:

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

Step 9 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 10 Enter **2** to modify the virtual sensor vs0 configuration.

The following appears:

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

No Interfaces to remove.

 Unassigned:
  Monitored:
   [1] GigabitEthernet0/1
Add Interface:
```

Step 11 Enter **1** to add GigabitEthernet0/1 to virtual sensor vs0.

Note With ASA 7.2 and earlier, one virtual sensor is supported. The virtual sensor to which GigabitEthernet0/1 is assigned is used for monitoring packets coming from the adaptive security appliance. We recommend that you assign GigabitEthernet0/1 to vs0, although you can assign it to another virtual sensor.

Note With ASA 7.2.3 and later running IPS 6.0 or later, multiple virtual sensors are supported. The ASA 7.2.3 can direct packets to specific virtual sensors or can send packets to be monitored by a default virtual sensor. The default virtual sensor is the virtual sensor to which you assign GigabitEthernet0/1. We recommend that you assign GigabitEthernet0/1 to vs0, although you can assign it to another virtual sensor.

Step 12 Press **Enter** to return to the main virtual sensor menu.

Step 13 Enter **3** to create a virtual sensor.

The following option appears:

```
Name[]:
```

Step 14 Enter a name and description for your virtual sensor.

```
Name[]: newVs
Description[Created via setup by user cisco]: New Sensor
Anomaly Detection Configuration
  [1] ad0
  [2] Create a new anomaly detection configuration
Option[2]:
```

Step 15 Enter **1** to use the existing anomaly-detection configuration, ad0.

The following options appear:

```
Signature Definition Configuration
  [1] sig0
  [2] Create a new signature definition configuration
Option[2]:
```

Step 16 Enter **2** to create a signature-definition configuration file.

Step 17 Enter the signature-definition configuration name, **newSig**.

The following options appear:

```
Event Action Rules Configuration
  [1] rules0
  [2] Create a new event action rules configuration
Option[2]:
```

Step 18 Enter **1** to use the existing event-action-rules configuration, rules0.

Note If GigabitEthernet0/1 has not been assigned to vs0, you are prompted to assign it to the new virtual sensor.

Note With ASA 7.2 and earlier, one virtual sensor is supported. The virtual sensor to which GigabitEthernet0/1 is assigned is used for monitoring packets coming from the adaptive security appliance. We recommend that you assign GigabitEthernet0/1 to vs0, although you can assign it to another virtual sensor.

Note With ASA 7.2.3 and later with IPS 6.0, multiple virtual sensors are supported. The ASA 7.2.3 can direct packets to specific virtual sensors or can send packets to be monitored by a default virtual sensor. The default virtual sensor is the virtual sensor to which you assign GigabitEthernet0/1. We recommend that you assign GigabitEthernet0/1 to vs0, although you can assign it to another virtual sensor.

The following options appear:

```
Virtual Sensor: newVs
  Anomaly Detection: ad0
  Event Action Rules: rules0
  Signature Definitions: newSig
  Monitored:
   GigabitEthernet0/1

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

Step 19 Press **Enter** to exit the interface and virtual sensor configuration menu.

The following option appears:

```
Modify default threat prevention settings?[no]:
```

Step 20 Enter **yes** 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.

The following appears:

```
   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 21 Enter **yes** to disable automatic threat prevention on all virtual sensors.

The following completed configuration appears:

```
The following configuration was entered.

service host
network-settings
host-ip 10.1.9.201/24,10.1.9.1
host-name aip-ssm
telnet-option disabled
access-list 10.0.0.0/8
```

```
access-list 64.0.0.0/8
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 analysis-engine
virtual-sensor newVs
description New Sensor
signature-definition newSig
event-action-rules rules0
anomaly-detection
anomaly-detection-name ad0
exit
physical-interfaces GigabitEthernet0/1
exit
exit
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 22 Enter **2** to save the configuration.

```
Enter your selection[2]: 2
Configuration Saved.
```

Step 23 Reboot the AIP SSM.

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

Step 24 Enter **yes** to continue the reboot.

Step 25 Apply the most recent service pack and signature update.

You are now ready to configure your AIP SSM for intrusion prevention.

For More Information

For the procedure for obtaining the most recent IPS software, see Obtaining Cisco IPS Software, page 21-1.

Advanced Setup for the IDSM2

To continue with advanced setup for the IDSM2, follow these steps:

Step 1 Session in to the IDSM2 using an account with administrator privileges:

- For Catalyst software:

```
console> enable
console> (enable) session module_number
```

- For Cisco IOS software:

```
router# session slot slot_number processor 1
```

Step 2 Enter the **setup** command.

The System Configuration Dialog is displayed.

Step 3 Enter **3** to access advanced setup.

Step 4 Specify the Telnet server status.

You can disable or enable Telnet services. The default is disabled.

Step 5 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 6 Enter **yes** to modify the interface and virtual sensor configuration.

The current interface configuration appears:

```
Current interface configuration
 Command control: GigabitEthernet0/2
 Unassigned:
  Promiscuous:
   GigabitEthernet0/7
   GigabitEthernet0/8

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

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

Step 7 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.

Note The IDSM2 does not support the Add/Modify Inline Interface Pair Vlan Groups option. When running in inline interface pair mode the two IDSM2 data ports are configured as access ports or a trunk port carrying only the native VLAN. The packets do not have 802.1q headers and cannot be separated by VLAN. To monitor multiple VLANs inline, use inline VLAN pairs.

The following options appear:

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

Step 8 Enter **3** to add promiscuous VLAN groups.

The list of available interfaces is displayed:

```
Available Interfaces
  [1] GigabitEthernet0/7
  [2] GigabitEthernet0/8
Option:
```

Step 9 Enter **2** to add VLAN groups to GigabitEthernet0/8.

```
Promiscuous Vlan Groups for GigabitEthernet0/8
   None
Subinterface Number:
```

a. Enter **10** to add subinterface 10.

```
Subinterface Number: 10
Description[Created via setup by user asmith]:
Select vlans:
  [1] All unassigned vlans.
  [2] Enter vlans range.
Option:
```

b. Enter **1** to assign all unassigned VLANs to subinterface 10.

```
Subinterface Number:
```

c. Enter **9** to add subinterface 9.

```
Subinterface Number: 9
Description[Created via setup by user asmith]:
Vlans[]:
```

d. Enter **1-100** to assign VLANs 1-100 to subinterface 9.

Note This removes VLANs 1-100 from the unassigned VLANs contained in subinterface 10.

e. Repeat Steps c and d until you have added all VLAN groups.

f. Press **Enter** at a blank subinterface line to return to list of interfaces available for VLAN groups.

The following options appear:

```
  [1] GigabitEthernet0/7
  [2] GigabitEthernet0/8
Option:
```

Step 10 Press **Enter** to return to the top-level interface configuration menu.

The following options appear:

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

Step 11 Press **Enter** to return to the top-level menu.

The following options appear:

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

Step 12 Enter **2** to edit the virtual sensor configuration.

The following option appears:

```
  [1] Remove vs
  [2] Modify "vs0"
  [3] Create new vs
Option:
```

Step 13 Enter **2** to modify the virtual sensor vs0 configuration.

The following options appear:

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

No Interfaces to remove.

Unassigned:
  Promiscuous:
   [1] GigabitEthernet0/7
```

Step 14 Enter **2** to add VLAN group GigabitEthernet0/8:10 to the virtual sensor vs0.

```
Promiscuous Vlan Groups:
   [2] GigabitEthernet0/8:10  (Vlans: unassigned)
   [3] GigabitEthernet0/8:9  (Vlans: 1-100)
Add Interface:
```

Step 15 Press **Enter** to return to the top-level virtual sensor configuration menu.

The following options appear:

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

```
   GigabitEthernet0/8:10  (Vlans: unassigned)
   GigabitEthernet0/8:9  (Vlans: 1-100)

  [1] Remove vs
  [2] Modify "vs0"
  [3] Create new vs
Option:
```

Step 16 Press **Enter** to return to the top-level interface and virtual sensor configuration menu.

The following options appear:

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

Step 17 Press **Enter** to exit the interface and virtual sensor configuration menu.

Step 18 Enter **yes** 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.

The following appears:

```
   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 19 Enter **yes** to disable automatic threat prevention on all virtual sensors.

The following completed configuration appears:

```
The following configuration was entered.
service host
network-settings
host-ip 10.1.9.201/24,10.1.9.1
host-name idsm-2
telnet-option disabled
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 GigabitEthernet0/8
admin-state enabled
subinterface-type vlan-group
subinterface 9
description Created via setup by user asmith
vlans range 1-100
exit
subinterface 10
description Created via setup by user asmith
```

```
vlans unassigned
exit
exit
exit
exit
service analysis-engine
virtual-sensor vs0
description Created via setup by user cisco
signature-definition sig0
event-action-rules rules0
anomaly-detection
anomaly-detection-name ad0
operational-mode inactive
exit
physical-interface GigabitEthernet0/8 subinterface-number 9
physical-interface GigabitEthernet0/8 subinterface-number 10
service event-action-rules rules0
overrides deny-packet-inline
override-item-status Disabled
risk-rating-range 90-100
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 20 Enter **2** to save the configuration.

```
Enter your selection[2]: 2
Configuration Saved.
```

Step 21 Reboot the IDSM2:

```
idsm-2# reset
Warning: Executing this command will stop all applications and reboot the node.
Continue with reset? []:
```

Step 22 Enter **yes** to continue the reboot.

Step 23 Apply the most recent service pack and signature update.

You are now ready to configure your IDSM2 for intrusion prevention.

For More Information

For the procedure for obtaining the most recent IPS software, see Obtaining Cisco IPS Software, page 21-1.

Advanced Setup for the NME IPS

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

Step 1 Session in to the NME IPS using an account with administrator privileges:

```
router# service-module ids-sensor 1/0 session
Trying 10.1.9.1, 2322 ... Open


sensor login: cisco
Password: ********
```

Step 2 Enter the **setup** command.

The System Configuration Dialog is displayed.

Step 3 Enter **3** to access advanced setup.

Step 4 Specify the Telnet server status.

You can disable or enable Telnet services. The default is disabled.

Step 5 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 6 Enter **yes** to modify the interface and virtual sensor configuration.

You may receive a warning that Analysis Engine is initializing and you cannot modify the virtual sensor configuration at this time. Press the space bar to receive the following menu:

```
[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.

Enter your selection[2]:
```

If you receive the warning that Analysis Engine is initializing, enter **2** to save your configuration thus far and exit setup. You can then reenter setup and press **Enter** until you are back to the interface and virtual sensor menu.

Step 7 Enter **2** to modify the virtual sensor configuration.

```
Modify interface/virtual sensor configuration?[no]: yes
Current interface configuration
 Command control: Management0/1
 Unassigned:
  Monitored:
   GigabitEthernet0/1

 Virtual Sensor: vs0
  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 **2** to edit the virtual sensor vs0 configuration.

The following appears:

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

```
No Interfaces to remove.

Unassigned:
  Monitored:
   [1] GigabitEthernet0/1
Add Interface:
```

Step 9 Enter **1** to add GigabitEthernet0/1 to virtual sensor vs0.

```
Add Interface: 1

 Virtual Sensor: vs0
  Anomaly Detection: ad0
  Event Action Rules: rules0
  Signature Definitions: sig0
  Monitored:
   GigabitEthernet0/1

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

Step 10 Press **Enter** to exit the interface and virtual sensor configuration menu.

The following option appears:

```
Modify default threat prevention settings?[no]:
```

Step 11 Enter **yes** 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.

The following appears:

```
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 12 Enter **yes** to disable automatic threat prevention on all virtual sensors; otherwise, press **Enter** to accept the default of no.

The following completed configuration appears:

```
The following configuration was entered.

service host
network-settings
host-ip 192.168.1.2/24,192.168.1.1
host-name nme-ips
telnet-option enabled
access-list 10.0.0.0/8
access-list 64.0.0.0/8
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
service analysis-engine
virtual-sensor vs0
physical-interface GigabitEthernet0/1
exit
exit
service event-action-rules rules0
overrides
override-item-status Enabled
risk-rating-range 90-100
exit
exit

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

Step 13 Enter **2** to save the configuration.

```
Enter your selection[2]: 2
Configuration Saved.
```

Step 14 Reboot the NME IPS.

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

Step 15 Enter **yes** to continue the reboot.

Step 16 Apply the most recent service pack and signature update.

You are now ready to configure your NME IPS for intrusion prevention.

For More Information

For the procedure for obtaining the most recent IPS software, see Obtaining Cisco IPS Software, page 21-1.

Verifying Initialization

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 Mar 28 19:24:58 2008
! ------------------------------
! Version 6.1(1)
! Host:
!     Realm Keys          key1.0
! Signature Definition:
!     Signature Update    S310.0   2007-12-05
```

```
!     Virus Update        V1.2     2005-11-24
! ------------------------------
service interface
exit
! ------------------------------
service authentication
exit
! ------------------------------
service event-action-rules rules0
exit
! ------------------------------
service host
network-settings
host-ip 10.89.147.24/25,10.89.147.126
host-name sensor
telnet-option enabled
access-list 0.0.0.0/0
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 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
MD5: C4:BC:F2:92:C2:E2:4D:EB:92:0F:E4:86:53:6A:C6:01
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 procedures for logging in to the various sensor platforms, see Chapter 2, "Logging In to the Sensor."

CHAPTER 4

Setting Up the Sensor

This chapter contains procedures for the setting up the sensor, such 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. It contains the following sections:

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 basic sensor settings, and contains the following topics:

Changing the Hostname

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.

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.

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: 10.89.130.108/23,10.89.130.1 default:
      10.1.9.201/24,10.1.9.1
      host-name: firesafe default: sensor
      telnet-option: enabled 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: 10.89.130.108/23,10.89.130.1 default:
      10.1.9.201/24,10.1.9.1
      host-name: sensor <defaulted>
      telnet-option: enabled 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.

For More Information

For the procedure for initializing your sensor, see Chapter 3, "Initializing the Sensor."

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 10.1.9.201/24,10.1.9.1.

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 10.89.146.110/24,10.89.146.254
```

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: 10.89.146.110/24,10.89.146.254
      default: 10.1.9.201/24,10.1.9.1
      host-name: sensor default: sensor
      telnet-option: enabled 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 10.1.9.201/24,10.1.9.1.

```
sensor(config-hos-net)# show settings
   network-settings
   ---------------------------------------------
      host-ip: 10.1.9.201/24,10.1.9.1 <defaulted>
      host-name: sensor default: sensor
      telnet-option: enabled 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.

For More Information

For the procedure for initializing your sensor, see Chapter 3, "Initializing the Sensor."

Enabling and Disabling Telnet

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.

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.

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: 10.89.130.108/23,10.89.130.1
    default: 10.1.9.201/24,10.1.9.1
    host-name: sensor default: sensor
    telnet-option: enabled 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 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.

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 initializing your sensor, see Chapter 3, "Initializing the Sensor."
- For the procedure for configuring the access list, see Changing the Access List, page 4-5.

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 IDM and IME, that need to access your sensor from a web browser.
- Management stations, such as 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.

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.

```
sensor(config-hos-net)# access-list 10.89.146.110/32
```

The netmask for a single host is 32.

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

```
sensor(config-hos-net)# show settings
   network-settings
   ----------------------------------------------
      host-ip: 10.1.9.201/24,10.1.9.1 <defaulted>
      host-name: sensor <defaulted>
      telnet-option: enabled default: disabled
      access-list (min: 0, max: 512, current: 2)
      ---------------------------------------------
         network-address: 10.1.9.0/24
         ---------------------------------------------
         network-address: 10.89.146.110/32
         ---------------------------------------------
      ---------------------------------------------
      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 10.89.146.110/32
```

Step 6 Verify the entry has been removed.

```
sensor(config-hos-net)# show settings
   network-settings
   ----------------------------------------------
      host-ip: 10.1.9.201/24,10.1.9.1 <defaulted>
      host-name: sensor <defaulted>
      telnet-option: enabled 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)#
```

The host is no longer in the list.

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: 10.89.130.108/23,10.89.130.1
      default: 10.1.9.201/24,10.1.9.1
      host-name: sensor <defaulted>
      telnet-option: enabled default: disabled
      access-list (min: 0, max: 512, current: 0)
      ------------------------------------------------
      ------------------------------------------------
      ftp-timeout: 300 seconds <defaulted>
      login-banner-text: <defaulted>
   ------------------------------------------------
sensor(config-hos-net)#
```

There are no hosts or networks in the list.

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.

For More Information

For the procedure for initializing your sensor, see Chapter 3, "Initializing the Sensor."

Changing the FTP Timeout

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.

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

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: 10.89.130.108/23,10.89.130.1
      default: 10.1.9.201/24,10.1.9.1
      host-name: sensor default: sensor
```

```
      telnet-option: enabled 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: 10.89.130.108/23,10.89.130.1
      default: 10.1.9.201/24,10.1.9.1
      host-name: sensor default: sensor
      telnet-option: enabled 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.

For More Information

For the procedure for initializing your sensor, see Chapter 3, "Initializing the Sensor."

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: 10.89.130.108/23,10.89.130.1
      default: 10.1.9.201/24,10.1.9.1
      host-name: sensor default: sensor
      telnet-option: enabled 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: 10.89.130.108/23,10.89.130.1
      default: 10.1.9.201/24,10.1.9.1
      host-name: sensor default: sensor
      telnet-option: enabled 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.

Changing Web Server Settings

After you run the **setup** command, you can change the following web server settings: the web server port, whether TLS encryption is being used, and the HTTP server header message.

You can also enable RDEP event server subscriptions if you are using a third-party event client that is only able to parse IDS 4.x alerts.

Note The RDEP event interface was deprecated in Cisco IPS 5.0 and replaced by SDEE/CIDEE.

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

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**.

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 the following 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 or disable TLS,

```
sensor(config-web)# enable-tls {true | false}
```

If you disable TLS, you receive the following 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 Enable RDEP event server subscriptions if you are using a third-party event client that is only able to parse IDS 4.x alerts.

```
sensor(config-web)# configurable-service rdep-event-server
sensor(config-web-con)# enabled true
```

Step 7 Verify the web server changes.

```
sensor(config-web)# show settings
   enable-tls: true default: true
   port: 8001 default: 443
   server-id: Nothing to see here. Move along. default: HTTP/1.1 compliant
sensor(config-web)#
```

Step 8 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 9 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: 1)
   ----------------------------------------------
      <protected entry>
      service-name: rdep-event-server
      ----------------------------------------------
         enabled: true default: false
         file-name: event-server <protected>
      ----------------------------------------------
   ----------------------------------------------
sensor(config-web)#
```

Step 10 Exit web server submode.

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

Step 11 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 initializing your sensor, see Chapter 3, "Initializing the Sensor."
- For the procedure for resetting the appliance, see Resetting the Appliance, page 16-39.
- For the procedure for resetting the AIM IPS, see Rebooting, Resetting, and Shutting Down the AIM IPS, page 17-17.
- For the procedure for resetting the AIP SSM, see Reloading, Shutting Down, Resetting, and Recovering the AIP SSM, page 18-14.
- For the procedure for resetting the IDSM2, see Resetting the IDSM2, page 19-41.
- For more information about RDEP and SDEE/CIDEE, see Appendix A, "System Architecture."

Configuring User Parameters

The following section explains how to create the service account, create users, configure and recover passwords, specify privilege level, and view a list of users. It contains the following topics:

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. The password must conform to the requirements set by the sensor administrator.

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

- `Error: setEnableAuthenticationTokenStatus : The password is too short.`
- `Error: setEnableAuthenticationTokenStatus : 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. To give an existing user service privileges, you must remove that user and then use the **username** command to create 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.

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
```

Note 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. The password must conform to the requirements set by the sensor administrator.

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.

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

A list of users is displayed.

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.

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

The user `jsmith` has been removed.

For More Information

For the procedure for creating the service account, see Creating the Service Account, page 4-14.

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.

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.

Note 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 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.

```
sensor(config)# user username privilege service
```

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.

Step 4 Specify a password when prompted.

The password must conform to the requirements set by the sensor administrator. 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 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 reimaged to guarantee proper operation.
*****************************************************************************************
```

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 8 to 32 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:

a. Log in to the CLI using an account with administrator privileges.

b. Enter configuration mode.

```
sensor# configure terminal
```

c. Change the password for a specific user.

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

Note This example modifies the password for the user "tester."

Step 2 To change your password, follow these steps:

a. Log in to the CLI.

b. Enter configuration mode.

```
sensor# configure terminal
```

c. Change your password.

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

Changing User Privilege Levels

Use the **privilege** command to change the privilege level—administrator, operator, viewer—for a user.

Note You cannot use the **privilege** command to give a user service privileges. 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.

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 user's privilege has been changed.

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

    CLI ID   User       Privilege
*   13491    cisco      administrator
             jsmith     operator
```

```
            operator   operator
            service    service
            viewer     viewer
sensor#
```

The privilege of the user `jsmith` has been changed from `viewer` to `operator`.

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 4-14.

Showing User Status

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 * 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.

All IPS platforms allow ten concurrent log in sessions.

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.

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

The account of the user `jsmith` is locked.

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 administrator, you can configure how passwords are created. All user-created passwords must conform to the policy that you set up. 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.

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.

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.

```
sensor(config-aut-pas)# digits-min 6
```

The range is 0 to 64.

Step 4 Set the minimum number of nonalphanumeric printable characters that must be in a password.

```
sensor(config-aut)# other-min 3
```

The range is 0 to 64.

Step 5 Set the minimum number of uppercase alphabet characters that must be in a password.

```
sensor(config-aut)# uppercase-min 3
```

The range is 0 to 64.

Step 6 Set the minimum number of lower-case alphabet characters that must be in a password.

```
sensor(config-aut)# lowercase-min 3
```

Step 7 Set the number of old passwords to remember for each account.

```
sensor(config-aut)# number-old-passwords 3
```

A new password cannot match any of the old passwords of an account.

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)#
```

Configuring Account Locking

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 To 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.

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
```

The account of the user `jsmith` is locked as indicated by the parenthesis.

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: ******
```

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

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.

Note Administrators may need to disable the password recovery feature for security reasons.

Table 4-1 lists the password recovery methods according to platform.

Table 4-1 Password Recovery Methods According to Platform

Platform	Description	Recovery Method
4200 series sensors	Standalone IPS appliances	GRUB prompt or ROMMON
AIM IPS NME IPS	Router IPS modules	Bootloader command

Table 4-1 Password Recovery Methods According to Platform (continued)

Platform	Description	Recovery Method
AIP SSM	ASA 5500 series adaptive security appliance modules	ASA CLI command
IDSM2	Switch IPS module	Password recovery image file

For More Information

For More information on when to disable the password recovery features, see Disabling Password Recovery, page 4-27.

Password Recovery for Appliances

This section describes the two ways to recover the password for appliances. It contains the following topics:

Using the GRUB Menu

For 4200 series 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.

Note You must have a terminal server or direct serial connection to the appliance to use the GRUB menu to recover the password.

To recover the password on appliances, follow these steps:

Step 1 Reboot the appliance.

The following menu appears:

```
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**. You can change the password the next time you log in to the CLI.

For More Information

For more information on connecting a terminal server or direct serial connection, see Connecting an Appliance to a Terminal Server, page 22-13.

Using ROMMON

For the IPS 4240 and the IPS 4255 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.

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-4240-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
```

Password Recovery for the AIM IPS

To recover the password for the AIM IPS, use the **clear password** command. You must have console access to the AIM IPS and administrative access to the router. To recover the password for the AIM IPS, follow these steps:

Step 1 Log in to the router.

Step 2 Enter privileged EXEC mode on the router.

```
router> enable
```

Step 3 Confirm the module slot number in your router.

```
router# show run | include ids-sensor
interface IDS-Sensor0/0
router#
```

Step 4 Session in to the AIM IPS.

```
router# service-module ids-sensor slot/port session
```

Example:

```
router# service-module ids-sensor 0/0 session
```

Step 5 Press **Control-shift-6** followed by **x** to navigate to the router CLI.

Step 6 Reset the AIM IPS from the router console.

```
router# service-module ids-sensor 0/0 reset
```

Step 7 Press **Enter** to return to the router console.

Step 8 When prompted for boot options, enter ******* quickly.

You are now in the bootloader.

Step 9 Clear the password.

```
ServicesEngine boot-loader# clear password
```

The AIM IPS reboots. The password is reset to **cisco**. Log in to the CLI with username **cisco** and password **cisco**. You can then change the password.

Password Recovery for the AIP SSM

You can reset the password to the default (**cisco**) for the AIP SSM 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 7.2.2 or later.

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.
```

Resetting the Password Using the CLI

To reset the password on the AIP SSM, follow these steps:

Step 1 Log into the adaptive security appliance and enter the following command to verify the module slot number:

```
asa# show module
Mod Card Type                                    Model              Serial No.
--- -------------------------------------------- ------------------ -----------
  0 ASA 5510 Adaptive Security Appliance         ASA5510            JMX1135L097
  1 ASA 5500 Series Security Services Module-40  ASA-SSM-40         JAF1214AMRL

Mod MAC Address Range                 Hw Version   Fw Version   Sw Version
--- --------------------------------- ------------ ------------ ---------------
  0 001b.d5e8.e0c8 to 001b.d5e8.e0cc  2.0          1.0(11)2     8.4(3)
  1 001e.f737.205f to 001e.f737.205f  1.0          1.0(14)5     7.0(7)E4

Mod SSM Application Name           Status           SSM Application Version
--- ------------------------------ ---------------- -------------------------
  1 IPS                            Up               7.0(7)E4

Mod Status             Data Plane Status     Compatibility
--- ------------------ --------------------- -------------
  0 Up Sys             Not Applicable
  1 Up                 Up
```

Step 2 Reset the password for module 1.

```
asa# hw-module module 1 password-reset
Reset the password on module in slot 1? [confirm]
```

Step 3 Press **Enter** to confirm.

```
Password-Reset issued for slot 1.
```

Step 4 Verify the status of the module. Once the status reads Up, you can session to the AIP SSM.

```
asa# show module 1
Mod Card Type                                    Model              Serial No.
--- -------------------------------------------- ------------------ -----------
  1 ASA 5500 Series Security Services Module-40  ASA-SSM-40         JAF1214AMRL

Mod MAC Address Range                 Hw Version   Fw Version   Sw Version
--- --------------------------------- ------------ ------------ ---------------
  1 001e.f737.205f to 001e.f737.205f  1.0          1.0(14)5     7.0(7)E4

Mod SSM Application Name           Status           SSM Application Version
--- ------------------------------ ---------------- -------------------------
  1 IPS                            Up               7.0(7)E4

Mod Status             Data Plane Status     Compatibility
--- ------------------ --------------------- -------------
  1 Up                 Up
```

Step 5 Session to the AIP SSM.

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

Step 6 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 7 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.
aip_ssm#
```

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.

Password Recovery for the IDSM2

To recover the password for the IDSM2, you must install a special password recovery image file. This installation only resets the password, all other configuration remains intact. The password recovery image is version-dependent and can be found on the Cisco Download Software site. For IPS 6.*x*, download WS-SVC-IDSM2-K9-a-6.0-password-recovery.bin.gz. For IPS 7.*x*, download WS-SVC-IDSM2-K9-a-7.0-password-recovery.bin.gz.

FTP is the only supported protocol for image installations, so make sure you put the password recovery image file on an FTP server that is accessible to the switch. You must have administrative access to the Cisco 6500 series switch to recover the password on the IDSM2.

During the password recovery image installation, the following message appears:

```
Upgrading will wipe out the contents on the hard disk.
Do you want to proceed installing it [y|n]:
```

This message is in error. Installing the password recovery image does not remove any configuration, it only resets the login account.

Once you have downloaded the password recovery image file, follow the instructions to install the system image file but substitute the password recovery image file for the system image file. The IDSM2 should reboot into the primary partition after installing the recovery image file. If it does not, enter the following command from the switch:

```
hw-module module module_number reset hdd:1
```

Note The password is reset to **cisco**. Log in to the CLI with username **cisco** and password **cisco**. You can then change the password.

For More Information

- For the procedures for reimaging the IDSM2, see Installing the IDSM2 System Image, page 22-26.
- For more information on downloading Cisco IPS software, see Obtaining Cisco IPS Software, page 21-1.

Password Recovery for the NME IPS

To recover the password for the NME IPS, use the **clear password** command. You must have console access to the NME IPS and administrative access to the router.

To recover the password for the NME IPS, follow these steps:

Step 1 Log in to the router.

Step 2 Enter privileged EXEC mode on the router.

```
router> enable
```

Step 3 Confirm the module slot number in your router.

```
router# show run | include ids-sensor
interface IDS-Sensor1/0
router#
```

Step 4 Session in to the NME IPS.

```
router# service-module ids-sensor slot/port session
```

Example:

```
router# service-module ids-sensor 1/0 session
```

Step 5 Press **Control-shift-6** followed by **x** to navigate to the router CLI.

Step 6 Reset the NME IPS from the router console.

```
router# service-module ids-sensor 1/0 reset
```

Step 7 Press **Enter** to return to the router console.

Step 8 When prompted for boot options, enter ******* quickly.

You are now in the bootloader.

Step 9 Clear the password.

```
ServicesEngine boot-loader# clear password
```

The NME IPS reboots. The password is reset to **cisco**. Log in to the CLI with username **cisco** and password **cisco**. You can then change the password.

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, 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 IDM

To disable password recovery in IDM or IME, follow these steps:

Step 1 Log in to 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.

For More Information

- To determine whether password recovery is enabled or disabled, see Verifying the State of Password Recovery, page 4-28.
- For more information on what to do if you forget the password and password recovery is set to disabled, see Troubleshooting Password Recovery, page 4-28.

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, such as the AIM IPS and the NME IPS bootloader, ROMMON, and the maintenance partition for the IDSM2, 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.
- When performing password recovery on the IDSM2, you see the following message: `Upgrading will wipe out the contents on the storage media`. You can ignore this message. Only the password is reset when you use the specified password recovery image.

For More Information

- For information on reimaging the sensor, see Chapter 22, "Upgrading, Downgrading, and Installing System Images."
- For more information on disabling password recovery, see Disabling Password Recovery, page 4-27.
- For the procedure for checking the state of password recovery, see Verifying the State of Password Recovery, page 4-28.

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

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.

Note We recommend that you use an NTP server. 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.

Here is a summary of ways to set the time on sensors:

- For appliances
  - Use the **clock set** command to set the time. This is the default.
  - Use NTP—You can configure the appliance to get its time from an NTP time synchronization source.
- For the IDSM2
  - The IDSM2 can automatically synchronize its clock with the switch time. This is the default. The UTC time is synchronized between the switch and the IDSM2. The time zone and summertime settings are not synchronized between the switch and the IDSM2.

    Note Be sure to set the time zone and summertime settings on both the switch and the IDSM2 to ensure that the UTC time settings are correct. The local time of the IDSM2 could be incorrect if the time zone and/or summertime settings do not match between the IDSM2 and the switch.

  - Use NTP—You can configure the IDSM2 to get its time from an NTP time synchronization source.
- For the AIM IPS and the NME IPS
  - The AIM IPS and the NME IPS can automatically synchronize their clock with the clock in the router chassis in which they are installed (parent router). This is the default. The UTC time is synchronized between the parent router and the AIM IPS and the NME IPS. The time zone and summertime settings are not synchronized between the parent router and the AIM IPS and the NME IPS.

    Note Be sure to set the time zone and summertime settings on both the parent router and the AIM IPS and the NME IPS to ensure that the UTC time settings are correct. The local time of the AIM IPS and the NME IPS could be incorrect if the time zone and/or summertime settings do not match between the AIM IPS and the NME IPS and the router.

  - Use NTP—You can configure the AIM IPS and the NME IPS to get their time from an NTP time synchronization source, such as a Cisco router, other than the parent router.
- For the AIP SSM
  - The AIP SSM can automatically synchronize its clock with the clock in the adaptive security appliance in which it is installed. This is the default. The UTC time is synchronized between the adaptive security appliance and the AIP SSM. The time zone and summertime settings are not synchronized between the adaptive security appliance and the AIP SSM.

    Note Be sure to set the time zone and summertime settings on both the adaptive security appliance and the AIP SSM to ensure that the UTC time settings are correct. The local time of the AIP SSM could be incorrect if the time zone and/or summertime settings do not match between the AIP SSM and the adaptive security appliance.

  - Use NTP—You can configure the AIP SSM to get its 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

All IPS modules (AIM IPS, AIP SSM, IDSM2, and NME IPS) synchronize their system clocks to the parent chassis clock (switch, router, or security appliance) each time the module boots up and any time the parent chassis clock is set. The module 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 module clock and the parent clock are synchronized to an external NTP server. If only the module clock or only the parent chassis clock is synchronized to an NTP server, the time drift occurs.

For More Information

- For more information on NTP, see Configuring NTP, page 4-38.
- For More information on verifying that the module and the NTP server are synchronized, see Verifying the Sensor is Synchronized with the NTP Server, page C-17.

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 more information on the **clear events** command, see Clearing Events from Event Store, page 16-21.

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

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 4-2 lists the system clock flags.

Table 4-2 System Clock Flags

Symbol	Description
*	Time is not authoritative.

Table 4-2 System Clock Flags (continued)

Symbol	Description
(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.

```
sensor# show clock detail
20:09:43 UTC Thu Apr 03 2008
Time source is NTP
Summer time starts 03:00:00 UTC Sun Mar 09 2008
Summer time stops 01:00:00 UTC Sun Nov 02 2008
```

This 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 2008
No time source
Summer time starts 03:00:00 UTC Sun Mar 09 2008
Summer time stops 01:00:00 UTC Sun Nov 02 2008
```

This indicates that no time source is configured.

Manually Setting the System Clock

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.

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.

The **clock set** command does not apply to the following platforms:

- AIM IPS
- AIP SSM
- IDSM2
- NME IPS

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 2008
```

Note The time format is 24-hour time.

For More Information

- For the procedure for configuring NTP, see Configuring NTP, page 4-38.
- For an explanation of the importance of having a valid time source for the sensor, see Time Sources and the Sensor, page 4-29.
- For an explanation of what to do if you set the clock incorrectly, see Correcting Time on the Sensor, page 4-31.

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
```

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
```

c. 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
```

d. 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
```

Note Changing the time zone offset requires the sensor to reboot.

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
```

Note Changing the time zone offset requires the sensor to reboot.

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
```

Note Changing the time zone offset requires the sensor to reboot.

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 4-38
- Configuring the Sensor to Use an NTP Time Source, page 4-39

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.

```
router(config)# ntp authentication-key key_ID md5 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.

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).

```
router(config)# ntp trusted-key key_ID
```

The trusted key ID is the same number as the key ID in Step 3.

Example

```
router(config)# ntp trusted-key 100
```

Step 5 Specify the interface on the router that the sensor will communicate with.

```
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.

```
router(config)# ntp master stratum_number
```

Example

```
router(config)# ntp master 6
```

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.

For More Information

For the procedure for using authenticated NTP, see Configuring the Sensor to Use an NTP Time Source, page 4-39.

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 For 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 For authenticated NTP:

a. Enter NTP configuration mode.

```
sensor(config-hos)# ntp-option enable
```

b. Specify the NTP server IP address and key ID.

```
sensor(config-hos-ena)# ntp-servers ip_address key-id 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.

Example:

```
sensor(config-hos-ena)# ntp-servers 10.16.0.0 key-id 100
```

c. Specify the key value NTP server.

```
sensor(config-hos-ena)# ntp-keys key_ID md5-key key_value
```

The key value is text (numeric or character). This is the key value that you already set up on the NTP server.

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

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.

SSH authenticates the hosts or networks using one or both of the following:

- Password
- User RSA public key

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.

Note SSH never sends passwords in clear text.

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, PIX Firewalls, and Catalyst switches that the sensor will connect to for blocking.

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 MD5 fingerprint and bubble babble for the requested IP address. You can then select to add the key to the list.

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 by 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 a key of an attacker.

Note 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.

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
MD5 fingerprint is F3:10:3E:BA:1E:AB:88:F8:F5:56:D3:A6:63:42:1C:11
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 MD5 fingerprint appears. You are prompted to add it to the known hosts list:

If the host is not accessible when the command is issued, the following 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
601068614812749969593513740598331393154884988302302182922353335152653860589163651944997842
874583627883277460138506084043415861927
```

```
MD5: 49:3F:FD:62:26:58:94:A3:E9:88:EF:92:5F:52:6E:7B
Bubble Babble: xebiz-vykyk-fekuh-rukuh-cabaz-paret-gosym-serum-korus-fypop-huxyx
sensor#
```

Step 7 Remove an entry.

```
sensor(config)# no ssh host-key 10.16.0.0
```

The host is removed from the SSH known hosts list.

Step 8 Verify the host was removed.

```
sensor(config)# exit
sensor# show ssh host-keys
```

The IP address no longer appears in the list.

Adding SSH Authorized Public Keys

Use the **ssh authorized-key** command to define public keys for a client allowed to use RSA authentication to log in to the local SSH server.

The following options apply:

- *id*—1 to 256-character string that uniquely identifies the authorized key. You can use numbers, "_," and "-," but spaces and "?" are not acceptable.
- *key-modulus-length*—An ASCCI decimal integer in the range[511, 2048].
- *public-exponent*—An ASCII decimal integer in the range [3, 2^32].
- *public-modulus*—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.

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.

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.

Note 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.

To add a key entry to the SSH 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.

```
sensor# configure terminal
sensor(config)# ssh authorized-key system1 1023 37
66022272955660983338089706716372943357082868686000817201780243492180421420781303592082950 9
10170135848052503999393211250314745276837862091118998665371608981314792208604473991134136 9
64287068231936192814852186409455741630613878646833511583591040494021313695435339616344979 3
49705016792583146548622146467421997057
sensor(config)#
```

Step 3 Verify that the key was added.

```
sensor(config)# exit
sensor# show ssh authorized-keys
system1
sensor#
```

Step 4 View the key for a specific ID.

```
sensor# show ssh authorized-keys system1
1023 37 66022272955660983338089706716372943357082868686000817201780243492180421 4
2078130359208295091017013584805250399939321125031474527683786209111899866537160 8
9813147922086044739911341369642870682319361928148521864094557416306138786468335 1
1583591040494021313695435339616344979349705016792583146548622146467421997057
sensor#
```

Step 5 Remove an entry from the list of SSH authorized keys.

```
sensor# configure terminal
sensor(config)# no ssh authorized-key system1
```

The key is removed from the SSH authorized keys list.

Step 6 Verify the entry was removed.

```
sensor(config)# exit
sensor# show ssh authorized-keys
```

The key system1 no longer appears in the list:

If you enter the former ID, you receive an error message:

```
sensor# show ssh authorized-keys system1
Error: Requested id does not exist for the current user.
sensor#
```

Generating a New SSH Server Key

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 1.5.

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
MD5: 93:F5:51:58:C7:FD:40:8C:07:26:5E:29:13:C8:33:AE
Bubble Babble: ximal-sudez-kusot-gosym-levag-fegoc-holez-cakar-kunel-nylis-kyxox
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.

```
sensor# show ssh server-key
1024 35
137196765426571419509124895787229630062726389801071715581921573847280637533000158590028798
074385824867184332364758899959675370523879609376174812179228415215782949029183962207840731
771645803509837259475421477212459797170806510716077556010753169312675023860474987441651041
217710152766990480431898217878170000647
MD5: 93:F5:51:58:C7:FD:40:8C:07:26:5E:29:13:C8:33:AE
Bubble Babble: ximal-sudez-kusot-gosym-levag-fegoc-holez-cakar-kunel-nylis-kyxox
sensor#
```

For More Information

For the procedure for updating the known hosts table, see Adding Hosts to the SSH Known Hosts List, page 4-42.

Configuring TLS

This section describes TLS on the sensor, and contains the following topics:

Understanding TLS

Cisco IPS 6.1 contains a web server that is running 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 IDM because it does not trust the CA.

Note 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 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 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 by 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 MD5 fingerprint is 4F:BA:15:67:D3:E6:FB:51:8A:C4:57:93:4D:F2:83:FE
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 MD5 and 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.

```
Certificate ID: 10.89.146.110 successfully added to the TLS trusted host table.
sensor(config)#
```

The host has been added to the TLS trusted host list. The Certificate ID stored for the requested certificate is displayed when the command is successful.

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
MD5: 4F:BA:15:67:D3:E6:FB:51:8A:C4:57:93:4D:F2:83:FE
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
```

The host is removed from the trusted hosts list.

Step 7 Verify the entry was removed from the trusted host list.

```
sensor(config)# exit
sensor# show tls trusted-hosts
No entries
```

The IP address no longer appears in the list:

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.

Note 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
MD5 fingerprint is FD:83:6E:41:D3:88:48:1F:44:7F:AF:5D:52:60:89:DE
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
MD5: FD:83:6E:41:D3:88:48:1F:44:7F:AF:5D:52:60:89:DE
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 4-46.

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

Although the sensor functions without the license key, you must have a license key to obtain signature updates. 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 IDM or IME, for IDM choose **Configuration > Sensor Management > Licensing**, and for 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:

- IDM Home window Licensing section on the Health tab
- IDM Licensing pane (**Configuration > Licensing**)
- IME Home page in the Device Details section on the Licensing tab
- License Notice at CLI login

Whenever you start IDM, 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 IDM, 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 IDM or 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.

For More Information

- For information on Cisco service programs, see Service Programs for IPS Products, page 4-50.
- For the procedure for obtaining and installing the license key, see Obtaining and Installing the License Key, page 4-50.

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 4240
- IPS 4255
- IPS 4260
- IPS 4270-20
- AIM IPS
- IDSM2
- NME IPS

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 the AIP SSM installed, or if you purchase the AIP SSM 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 purchased an ASA 5510 and then later wanted to add IPS and purchased an ASA-SSM-AIP-10-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 will change. You must then get a new license key for the new serial number.

Obtaining and Installing the License Key

Use the **copy** *source-url license_file_name* **license-key** command to copy the license key to your sensor.

The following options 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.

Note You cannot install an older license key over a newer license key.

The exact format of the source and destination URLs varies according to the file. Here are the valid types:

- 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.

Installing the License Key

To install the license key, follow these steps:

Step 1 Apply for the license key at this URL: 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 2 Fill in the required fields.

Note You must have the correct IPS device serial number because the license key only functions on the device with that number.

Your Cisco IPS Signature Subscription Service license key will be sent by e-mail to the e-mail address you specified.

Step 3 Save the license key to a system that has a web server, FTP server, or SCP server.

Step 4 Log in to the CLI using an account with administrator privileges.

Step 5 Copy the license key to the sensor.

```
sensor# copy scp://user@10.89.147.3://tftpboot/dev.lic license-key
Password: *******
```

Step 6 Verify the sensor is licensed.

```
sensor# show version
Application Partition:

Cisco Intrusion Prevention System, Version 6.1(1)E1

Host:
    Realm Keys          key1.0
Signature Definition:
    Signature Update    S391.0                  2008-04-16
    Virus Update        V1.2                    2005-11-24
OS Version:             2.4.30-IDS-smp-bigphys
Platform:               ASA-SSM-20
Serial Number:          P300000220
Sensor up-time is 3 days.
Using 1031888896 out of 2093682688 bytes of available memory (49% usage)
system is using 17.8M out of 29.0M bytes of available disk space (61% usage)
application-data is using 52.4M out of 166.6M bytes of available disk space (33% usage)
boot is using 37.8M out of 68.5M bytes of available disk space (58% usage)


MainApp          N-2007_JUN_19_16_45   (Release)   2007-06-19T17:10:20-0500   Running
AnalysisEngine   N-2007_JUN_19_16_45   (Release)   2007-06-19T17:10:20-0500   Running
CLI              N-2007_JUN_19_16_45   (Release)   2007-06-19T17:10:20-0500


Upgrade History:

  IPS-K9-6.1-1-E1   15:36:05 UTC Thu Apr 24 2008

Recovery Partition Version 1.1 - 6.1(1)E1

Host Certificate Valid from: 25-Apr-2008 to 26-Apr-2010

sensor#
```

Step 7 Copy your license key from a sensor to a server to keep a backup copy of the license.

```
sensor# copy license-key scp://user@10.89.147.3://tftpboot/dev.lic
Password: *******
sensor#
```

For More Information

- For the procedure for adding remote hosts to the SSH known hosts list, see Adding Hosts to the SSH Known Hosts List, page 4-42.
- For the procedure for making a remote host a TLS trusted host, see Adding TLS Trusted Hosts, page 4-46.

CHAPTER 5

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. contains the following sections:

Understanding Interfaces

This section describes IPS interfaces and modes, and contains the following topics:

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 PCI expansion slots are numbered beginning with slot 1 for the bottom slot with the slot numbers increasing from bottom to top (except for the IPS 4270-20, where the ports are numbered from top to bottom). Interfaces with a given slot are numbered beginning with port 0 for the right port with the port numbers increasing from right to left. For example, GigabitEthernet2/1 supports a maximum speed of 1 Gigabit and is the second-from-the-right interface in the second-from-the bottom PCI expansion slot. The IPS 4240, IPS 4255, IPS 4260, and IPS 4270-20 are exceptions to this rule. The command and control interface on these sensors is called Management0/0 rather than GigabitEthernet0/0. The IPS 4270-20 has an additional interface called Management0/1, which is reserved for future use.

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:

- Because the AIM IPS, AIP SSM, and NME IPS only have one sensing interface, you cannot configure a TCP reset interface.
- Because of hardware limitations on the Catalyst switch, both of the IDSM2 sensing interfaces are permanently configured to use System0/1 as the TCP reset interface.
- 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.

Note Each physical interface can be divided into VLAN group subinterfaces, each of which consists of a group of VLANs on that interface.

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
AIM IPS	Management0/0
AIP SSM-10	GigabitEthernet0/0
AIP SSM-20	GigabitEthernet0/0
AIP SSM-40	GigabitEthernet0/0
IDSM2	GigabitEthernet0/2
IPS 4240	Management0/0
IPS 4255	Management0/0
IPS 4260	Management0/0
IPS 4270-20	Management0/0
NME IPS	Management0/1

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 5-5.
- For more information on interface modes, see Configuring Promiscuous Mode, page 5-15, Configuring Inline Interface Mode, page 5-16, Configuring Inline VLAN Pair Mode, page 5-20, Configuring VLAN Group Mode, page 5-25, Configuring Inline Bypass Mode, page 5-33.

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.

With the exception of the IDSM2, any sensing interface can serve as the alternate TCP reset interface for another sensing interface. The alternate TCP reset interface on the IDSM2 is fixed because of hardware limitation.

Note There is only one sensing interface on IPS modules (AIM IPS, AIP SSM, and NME IPS).

Table 5-2 lists the alternate TCP reset interfaces.

Table 5-2 Alternate TCP Reset Interfaces

Sensor	Alternate TCP Reset Interface
AIM IPS	None
AIP SSM-10	None
AIP SSM-20	None
AIP SSM-40	None
IDSM2	System0/1[1]
IPS 4240	Any sensing interface
IPS 4255	Any sensing interface
IPS 4260	Any sensing interface
IPS 4270-20	Any sensing interface
NME IPS	None

1. This is an internal interface on the Catalyst backplane.

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

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.

  Note 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.

  Note Taps do not permit incoming traffic from the sensor.

  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 6.1.

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)
AIM IPS	—	GigabitEthernet0/1 by **ids-service-module** command in the router configuration instead of VLAN pair or inline interface pair	GigabitEthernet0/1 by **ids-service-module** command in the router configuration instead of VLAN pair or inline interface pair	Management0/0
AIP SSM-10	—	GigabitEthernet0/1 by security context instead of VLAN pair or inline interface pair	GigabitEthernet0/1 by security context instead of VLAN pair or inline interface pair	GigabitEthernet0/0
AIP SSM-20	—	GigabitEthernet0/1 by security context instead of VLAN pair or inline interface pair	GigabitEthernet0/1 by security context instead of VLAN pair or inline interface pair	GigabitEthernet0/0
AIP SSM-40	—	GigabitEthernet0/1 by security context instead of VLAN pair or inline interface pair	GigabitEthernet0/1 by security context instead of VLAN pair or inline interface pair	GigabitEthernet0/0
IDSM2	—	GigabitEthernet0/7 GigabitEthernet0/8	0/7<->0/8	GigabitEthernet0/2

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 4240	—	GigabitEthernet0/0 GigabitEthernet0/1 GigabitEthernet0/2 GigabitEthernet0/3	0/0<->0/1 0/0<->0/2 0/0<->0/3 0/1<->0/2 0/1<->0/3 0/2<->0/3	Management0/0
IPS 4255	—	GigabitEthernet0/0 GigabitEthernet0/1 GigabitEthernet0/2 GigabitEthernet0/3	0/0<->0/1 0/0<->0/2 0/0<->0/3 0/1<->0/2 0/1<->0/3 0/2<->0/3	Management0/0
IPS 4260	—	GigabitEthernet0/1	N/A	Management0/0
IPS 4260	4GE-BP Slot 1 Slot 2	GigabitEthernet0/1 GigabitEthernet2/0 GigabitEthernet2/1 GigabitEthernet2/2 GigabitEthernet2/3 GigabitEthernet3/0 GigabitEthernet3/1 GigabitEthernet3/2 GigabitEthernet3/3	 2/0<->2/1[1] 2/2<->2/3 3/0<->3/1 3/2<->3/3	Management0/0
IPS 4260	2SX Slot 1 Slot 2	GigabitEthernet0/1 GigabitEthernet2/0 GigabitEthernet2/1 GigabitEthernet3/0 GigabitEthernet3/1	All sensing ports can be paired together	Management0/0
IPS 4260	10GE Slot 1	GigabitEthernet0/1 TenGigabitEthernet2/0 TenGigabitEthernet2/1	 2/0<->2/1[2]	Management0/0
IPS 4270-20	—	—	N/A	Management0/0 Management0/1[3]

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 4270-20	4GE-BP Slot 1 Slot 2	 GigabitEthernet3/0 GigabitEthernet3/1 GigabitEthernet3/2 GigabitEthernet3/3 GigabitEthernet4/0 GigabitEthernet4/1 GigabitEthernet4/2 GigabitEthernet4/3	 3/0<->3/1[4] 3/2<->3/3 4/0<->4/1 4/2<->4/3	Management0/0 Management0/1[5]
IPS 4270-20	2SX Slot 1 Slot 2	 GigabitEthernet3/0 GigabitEthernet3/1 GigabitEthernet4/0 GigabitEthernet4/1	All sensing ports can be paired together	Management0/0 Management0/1[6]
IPS 4270-20	10GE Slot 1 Slot 2	 TenGigabitEthernet5/0 TenGigabitEthernet5/1 TenGigabitEthernet7/0 TenGigabitEthernet7/1	All sensing ports can be paired together	Management0/0 Management0/1[7]
NME IPS	—	GigabitEthernet0/1 by **ids-service-module** command in the router configuration instead of VLAN pair or inline interface pair	GigabitEthernet0/1 by **ids-service-module** command in the router configuration instead of VLAN pair or inline interface pair	Management0/1

1. To disable hardware bypass, pair the interfaces in any other combination (2/0<->2/2 and 2/1<->2/3, for example).
2. To disable hardware bypass, pair the interfaces in any other combination (2/0<->2/2 and 2/1<->2/3, for example).
3. Reserved for future use.
4. To disable hardware bypass, pair the interfaces in any other combination (2/0<->2/2 and 2/1<->2/3, for example).
5. Reserved for future use.
6. Reserved for future use.
7. Reserved for future use.

Note The IPS 4260 supports a mixture of 4GE-BP, 2SX, and 10GE cards. The IPS 4270-20 also supports a mixture of 4GE-BP, 2SX, and 10GE cards up to a total of either six cards, or sixteen total ports, which ever is reached first, but is limited to only two 10GE card in the mix of cards.

Hardware Bypass Mode

In addition to Cisco IPS 6.1 software bypass, the IPS 4260 and the IPS 4270-20 also support hardware bypass. This section describes the hardware bypass card and its configuration restrictions, and contains the following topics:

- Hardware Bypass Card, page 5-8
- Hardware Bypass Configuration Restrictions, page 5-9

Hardware Bypass Card

The IPS 4260 and IPS 4270-20 support the 4-port GigabitEthernet card (part number IPS-4GE-BP-INT=) with hardware bypass. This 4GE bypass interface card supports hardware bypass only between ports 0 and 1 and between ports 2 and 3.

Note To disable hardware bypass, pair the interfaces in any other combination, for example 2/0<->2/2 and 2/1<->2/3.

Hardware bypass complements the existing software bypass feature in Cisco IPS 6.1. The following conditions apply to hardware bypass and software bypass:

- When bypass is set to OFF, software bypass is not active.

  For each inline interface for which hardware bypass is available, the component interfaces are set to disable the fail-open capability. If SensorApp fails, the sensor is powered off, reset, or if the NIC interface drivers fail or are unloaded, the paired interfaces enter the fail-closed state (no traffic flows through inline interface or inline VLAN subinterfaces).

- When bypass is set to ON, software bypass is active.

  Software bypass forwards packets between the paired physical interfaces in each inline interface and between the paired VLANs in each inline VLAN subinterface. For each inline interface on which hardware bypass is available, the component interfaces are set to standby mode. If the sensor is powered off, reset, or if the NIC interfaces fail or are unloaded, those paired interfaces enter fail-open state in hardware (traffic flows unimpeded through inline interface). Any other inline interfaces enter fail-closed state.

- When bypass is set to AUTO (traffic flows without inspection), software bypass is activated if sensorApp fails.

  For each inline interface on which hardware bypass is available, the component interfaces are set to standby mode. If the sensor is powered off, reset, or if the NIC interfaces fail or are unloaded, those paired interfaces enter fail-open state in hardware. Any other inline interfaces enter the fail-closed state.

Note To test fail-over, set the bypass mode to ON or AUTO, create one or more inline interfaces and power down the sensor and verify that traffic still flows through the inline path.

For More Information

For the procedure for configuring bypass mode, see Configuring Inline Bypass Mode, page 5-33.

Hardware Bypass Configuration Restrictions

To use the hardware bypass feature on the 4GE bypass interface card, you must pair interfaces to support the hardware design of the card. If you create an inline interface that pairs a hardware-bypass-capable interface with an interface that violates one or more of the hardware-bypass configuration restrictions, hardware bypass is deactivated on the inline interface and you receive a warning message similar to the following:

```
Hardware bypass functionality is not available on Inline-interface pair0.
Physical-interface GigabitEthernet2/0 is capable of performing hardware bypass only when
paired with GigabitEthernet2/1, and both interfaces are enabled and configured with the
same speed and duplex settings.
```

The following configuration restrictions apply to hardware bypass:

- The 4-port bypass card is only supported on the IPS 4260 and IPS 4270-20.
- Fail-open hardware bypass only works on inline interfaces (interface pairs), not on inline VLAN pairs.
- Fail-open hardware bypass is available on an inline interface if all of the following conditions are met:
  - Both of the physical interfaces support hardware bypass.
  - Both of the physical interfaces are on the same interface card.
  - The two physical interfaces are associated in hardware as a bypass pair.
  - The speed and duplex settings are identical on the physical interfaces.
  - Both of the interfaces are administratively enabled.
- Autonegotiation must be set on MDI/X switch ports connected to the IPS 4260 and IPS 4270-20.

  You must configure both the sensor ports and the switch ports for autonegotiation for hardware bypass to work. The switch ports must support MDI/X, which automatically reverses the transmit and receive lines if necessary to correct any cabling problems. The sensor is only guaranteed to operate correctly with the switch if both of them are configured for identical speed and duplex, which means that the sensor must be set for autonegotiation too.

Interface Configuration Restrictions

The following restrictions apply to configuring interfaces on the sensor:

- Physical Interfaces
  - On modules (AIM IPS, AIP SSM, IDSM2, and NME IPS), 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 4240, IPS 4255, IPS 4260, and IPS 4270-20), 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**.
- 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.
- 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.

    Note The exception to this restriction is the IDSM2. The alternate TCP reset interface assignments for both sensing interfaces is System0/1 (protected).

- 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.
- CLI, IDM, and IME prompt you to remove any dangling references. You can leave the dangling references and continue editing the configuration.
- CLI, IDM, and IME do not allow configuration changes in Analysis Engine that conflict with the interface configuration.
- CLI allows configuration changes in the interface configuration that cause conflicts in the Analysis Engine configuration. IDM and IME do *not* allow changes in the interface configuration that cause conflicts in the Analysis Engine configuration.

For More Information

- For a list of supported sensor interfaces, see Interface Support, page 5-5.
- 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 5-12.

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 5-12.
- 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 5-20, Configuring VLAN Group Mode, page 5-25, and Configuring Inline Bypass Mode, page 5-33.
- For the procedure, see Adding, Editing, and Deleting Virtual Sensors, page 6-3.

Configuring Physical Interfaces

Note For information on what you need to configure if you are using the hardware bypass card on the IPS 4260 and the IPS 4270-20, see Hardware Bypass Configuration Restrictions, page 5-9.

Use the **physical-interfaces** *interface_name* command in the service interface submode to configure promiscuous interfaces. The interface name is FastEthernet or GigabitEthernet.

Note The AIP SSM is configured for promiscuous mode from the Cisco ASA CLI and not from the Cisco IPS CLI.

The following options apply:

- **admin-state {enabled | disabled}**—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 This option is protected on modules.

  - *interface_name*—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**—Your description of the promiscuous interface.
- **duplex**—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**—Remove an entry or selection setting.
- **speed**—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.

To configure the promiscuous interface 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
```

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.

```
sensor(config-int-phy)# admin-state enabled
```

You must assigned the interface to a virtual sensor and enable it before it can monitor traffic.

Step 6 Add a description of this interface.

```
sensor(config-int-phy)# description INT1
```

Step 7 Specify the duplex settings.

```
sensor(config-int-phy)# duplex full
```

This option is not available on modules.

Step 8 Specify the speed.

```
sensor(config-int-phy)# speed 1000
```

This option is not available on modules.

Step 9 Enable TCP resets for this interface if needed.

```
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 To 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 5-5.
- For the procedure for setting up traffic on the AIP SSM, see Sending Traffic to the AIP SSM, page 18-9.
- 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 more information about configuring inline VLAN pairs, see Configuring Inline VLAN Pairs, page 5-21.
- For the procedure for adding interfaces to virtual sensors, see Adding, Editing, and Deleting Virtual Sensors, page 6-3.

Configuring Promiscuous Mode

This section describes promiscuous mode on the sensor, and contains the following topics:

- Understanding Promiscuous Mode, page 5-15
- Configuring Promiscuous Mode, 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.

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.

Configuring Inline Interface Mode

This section describes inline mode on the sensor, and contains the following topics:

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 AIM IPS, AIP SSM, and NME IPS to operate inline even though these modules 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.

Configuring Inline Interface Pairs

Note For information on what you need to configure if you are using the hardware bypass card on the IPS 4260 and the IPS 4270-20, see Hardware Bypass Configuration Restrictions, page 5-9.

Use the **inline-interfaces** *name* command in the service interface submode to create inline interface pairs.

Note The AIP SSM is configured for inline interface mode from the Cisco ASA CLI and not from the Cisco IPS CLI.

The following options apply:

- **inline-interfaces** *name*—Name of the logical inline interface pair.
- **default**—Sets the value back to the system default setting.
- **description**—Your description of the inline interface pair.

- **interface1** *interface_name*—The first interface in the inline interface pair.
- **interface2** *interface_name*—The second interface in the inline interface pair.
- **no**—Removes an entry or selection setting.
- **admin-state {enabled | disabled}**—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.

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.

```
sensor(config-int-inl)# interface1 GigabitEthernet0/0
sensor(config-int-inl)# interface2 GigabitEthernet0/1
```

You must assign the interface to a virtual sensor and enable it before it can monitor traffic (see Step 10).

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 To delete an inline interface pair and return the interfaces to promiscuous mode:

```
sensor(config-int)# no inline-interfaces PAIR1
```

You must also delete the inline interface pair from the virtual sensor to which it is assigned.

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 submode.

```
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 AIP SSM, see Sending Traffic to the AIP SSM, page 18-9.
- 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 6-3.

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

Note For information on what you need to configure if you are using the hardware bypass card on the IPS 4260 and the IPS 4270-20, see Hardware Bypass Configuration Restrictions, page 5-9.

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.

Note Inline VLAN pairs are supported on all sensors that are compatible with Cisco IPS 6.1 except on the AIM IPS, AIP SSM, and NME IPS.

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.

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 options apply:

- **admin-state {enabled | disabled}**—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**—Your description of the interface.
- **duplex**—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**—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**—The first VLAN in the inline VLAN pair.
  - **vlan2**—The second VLAN in the inline VLAN pair.

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.

```
sensor(config-int)# no inline-interfaces interface_name
```

You must also delete the inline interface from the virtual sensor to which it is assigned.

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.

```
sensor(config-int-phy)# admin-state enabled
```

You must assign the interface to a virtual sensor and enable it before it can monitor traffic.

Step 8 Add a description of this interface.

```
sensor(config-int-phy)# description INT1
```

Step 9 Configure the duplex settings.

```
sensor(config-int-phy)# duplex full
```

This option is not available on modules.

Step 10 Configure the speed.

```
sensor(config-int-phy)# speed 1000
```

This option is not available on modules.

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. To 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. To delete all VLAN pairs:

```
sensor(config-int-phy-inl-sub)# exit
sensor(config-int-phy-inl)# exit
sensor(config-int-phy)# subinterface-type none
```

You must also delete the interface from the virtual sensor to which it is assigned.

Step 15 Exit interface submode.

```
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 6-3.

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

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 is 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. The IDSM2 can read the 802.1q headers for all nonnative traffic to determine the VLAN ID for that packet. However, the IDSM2 does not know which VLAN is configured as the native VLAN for the port in the switch configuration, so it does not know what VLAN the native packets are in. Therefore, you must tell the IDSM2 which VLAN is the native VLAN for that port. Then the IDSM2 treats any untagged packets as if they were tagged with the native VLAN ID.

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. The IDSM2 also operates in this manner, because its two data ports are always connected to the same switch.

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. The second variation does not apply to the IDSM2 because it cannot be connected in this way.

For More Information

For more information on configuring the IDSM2 in VLAN groups, see Chapter 19, "Configuring the IDSM2."

Configuring VLAN Groups

Note For information on what you need to configure if you are using the hardware bypass card on the IPS 4260 and the IPS 4270-20, see Hardware Bypass Configuration Restrictions, page 5-9.

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 options apply:

- **admin-state {enabled | disabled}**—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**—Your description of the interface.
- **duplex**—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**—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**—No subinterfaces defined.
- **subinterface** *name*—Defines the subinterface as a VLAN group.
  - **vlans [range | unassigned]**—The set of VLANs in the VLAN group

  Note 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.

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.

```
sensor(config-int-phy)# admin-state enabled
```

You must also assign the interface to a virtual sensor and enable it before it can monitor traffic.

Step 8 Add a description of this interface.

```
sensor(config-int-phy)# description INT1
```

Step 9 Specify the duplex settings.

```
sensor(config-int-phy)# duplex full
```

This option is not available on modules.

Step 10 Specify the speed.

```
sensor(config-int-phy)# speed 1000
```

This option is not available on modules.

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. For 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. For 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 To delete VLAN groups:

a. To 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 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 group.

b. To delete all VLAN groups:

```
sensor(config-int-phy-vla-sub)# exit
sensor(config-int-phy-vla)# exit
sensor(config-int-phy)# subinterface-type none
```

You must also delete the VLAN group from the virtual sensor to which it is assigned.

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 6-3.

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

Note For information on what you need to configure if you are using the hardware bypass card on the IPS 4260 and the IPS 4270-20, see Hardware Bypass Configuration Restrictions, page 5-9.

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, 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.

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.

Note 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.

Configuring Inline Bypass Mode

Use the **bypass-mode** command in the service interface submode to configure bypass mode.

The following options apply:

- **off**—Turns off inline bypassing. Packet inspection is performed on inline data traffic. However, inline traffic is interrupted if Analysis Engine is stopped.
- **on**—Turns on inline bypassing. No packet inspection is performed on the traffic. Inline traffic continues to flow even if Analysis Engine is stopped.
- **auto**—Automatically begins bypassing inline packet inspection if Analysis Engine stops processing packets. This prevents data interruption on inline interfaces. This is the default.

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 information on the adaptive security appliance, bypass mode, and the AIP SSM, Adaptive Security Appliance, the AIP SSM, and Bypass Mode, page 18-12

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 options apply:

- **default**—Sets the value back to the system default setting.
- **idle-interface-delay**—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**—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**—Interval to check for missed packet percentage. The valid range is 5 to 3600. The default is 30 seconds

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.

Displaying Interface Statistics

Use the **show interfaces** [**clear** | **brief**] command in EXEC mode to display statistics for all system interfaces. Use the **show interfaces** [**FastEthernet** | **GigabitEthernet** | **Management**] [*slot/port*] command to display statistics for specific interfaces.

The following options 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.

- *slot/port*—Displays statistics for the specific slot/port of the interface.

Note For information on slot and port numbers and which platforms have a Management port, refer to *Installing Cisco Intrusion Prevention System Appliances and Modules 6.1*.

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.

```
sensor# show interfaces brief
CC   Interface            Sensing State   Link   Inline Mode   Pair Status
     GigabitEthernet0/0   Disabled        Down   Unpaired      N/A
*    Management0/0        Disabled        Up
     GigabitEthernet0/1   Disabled        Down   Unpaired      N/A
     GigabitEthernet0/2   Disabled        Down   Unpaired      N/A
     GigabitEthernet0/3   Disabled        Down   Unpaired      N/A
sensor#
```

The * indicates that the interface is the command and control interface.

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 To 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--
```

CHAPTER 6

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:

Understanding Analysis Engine

Analysis Engine performs packet analysis and alert detection. It monitors traffic that flows through specified interfaces.

You create virtual sensors in 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 Cisco IPS 6.1 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:

- IPS 4240
- IPS 4255
- IPS 4260

- IPS 4270-20
- AIP SSM

IDSM2 supports virtualization with the exception of VLAN groups on inline interface pairs. The AIM IPS and NME IPS do not support virtualization.

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.

For More Information

- For more information on the modify packet inline event action, see Event Actions, page 7-8.
- For more information on the Normalizer engine, see Normalizer Engine, page B-22.

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

Note You can create four virtual sensors.

Use the **virtual-sensor** *name* command in service analysis engine submode to create a virtual sensor. 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.

The following options apply:

- **anomaly-detection**—Anomaly detection parameters.
  - **anomaly-detection-name** *name*—Name of the anomaly detection policy.
  - **operational-mode {inactive**, **learn**, **detect}**—Anomaly detection modes.
- **description**—Description of the virtual sensor.
- **event-action-rules**—Name of the event action rules policy.
- **inline-TCP-evasion-protection-mode**—Lets you choose which type of Normalizer mode you need for traffic inspection:
  - **asymmetric** —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**—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.

- **inline-TCP-session-tracking-mode**—Advanced method by which to identify duplicate TCP session in inline traffic. The default is virtual sensor, which is almost always the best choice.
  - **virtual-sensor** —All packets with the same session key (AaBb) within a virtual sensor belong to the same session.
  - **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 or interfaces are tracked independently.
  - **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 independently.
- **signature-definition**—Name of the signature definition policy.

- **logical-interfaces**—Name of the logical interfaces (inline interface pairs).
- **physical-interfaces**—Name of the physical interfaces (promiscuous, inline VLAN pairs, and VLAN groups).
  - **subinterface-number**—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.

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 Assign the inline TCP session tracking mode.

```
sensor(config-ana-vir)# inline-TCP-session-tracking-mode virtual-sensor
```

The default is virtual sensor mode, which is almost always the best option to choose.

Step 9 Assign the inline TCP evasion protection mode.

```
sensor(config-ana-vir)# inline-TCP-evasion-protection-mode strict
```

The default is strict mode, which is almost always the best option to choose.

Step 10 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 11 Assign the promiscuous mode interfaces you want to add to this virtual sensor.

```
sensor(config-ana-vir)# physical-interface GigabitEthernet0/3
```

Repeat Step 11 for all the promiscuous interfaces that you want to assign to this virtual sensor.

Step 12 Assign the inline interface pairs you want to add to this virtual sensor.

```
sensor(config-ana-vir)# logical-interface inline_interface_pair_name
```

You must have already paired the interfaces.

Step 13 Assign the subinterfaces of the inline VLAN pairs or groups you want to add to this virtual sensor:

```
sensor(config-ana-vir)# physical-interface GigabitEthernet2/0 subinterface-number
subinterface_number
```

You must have already subdivided any interfaces into VLAN pairs or groups.

Step 14 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 15 Exit analysis engine mode.

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

Step 16 Press **Enter** to apply the changes or enter **no** to discard them.

For More Information

- For the procedure for creating virtual sensors on the AIP SSM, see Creating Virtual Sensors, page 18-3.
- 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 7-11.
- For more information on creating and configuring signature definition policies, see Working With Signature Definition Policies, page 8-1.
- For the procedure for pairing inline interfaces, see Configuring Inline Interface Pairs, page 5-16. 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 5-21 and Configuring VLAN Groups, page 5-27. Repeat Step 12 for all inline VLAN pairs or VLAN groups that you want to assign to this virtual sensor.

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
- Anomaly detection operational mode
- Inline TCP session tracking mode
- Description
- Interfaces assigned

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.

```
sensor(config-ana-vir)# inline-TCP-session-tracking-mode interface-and-vlan
```

The default is virtual sensor mode, which is almost always the best option to choose.

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.

```
sensor(config-ana-vir)# logical-interface inline_interface_pair_name
```

You must have already paired the interfaces.

Step 12 Change the subinterface with the inline VLAN pairs or groups assigned to this virtual sensor:

```
sensor(config-ana-vir)# physical-interface GigabitEthernet2/0 subinterface-number
subinterface_number
```

You must have already subdivided any interfaces into VLAN pairs or groups.

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 To delete a virtual sensor:

```
sensor(config-ana-vir)# exit
sensor(config-ana)# no virtual-sensor vs1
```

Step 15 Verify the deleted virtual sensor.

```
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)#
```

Only the default virtual sensor, vs0, is present.

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 7-11.
- For more information on creating and configuring signature definition policies, see Working With Signature Definition Policies, page 8-1.
- For the procedure for pairing inline interfaces, see Configuring Inline Interface Pairs, page 5-16. 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 5-21 and Configuring VLAN Groups, page 5-27. Repeat Step 12 for all inline VLAN pairs or VLAN groups that you want to assign to this virtual sensor.

Configuring Global Variables

Note Configuring the maximum number of open IP log files is the only global variable in Cisco IPS 6.1.

Use the **global-parameters** command in service analysis engine submode to create global variables.

The following options apply:

- **ip-logging**—Global IP logging parameters.
  - **max-open-iplog-files**—The maximum number of concurrently open log files. The range is 20 to 100. The default is 20.

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
```

Step 4 Verify the global variable settings:

```
sensor(config-ana-glo-ip)# show settings
   ip-logging
   ----------------------------------------------
      max-open-iplog-files: 50 default: 20
   ----------------------------------------------
sensor(config-ana-glo-ip)#
```

Step 5 Exit analysis engine mode.

```
sensor(config-ana-glo-ip)# exit
sensor(config-ana-glo)# exit
sensor(config-ana)# exit
sensor(config)#
Apply Changes:?[yes]:
```

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

CHAPTER 7

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:

Understanding Security Policies

Note You cannot create event action rules policies for the AIM IPS and NME IPS.

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 6.1 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.

Event Action Rules Components

This section describes the various components of event action rules, and contains the following topics:

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

Calculating the Risk Rating

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 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 (attack severity rating and signature fidelity rating) and on a per-server basis (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, 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 OS.

  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.

- 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 7-1 illustrates the risk rating formula:

Figure 7-1 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

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 SNMP Trap to 85-100. If you do not want to use action overrides, you can disable the entire event action override component.

Understanding Event Action Filters

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.

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.

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. The 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:

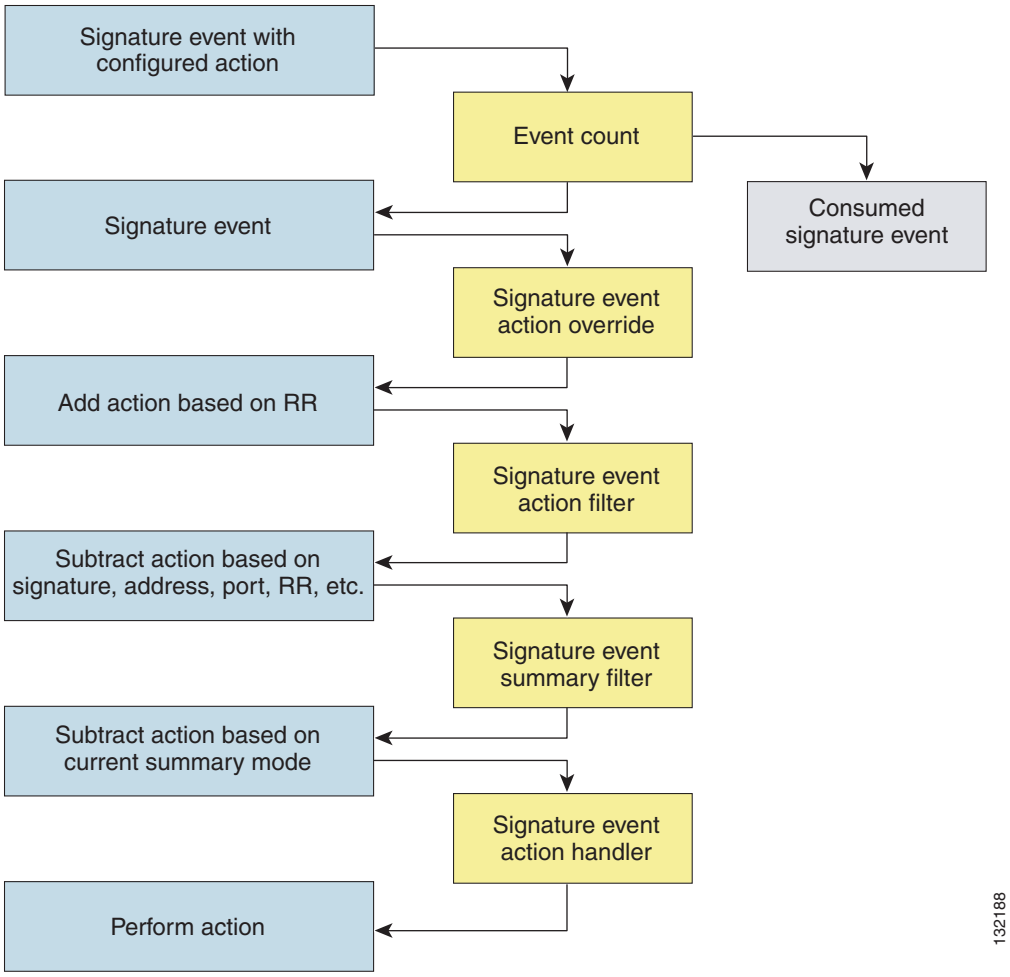
  - Signature ID
  - Subsignature ID
  - Attacker address
  - Attacker port
  - Victim address
  - Victim port
  - Risk rating threshold range
  - Actions to subtract
  - Sequence identifier (optional)
  - Stop-or-continue bit
  - Enable action filter line bit
  - Victim OS relevance or OS relevance

- 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 7-2 on page 7-7 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 7-2 Signature Event Through Signature Event Action Processor



For More Information

For more information on risk rating, see Calculating the Risk Rating, page 7-2.

Event Actions

Cisco IPS 6.1 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.

  - 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— Modifies packet data to remove ambiguity about what the end point might do with the packet.

  Note You cannot use modify-packet-inline for Add Event Action Filter or Add Event Action Override.

- Other Actions
  - request-block-connection—Sends a **r**equest to ARC to block this connection. You must have blocking devices configured to implement this action.
  - request-block-host—Sends a request to ARC to block this attacker host. You must have blocking devices configured to implement this action.
  - 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 action request-rate-limit applies to a select set of signatures.

  - reset-tcp-connection—Sends TCP resets to hijack and terminate the TCP flow. reset-tcp-connection 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 AIP SSM

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 AIP SSM, 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.

For More Information

- For procedure for configuring denied attackers, see Monitoring and Clearing the Denied Attackers List, page 7-33.
- For the procedure for configuring the general settings, see Configuring the General Settings, page 7-30.
- For the procedures for configuring blocking devices, see Chapter 13, “Configuring Attack Response Controller for Blocking and Rate Limiting.”
- For the procedures for configuring SNMP, see Chapter 14, “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 attack relevance rating in the calculation of 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.

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 To 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 To 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 To 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 Configuring Event Action Variables, page 7-12.
- For the procedure for configuring event action rules overrides, see Configuring Event Action Overrides, page 7-16.
- For the procedure for configuring event action rules filters, see Configuring Event Action Filters, page 7-18.
- For the procedure for configuring the general settings, see Configuring the General Settings, page 7-30.
- For the procedure for configuring event action rules target value ratings, see Configuring Target Value Ratings, page 7-14.
- For the procedure for configuring OS maps, see Configuring OS Identifications, page 7-23.

Configuring Event Action Variables

This section describes event action variables, and contains the following topics:

Understanding Event Action Variables

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 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.

When configuring IP addresses, specify the full IP address or ranges or set of ranges. For example:

- 10.89.10.10-10.89.10.23
- 10.90.1.1
- 192.56.10.1-192.56.10.255
- 10.1.1.1-10.2.255.255, 10.89.10.10-10.89.10.23

Timesaver For example, 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

Use the **variables** *variable_name* **address** *ip_address* command in service event action rules submode to create an event action variable. The IP address can be one address, a range, or ranges separated by a comma. Use the **no variables** *variable_name* command in service event action rules submode to delete an event action variable.

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 event action rules variable.

```
sensor(config-eve)# variables variable1 address 10.89.130.108
```

The valid values for **address** are A.B.C.D-A.B.C.D [,A.B.C.D-A.B.C.D].

Step 4 Verify the event action rules variable you added.

```
sensor(config-eve)# show settings
   variables (min: 0, max: 256, current: 2)
   -----------------------------------------------
      variableName: variable1
      -----------------------------------------------
         address: 10.89.130.108 default: 0.0.0.0-255.255.255.255
      -----------------------------------------------
   -----------------------------------------------
```

Step 5 To edit an event action rules variable, change the IP address.

```
sensor(config-eve)# variables variable1 address 10.89.130.191
```

Step 6 Verify the event action rules variable you edited.

```
sensor(config-eve)# show settings
   variables (min: 0, max: 256, current: 2)
   ----------------------------------------------
      variableName: variable1
      ---------------------------------------------
         address: 10.89.130.191 default: 0.0.0.0-255.255.255.255
      ---------------------------------------------
   ----------------------------------------------
```

Step 7 To delete an event action rules variable:

```
sensor(config-eve)# no variables variable1
```

Step 8 Verify the event action rules variable you deleted:

```
sensor(config-eve)# show settings
   variables (min: 0, max: 256, current: 0)
   ----------------------------------------------
   ----------------------------------------------
```

Step 9 Exit event action rules submode.

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

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

Configuring Target Value Ratings

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.

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.

The following options apply:

- Target value rating setting:
  - **zerovalue**—No value of this target.
  - **low**—Lower value of this target.
  - **medium**—Normal value of this target.
  - **high**—Elevated value of this target.
  - **mission-critical**—Extreme value of this target.
- **target-address** *ip_address*—Range set of IP address(es).

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 the target value rating to the network asset.

```
sensor(config-eve)# target-value mission-critical target-address 10.89.130.108
```

Step 4 Verify the target value rating you added.

```
sensor(config-eve)# show settings
----------------------------------------------
   target-value (min: 0, max: 5, current: 1)
   ----------------------------------------------
      target-value-setting: mission-critical
      target-address: 10.89.130.108 default: 0.0.0.0-255.255.255.255
   ----------------------------------------------
sensor(config-eve)#
```

Step 5 To edit a target value rating, change the target value rating setting of the asset.

```
sensor(config-eve)# target-value low target-address 10.89.130.108
```

Step 6 Verify the target value rating you edited.

```
sensor(config-eve)# show settings
----------------------------------------------
 target-value (min: 0, max: 5, current: 1)
 ----------------------------------------------
    target-value-setting: low
    target-address: 10.89.130.108 default: 0.0.0.0-255.255.255.255
 ----------------------------------------------
```

Step 7 To delete the target value rating:

```
sensor(config-eve)# no target-value low
```

Step 8 Verify the target value rating you deleted.

```
sensor(config-eve)# show settings
----------------------------------------------
 target-value (min: 0, max: 5, current: 0)
 ----------------------------------------------
 ----------------------------------------------
```

Step 9 Exit event action rules submode.

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

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

Configuring 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-snmp-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.

The following options apply:

- **no**—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**—Range of risk rating values for this override item. The default is 0 to 100.
- **show**—Displays system settings and/or history information.

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:

- To deny packets from the source IP address of the attacker:

```
sensor(config-eve)# overrides deny-attacker-inline
sensor(config-eve-ove)#
```

- To not transmit the single packet causing the alert:

```
sensor(config-eve)# overrides deny-packet-inline
sensor(config-eve-ove)#
```

- To not transmit packets on the specified TCP connection:

```
sensor(config-eve)# overrides deny-connection-inline
sensor(config-eve-ove)#
```

- To send TCP RST packets to terminate the connection:

```
sensor(config-eve)# overrides reset-tcp-connection
sensor(config-eve-ove)#
```

- To request a block of the connection:

```
sensor(config-eve)# overrides request-block-connection
sensor(config-eve-ove)#
```

- To request a block of the attacker host:

```
sensor(config-eve)# overrides request-block-host
sensor(config-eve-ove)#
```

- To log the packets from the attacker IP address:

```
sensor(config-eve)# overrides log-attacker-packets
sensor(config-eve-ove)#
```

- To log the packets from the victim IP address:

```
sensor(config-eve)# overrides log-victim-packets
sensor(config-eve-ove)#
```

- To log packets from both the attacker and victim IP addresses:

```
sensor(config-eve)# overrides log-pair-packets
sensor(config-eve-ove)#
```

- To write an alert to Event Store:

```
sensor(config-eve)# overrides produce-alert
sensor(config-eve-ove)#
```

- To write verbose alerts to Event Store:

```
sensor(config-eve)# overrides produce-verbose-alert
sensor(config-eve-ove)#
```

- To 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.

```
sensor(config-eve-ove)# risk-rating-range 85-100
```

Note The default risk rating range is 0 to 100. Set it to a different value, such as 85 to 100.

Step 5 To enable or disable the use of this override item:

```
sensor(config-eve-ove)# override-item-status [enabled | disabled]
```

The default is enabled.

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 the event action override that you edited.

```
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)#
```

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.

Step 10 Verify the event action override that you deleted.

```
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 description of all the event actions, see Event Actions, page 7-8.

Configuring Event Action Filters

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 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.

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.

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 options apply:

- **actions-to-remove**—Event actions to remove for this filter item.
- **attacker-address-range**—Range set of attacker address(es) for this item (for example, 10.20.1.0-10.20.1.255,10.20.5.0-10.20.5.255).
- **attacker-port-range**—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**—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.
- **no**—Removes an entry or selection setting.
- **os-relevance**—Event OS relevance for this filter.
  - **relevant**—The event is relevant to the target OS.
  - **not-relevant**—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**—Range of risk rating values for this filter item.
- **signature-id-range**—Range set of signature ID(s) for this item (for example, 1000-2000,3000-3000).
- **stop-on-match {true | false}**—Continues evaluating filters or stops when this filter item is matched.
- **subsignature-id-range**—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**—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).
- **victim-port-range**—Range set of victim port(s) for this item (for example, 147-147,8000-10000).

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.

```
sensor(config-eve)# filters insert name1 begin
```

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.

Step 4 Specify the values for this filter:

a. Specify the signature ID range.

```
sensor(config-eve-fil)# signature-id-range 1000-1005
```

The default is 900 to 65535.

b. Specify the subsignature ID range.

```
sensor(config-eve-fil)# subsignature-id-range 1-5
```

The default is 0 to 255.

c. Specify the attacker address range.

```
sensor(config-eve-fil)# attacker-address-range 10.89.10.10-10.89.10.23
```

The default is 0.0.0.0 to 255.255.255.255.

d. Specify the victim address range.

```
sensor(config-eve-fil)# victim-address-range 192.56.10.1-192.56.10.255
```

The default is 0.0.0.0 to 255.255.255.255.

e. Specify the victim port range.

```
sensor(config-eve-fil)# victim-port-range 0-434
```

The default is 0 to 65535.

f. Specify the OS relevance.

```
sensor(config-eve-fil)# os-relevance relevant
```

The default is 0 to 100.

g. Specify the risk rating range.

```
sensor(config-eve-fil)# risk-rating-range 85-100
```

The default is 0 to 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.

```
sensor(config-eve-fil)# deny-attacker-percentage 90
```

The default is 100.

j. Specify the status of the filter to either disabled or enabled.

```
sensor(config-eve-fil)# filter-item-status {enabled | disabled}
```

The default is enabled.

k. Specify the stop on match parameter.

```
sensor(config-eve-fil)# stop-on-match [{rue | false}
```

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.

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: 10.89.10.10-10.89.10.23 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
      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
   ----------------------------------------------
senor(config-eve-fil)#
```

Step 6 To edit an existing filter:

```
sensor(config-eve)# filters edit name1
```

Step 7 Edit the parameters (see Steps 4a through 4l).

Step 8 To 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 To 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 detailed information about event actions, see Event Actions, page 7-8.

For More Information

- For the procedure for configuring event action variables, see Adding, Editing, and Deleting Event Action Variables, page 7-13.
- For a description of all the event actions, see Event Actions, page 7-8.

Configuring OS Identifications

This section describes OS identifications and how to configure OS maps, and contains the following topics:

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 mappings, which take precedence over learned OS mappings.

- 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 mappings—OS mappings you enter.

   Configured OS mappings reside in the Event Action Rules policy and can apply to one or many virtual sensors.

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.

2. Imported OS mappings—OS mappings imported from an external data source.

   Imported OS mappings are global and apply to all virtual sensors.

   Note Currently CSA MC is the only external data source.

3. Learned OS mappings—OS mappings observed by the sensor through the fingerprinting of TCP packets with the SYN control bit set.

   Learned OS mappings 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 mappings. If the target IP address is not in the configured OS mappings, the sensor looks in the imported OS mappings. If the target IP address is not in the imported OS mappings, the sensor looks in the learned OS mappings. 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 mappings

  We recommend configuring OS mappings to define the identity of the OS running on critical systems. It is best to configure OS mappings when the OS and IP address of the critical systems are unlikely to change.

- Limit 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 mappings

  Importing OS mappings 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 relevancy value of the target

  This provides a way to filter alerts solely on OS relevancy.

- Disable passive analysis

  Stops the sensor from learning new OS mappings.

- 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 7-1):

Table 7-1 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 options apply:

- **calc-arr-for-ip-range**—Calculate 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}**—Collection of administrator-defined mappings of IP addresses to OS IDs (configured OS mappings take precedence over imported and learned OS mappings).
- **ip**—The host IP address (or addresses) running the specified OS. he 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**—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.

To configure OS mapping, 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.

```
sensor(config-eve)# os-identification
sensor(config-eve-os)# configured-os-map insert name1 begin
sensor(config-eve-os-con)#
```

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.

Step 4 Specify the values for this OS map:

a. Specify the host IP address.

```
sensor(config-eve-os-con)# ip 10.20.1.0-10.20.1.255
```

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: 10.20.1.0-10.20.1.255 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 10.89.30.108-10.89.30.191
```

Step 7 Enable passive OS fingerprinting.

```
sensor(config-eve-os)# passive-traffic-analysis enabled
```

Step 8 To 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 To 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: 10.89.30.79 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: 10.89.30.79 default:
            os: aix
         ----------------------------------------------
         ----------------------------------------------
         NAME: name1
         ----------------------------------------------
            ip: 10.20.1.0-10.20.1.255 default:
            os: unix
         ----------------------------------------------
         ----------------------------------------------
      ----------------------------------------------
      passive-traffic-analysis: Enabled default: Enabled
   ----------------------------------------------
sensor(config-eve-os)#
```

Step 12 To 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: 10.89.30.79 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: 10.89.30.79 default:
            os: aix
         ----------------------------------------------
         ----------------------------------------------
      ----------------------------------------------
      INACTIVE list-contents
      ----------------------------------------------
         NAME: name1
         ----------------------------------------------
            ip: 10.20.1.0-10.20.1.255 default:
            os: unix
         ----------------------------------------------
         ----------------------------------------------
      ----------------------------------------------
      passive-traffic-analysis: Enabled default: Enabled
--MORE--#
```

Step 14 To delete an OS map:

```
sensor(config-eve-os)# no configured-os-map name2
```

Step 15 Verify the OS map has been deleted.

```
sensor(config-eve-os)# show settings
   os-identification
   ----------------------------------------------
      calc-arr-for-ip-range: 10.89.30.79 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: 10.20.1.0-10.20.1.255 default:
            os: unix
         ----------------------------------------------
         ----------------------------------------------
      ----------------------------------------------
      passive-traffic-analysis: Enabled default: Enabled
   ----------------------------------------------
sensor(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 options apply:

- *virtual-sensor*—(Optional) The learned addresses of the virtual sensor that should be displayed or cleared.
- *ip-address*—(Optional) The IP address to query or clear. The sensor displays or clears the OS ID mapped to the specified IP address.

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 10.1.1.12
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 10.1.1.12
```

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 the General Settings

Use the following commands in service event action rules submode to configure general event action rules settings:

- **global-block-timeout** —Number of minutes to block a host or connection. The valid range is 0 to 10000000. The default is 30 minutes.
- **global-deny-timeout**—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.

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 To enable or disable the meta event generator:

```
sensor(config-eve-gen)# global-metaevent-status [enabled | disabled]
```

The default is enabled.

Step 5 To enable or disable the summarizer:

```
sensor(config-eve-gen)# global-summarization-status [enabled | disabled]
```

The default is enabled.

Step 6 To configure the denied attackers inline event action:

a. To limit the number of denied attackers in the system at any given time:

```
sensor(config-eve-gen)# max-denied-attackers 100
```

The default is 1000.

b. To configure the amount of seconds to deny attackers in the system:

```
sensor(config-eve-gen)# global-deny-timeout 1000
```

The default is 3600 seconds.

Step 7 To configure the number of minutes to block a host or a connection:

```
sensor(config-eve-gen)# global-block-timeout 20
```

The default is 30 minutes.

Step 8 To enable or disable any overrides that you have set up:

```
sensor(config-eve-gen)# global-overrides-status [enabled | disabled]
```

The default is enabled.

Step 9 To enable or disable any filters that you have set up:

```
sensor(config-eve-gen)# global-filters-status [enabled | disabled]
```

The default is enabled.

Step 10 Check 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 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

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 options apply:

- *name*—(Optional) The name of the virtual sensor to which the deny attackers entry should be added.
- *attacker-ip-address*—The attacker IP address.
- *victim-ip-address*—(Optional) The victim IP address.
- *port-number*—(Optional) The victim port number. The valid range is 0 to 65535.

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 10.1.1.1.

```
sensor# deny attacker ip-address 10.1.1.1
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 To add a deny attacker entry to a specific virtual sensor:

```
sensor# deny attacker virtual-sensor vs0 ip-address 10.1.1.1
```

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 7-33.

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 options apply:

- *virtual_sensor*—(Optional) The virtual sensor whose denied attackers list should be cleared.
- *ip_address*—(Optional) The IP address to clear.

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.

```
sensor# show statistics denied-attackers
Denied Attackers and hit count for each.
   10.20.4.2 = 9
   10.20.5.2 = 5
```

The statistics show that there are two IP addresses being denied at this time.

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 10.1.1.1
Warning: Executing this command will delete ip address 10.1.1.1 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 To clear only the statistics:

```
sensor# show statistics virtual-sensor clear
```

Step 11 Verify that you have cleared the statistics.

```
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
```

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.

Monitoring Events

This section describes how to display and clear events from Event Store, and contains the following topics:

Displaying Events

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.

Note Events are displayed as a live feed. To cancel the request, press **Ctrl-C.**

The following options 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 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**—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 ARC (block) requests.

  Note ARC is formerly known as NAC. This name change has not been completely implemented throughout IDM, IME, and the CLI for Cisco IPS 6.1.
- **status**—Displays status events.
- **past**—Displays events starting in the past for the specified hours, minutes, and seconds.
- *hh:mm:ss*—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**.

To display events from Event Store, follow these steps:

Step 1 Log in to the CLI.

Step 2 Display all events starting now.

```
sensor# show events
evError: eventId=1041472274774840147 severity=warning vendor=Cisco
  originator:
    hostId: sensor2
    appName: cidwebserver
    appInstanceId: 12075
  time: 2008/01/07 04:41:45 2008/01/07 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: 2008/01/07 04:41:45 2008/01/07 04:41:45 UTC
  errorMessage: name=errTransport WebSession::sessionTask(6) TLS connection exce
ption: handshake incomplete.
```

The feed continues showing all events until you press **Ctrl-C**.

Step 3 Display the block requests beginning at 10:00 a.m. on February 9, 2008.

```
sensor# show events NAC 10:00:00 Feb 9 2008
evShunRqst: eventId=1106837332219222281 vendor=Cisco
  originator:
    deviceName: Sensor1
    appName: NetworkAccessControllerApp
    appInstance: 654
  time: 2008/02/09 10:33:31 2008/08/09 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, 2008.

```
sensor# show events error warning 10:00:00 Feb 9 2008
evError: eventId=1041472274774840197 severity=warning vendor=Cisco
  originator:
    hostId: sensor
    appName: cidwebserver
    appInstanceId: 12160
  time: 2008/01/07 04:49:25 2008/01/07 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: 2008/03/02 14:15:59 2008/03/02 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: 2008/01/08 02:41:00 2008/01/08 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: 2008/01/08 02:41:00 2008/01/08 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 Event Store.

To clear events from Event Store, follow these steps:

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

Step 2 Clear 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.

Generating Status Events from Health Monitoring Control Transactions

Sensor health monitoring generates control transactions every 5 seconds or so. By default, the status events generated from control transactions are filtered to reduce the large amount of status events. You can have status events generated from control transaction again by using **status-event-logging-categories** command.

To reenable status events generated from health monitoring control transactions, follow these steps:

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

Step 2 Enter service logger submode.

```
sensor(config)# service logger
sensor(config-log)#
```

Step 3 Enter event store submode.

```
sensor(config-log)# event-store
sensor(config-log-eve)#
```

Step 4 Enable event generation from control transactions.

```
sensor(config-log-eve)# status-event-logging-categories controlTransaction enabled true
sensor(config-log-eve)#
```

Step 5 Exit event store submode.

```
sensor(config-log-eve)# exit
sensor(config-log)#
```

Step 6 Exit service logger submode.

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

Step 7 Press **Enter** to apply the changes or enter **no** to discard them.

CHAPTER 8

Defining Signatures

This chapter describes how to define and create signatures. It contains the following sections:

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 6.1 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.

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 To display a list of signature definition policies on the sensor:

```
sensor# list signature-definition-configurations
Signature Definition
   Instance   Size   Virtual Sensor
   sig0       255    vs0
   temp       707    N/A
   MySig      255    N/A
   sig1       141    vs1
sensor#
```

Step 6 To 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
   sig0       255    vs0
   temp       707    N/A
   sig1       141    vs1
sensor#
```

Step 8 To 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 8-4.
- For the procedure for configuring the general settings, see Configuring Signatures, page 8-5.

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.

Cisco IPS 6.1 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 Signature Variables

This section describes signature variables, and contains the following topics:

- Understanding Signature Variables, page 8-4
- Adding, Editing, and Deleting Signature Variables, page 8-4

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 the 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.

Adding, Editing, and Deleting Signature Variables

Use the **variables** command in the signature definition submode to create signature variables.

The following options 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**—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**—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.

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.

```
sensor(config-sig)# variables WEBPORTS web-ports 80,3128,8000
```

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.

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 To 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 submode.

```
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

The following options 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 submode.
- **event-counter**—Sets the event count.
- **promisc-delta**—The delta value used to determine the seriousness of the alert.

Caution We do not recommend that you 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**—Rating of the fidelity of signature.
- **status**—Sets the status of the signature to enabled or retired.
- **vulnerable-os**—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 8-7.
- For more information about signature engines, see Appendix B, “Signature Engines.”
- For the procedure for assigning actions, see Assigning Actions to Signatures, page 8-15.
- For the procedure for configuring event counts, see Configuring Event Counter, page 8-10.

- For the procedure for configuring the signature fidelity rating, see Configuring Signature Fidelity Rating, page 8-11.
- For the procedure for enabling and disabling signatures, see Configuring the Status of Signatures, page 8-12.
- For the procedure for configuring vulnerable OSes, see Configuring the Vulnerable OSes for a Signature, page 8-13.

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 options 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**—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.

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 options 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.

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 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 options 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 signatures.
  - **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 mode:
  - **alert-interval**—Specifies the time in seconds before the event count is reset. The default is 60.

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.

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 options 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.

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 options apply:

- **general-os**—All OS types
- **ios**—Variants of Cisco IOS
- **mac-os**—Variants of Macintosh OS
- **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 Microsoft NT, 2000, and XP
- **win-nt**—Specific variants of Windows NT

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 options apply:

- **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. This action causes an alert to be written to Event Store, even if **produce-alert** is not selected.
- **log-pair-packets**—Starts IP logging of packets containing the attacker-victim address pair. This action causes an alert to be written to Event Store, even if **produce-alert** is not selected.
- **log-victim-packets**—Starts IP logging of packets containing the victim address. This action causes an alert to be written to Event Store, even if **produce-alert** is not selected.
- **produce-alert** —Writes the event to Event Store as an alert.
- **produce-verbose-alert**—Includes an encoded dump (possibly truncated) 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.
- **request-block-connection**—Sends a request to ARC to block this connection. You must have blocking devices configured to implement this action.
- **request-block-host**—Sends a request to ARC to block this attacker host. You must have blocking devices configured to implement this action.
- **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.
- **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.
- **reset-tcp-connection**—Sends TCP resets to hijack and terminate the TCP flow. **Reset TCP Connection** only works on TCP signatures that analyze a single connection. It does not work for sweeps or floods.
- **modify-packet-inline**— Modifies packet data to remove ambiguity about what the end point might do with the packet.
- **event-action-settings**—Lets you set the **external-rate-limit-type**:
  - **none**—No rate limiting configured.
  - **percentage**—Sets the rate limit by traffic percentage (**external-rate-limit-percentage**).

Assigning Actions to Signatures

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 detailed information about event actions, see Event Actions, page 7-8.

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

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 engine runs 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 8-20, AIC MIME Define Content Type Signatures, page 8-21, AIC Transfer Encoding Signatures, page 8-24, and AIC FTP Commands Signatures, page 8-24.
- For the procedure for creating a custom MIME signature, see Creating an AIC Signature, page 8-26.

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 options 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.

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-htt)# 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-htt)# max-outstanding-http-requests-per-connection 5
```

d. Edit the AIC ports.

```
sensor(config-sig-app-htt)# aic-web-ports 80-80,3128-3128
```

Step 5 Verify your settings.

```
sensor(config-sig-app-htt)# 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 submode.

```
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 8-1 lists the predefined define request method signatures. Enable the signatures that have the predefined method you need.

Table 8-1 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
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 8-12.

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 8-2 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 8-2 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 12623 1 12623 2	Content Type image/tiff Header Check Content Type image/tiff Invalid Message Length Content Type image/tiff Verification Failed
12624 0 12624 1 12624 2	Content Type image/x-3ds Header Check Content Type image/x-3ds Invalid Message Length Content Type image/x-3ds Verification Failed
12626 0 12626 1 12626 2	Content Type image/x-portable-bitmap Header Check Content Type image/x-portable-bitmap Invalid Message Length Content Type image/x-portable-bitmap Verification Failed
12627 0 12627 1 12627 2	Content Type image/x-portable-graymap Header Check Content Type image/x-portable-graymap Invalid Message Length Content Type image/x-portable-graymap Verification Failed
12628 0 12628 1 12628 2	Content Type image/jpeg Header Check Content Type image/jpeg Invalid Message Length Content Type image/jpeg Verification Failed
12629 0 12629 1	Content Type image/cgf Header Check Content Type image/cgf Invalid Message Length
12631 0 12631 1	Content Type image/x-xpm Header Check Content Type image/x-xpm Invalid Message Length
12633 0 12633 1 12633 2	Content Type audio/midi Header Check Content Type audio/midi Invalid Message Length Content Type audio/midi Verification Failed
12634 0 12634 1 12634 2	Content Type audio/basic Header Check Content Type audio/basic Invalid Message Length Content Type audio/basic Verification Failed

Table 8-2 Define Content Type Signatures (continued)

Signature ID	Signature Description
12635 0 12635 1 12635 2	Content Type audio/mpeg Header Check Content Type audio/mpeg Invalid Message Length Content Type audio/mpeg Verification Failed
12636 0 12636 1 12636 2	Content Type audio/x-adpcm Header Check Content Type audio/x-adpcm Invalid Message Length Content Type audio/x-adpcm Verification Failed
12637 0 12637 1 12637 2	Content Type audio/x-aiff Header Check Content Type audio/x-aiff Invalid Message Length Content Type audio/x-aiff Verification Failed
12638 0 12638 1 12638 2	Content Type audio/x-ogg Header Check Content Type audio/x-ogg Invalid Message Length Content Type audio/x-ogg Verification Failed
12639 0 12639 1 12639 2	Content Type audio/x-wav Header Check Content Type audio/x-wav Invalid Message Length Content Type audio/x-wav Verification Failed
12641 0 12641 1 12641 2	Content Type text/html Header Check Content Type text/html Invalid Message Length Content Type text/html Verification Failed
12642 0 12642 1	Content Type text/css Header Check Content Type text/css Invalid Message Length
12643 0 12643 1	Content Type text/plain Header Check Content Type text/plain Invalid Message Length
12644 0 12644 1	Content Type text/richtext Header Check Content Type text/richtext Invalid Message Length
12645 0 12645 1 12645 2	Content Type text/sgml Header Check Content Type text/sgml Invalid Message Length Content Type text/sgml Verification Failed
12646 0 12646 1 12646 2	Content Type text/xml Header Check Content Type text/xml Invalid Message Length Content Type text/xml Verification Failed
12648 0 12648 1 12648 2	Content Type video/flc Header Check Content Type video/flc Invalid Message Length Content Type video/flc Verification Failed
12649 0 12649 1 12649 2	Content Type video/mpeg Header Check Content Type video/mpeg Invalid Message Length Content Type video/mpeg Verification Failed
12650 0 12650 1	Content Type text/xmcd Header Check Content Type text/xmcd Invalid Message Length
12651 0 12651 1 12651 2	Content Type video/quicktime Header Check Content Type video/quicktime Invalid Message Length Content Type video/quicktime Verification Failed
12652 0 12652 1	Content Type video/sgi Header Check Content Type video/sgi Verification Failed

Table 8-2 Define Content Type Signatures (continued)

Signature ID	Signature Description
12653 0 12653 1	Content Type video/x-avi Header Check Content Type video/x-avi Invalid Message Length
12654 0 12654 1 12654 2	Content Type video/x-fli Header Check Content Type video/x-fli Invalid Message Length Content Type video/x-fli Verification Failed
12655 0 12655 1 12655 2	Content Type video/x-mng Header Check Content Type video/x-mng Invalid Message Length Content Type video/x-mng Verification Failed
12656 0 12656 1 12656 2	Content Type application/x-msvideo Header Check Content Type application/x-msvideo Invalid Message Length Content Type application/x-msvideo Verification Failed
12658 0 12658 1	Content Type application/ms-word Header Check Content Type application/ms-word Invalid Message Length
12659 0 12659 1	Content Type application/octet-stream Header Check Content Type application/octet-stream Invalid Message Length
12660 0 12660 1 12660 2	Content Type application/postscript Header Check Content Type application/postscript Invalid Message Length Content Type application/postscript Verification Failed
12661 0 12661 1	Content Type application/vnd.ms-excel Header Check Content Type application/vnd.ms-excel Invalid Message Length
12662 0 12662 1	Content Type application/vnd.ms-powerpoint Header Check Content Type application/vnd.ms-powerpoint Invalid Message Length
12663 0 12663 1 12663 2	Content Type application/zip Header Check Content Type application/zip Invalid Message Length Content Type application/zip Verification Failed
12664 0 12664 1 12664 2	Content Type application/x-gzip Header Check Content Type application/x-gzip Invalid Message Length Content Type application/x-gzip Verification Failed
12665 0 12665 1	Content Type application/x-java-archive Header Check Content Type application/x-java-archive Invalid Message Length
12666 0 12666 1	Content Type application/x-java-vm Header Check Content Type application/x-java-vm Invalid Message Length
12667 0 12667 1 12667 2	Content Type application/pdf Header Check Content Type application/pdf Invalid Message Length Content Type application/pdf Verification Failed
12668 0 12668 1	Content Type unknown Header Check Content Type unknown Invalid Message Length
12669 0 12669 1	Content Type image/x-bitmap Header Check 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 8-12.
- For the procedure for creating an ACI signature, see Creating an AIC Signature, page 8-26.

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 8-3 lists the predefined transfer encoding signatures. Enable the signatures that have the predefined transfer encoding method you need.

Table 8-3 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 8-12.

AIC FTP Commands Signatures

Table 8-4 lists the predefined FTP commands signatures. Enable the signatures that have the predefined FTP command you need.

Table 8-4 FTP Commands Signatures

Signature ID	FTP Command
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

Table 8-4 FTP Commands Signatures (continued)

Signature ID	FTP Command
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
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 8-12.

Creating an AIC Signature

The following example demonstrates how to create a MIME-type signature based on the AIC engine.

The following options 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. This action causes an alert to be written to Event Store, even if **produce-alert** is not selected.
  - **log-pair-packets**—Starts IP logging of packets containing the attacker-victim address pair. This action causes an alert to be written to Event Store, even if **produce-alert** is not selected.
  - **log-victim-packets**—Starts IP logging of packets containing the victim address. This action causes an alert to be written to Event Store, even if **produce-alert** is not selected.
  - **produce-alert** —Writes the event to Event Store as an alert.
  - **produce-verbose-alert**—Includes an encoded dump (possibly truncated) 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.
  - **request-block-connection**—Sends a request to ARC to block this connection. You must have blocking devices configured to implement this action.
  - **request-block-host**—Sends a request to ARC to block this attacker host. You must have blocking devices configured to implement this action.
  - **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.
  - **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.
  - **reset-tcp-connection**—Sends TCP resets to hijack and terminate the TCP flow. **Reset TCP Connection** only works on TCP signatures that analyze a single connection. It does not work for sweeps or floods.
  - **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

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.

For More Information

For detailed information about event actions, see Event Actions, page 7-8.

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

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 8-5 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 8-5 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[1]
1201 Fragment Overlap	Fires when the fragments queued for a datagram overlap each other.	None[2]	
1202 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[3]
1203 Fragment Overwrite	Fires when the fragments queued for a datagram overlap each other and the overlapping data is different.[4]	None	Deny Packet Inline Produce Alert[5]
1204 No Initial Fragment	Fires when the datagram is incomplete and missing the initial fragment.	None	Deny Packet Inline Produce Alert[6]

Table 8-5 IP Fragment Reassembly Signatures (continued)

Signature ID and Name	Description	Parameter With Default Value and Range	Default Action
1205 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 Produce Alert[7]
1206 Fragment Too Small	Fires when there are more than Max Small Frags of a size less than Min Fragment Size in one datagram.[8]	Specify Max Small Frags 2 (8-1500) Specify Min Fragment Size 400 (1-8)	Deny Packet Inline Produce Alert[9]
1207 Too Many Fragments	Fires when there are more than Max Fragments per Datagram in one datagram.	Specify Max Fragments per Datagram 170 (0-8192)	Deny Packet Inline Produce Alert[10]
1208 Incomplete Datagram	Fires when all of the fragments for a datagram have not arrived during the Fragment Reassembly Timeout.[11]	Specify Fragment Reassembly Timeout 60 (0-360)	Deny Packet Inline Produce Alert[12]
1220 Jolt2 Fragment Reassembly DoS attack	Fires when multiple fragments are received all claiming to be the last fragment of an IP datagram.	Specify Max Last Fragments 4 (1-50)	Deny Packet Inline Produce Alert[13]
1225 Fragment Flags Invalid	Fires when a bad combination of fragment flags is detected.	None[14]	

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. 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.
11. The timer starts when the packet for the datagram arrives.
12. 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.
13. 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.
14. 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-22.

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 needed) 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 options apply:

- **ip-reassemble-mode**—Identifies the method the sensor uses to reassemble the fragments based on the operating system:
  - **nt**—Windows systems (default).
  - **solaris**—Solaris systems.
  - **linux**—GNU/Linux systems.
  - **bsd**—BSD UNIX systems.

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 submode.

```
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 submode.

```
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

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 8-6 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 8-6 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 8-6 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 recv 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 8-6 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 that 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 exceed TCP Max MSS	TCP Max MSS1460 (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 8-6 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	

1. 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-22.

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 submode.

```
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 submode.

```
sensor(config-sig-sig-nor)# edit-default-sigs-only default-signatures-only
```

Step 6 Enable and change the default setting (if needed) 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 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 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 options 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.

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 Editing and Deleting Virtual Sensors, page 6-7.

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 the signature definition submode to configure IP logging.

The following options 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.

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 submode.

```
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

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 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.

The following example demonstrates how to create a custom String TCP signature. This procedure also applies to String UDP and ICMP signatures.

The following options 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. This action causes an alert to be written to Event Store, even if **produce-alert** is not selected.
  - **log-pair-packets**—Starts IP logging of packets containing the attacker-victim address pair. This action causes an alert to be written to Event Store, even if **produce-alert** is not selected.
  - **log-victim-packets**—Starts IP logging of packets containing the victim address. This action causes an alert to be written to Event Store, even if **produce-alert** is not selected.
  - **produce-alert** —Writes the event to Event Store as an alert.
  - **produce-verbose-alert**—Includes an encoded dump (possibly truncated) 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.
  - **request-block-connection**—Sends a request to ARC to block this connection. You must have blocking devices configured to implement this action.
  - **request-block-host**—Sends a request to ARC to block this attacker host. You must have blocking devices configured to implement this action.
  - **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.

  - **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.
  - **reset-tcp-connection**—Sends TCP resets to hijack and terminate the TCP flow. **Reset TCP Connection** only works on TCP signatures that analyze a single connection. It does not work for sweeps or floods.
  - **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 the 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.
- **specify-min-match-length {yes | no}**—(Optional) Enables min-match-length.
- **strip-telnet-options**—Strips Telnet option characters from data before searching.
- **swap-attacker-victim {true | false}**—Swaps the attacker and victim addresses and ports (source and destination) in the alert message and for any actions taken. The default is false for no swapping.

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.

```
sensor(config-sig)# signatures 60025 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 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.

```
sensor(config-sig-sig-str)# regex-string This-is-my-new-Sig-regex
```

Step 11 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**.

Step 12 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 13 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 14 Exit signature definition submode.

```
sensor(config-sig-sig-str)# 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 detailed information about event actions, see Event Actions, page 7-8.

Example Service HTTP 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 to web services 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 options 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. This action causes an alert to be written to Event Store, even if **produce-alert** is not selected.
  - **log-pair-packets**—Starts IP logging of packets containing the attacker-victim address pair. This action causes an alert to be written to Event Store, even if **produce-alert** is not selected.
  - **log-victim-packets**—Starts IP logging of packets containing the victim address. This action causes an alert to be written to Event Store, even if **produce-alert** is not selected.
  - **produce-alert** —Writes the event to Event Store as an alert.

  - **produce-verbose-alert**—Includes an encoded dump (possibly truncated) 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.
  - **request-block-connection**—Sends a request to ARC to block this connection. You must have blocking devices configured to implement this action.
  - **request-block-host**—Sends a request to ARC to block this attacker host. You must have blocking devices configured to implement this action.
  - **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.
  - **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.
  - **reset-tcp-connection**—Sends TCP resets to hijack and terminate the TCP flow. **Reset TCP Connection** only works on TCP signatures that analyze a single connection. It does not work for sweeps or floods.
  - **modify-packet-inline**— Modifies packet data to remove ambiguity about what the end point might do with the packet.
- **max-field-sizes** —Eanbles grouping for maximum field sizes:
  - **specify-max-arg-field-length {yes | no}**—(Optional) Enables max-arg-field-length.
  - **specify-max-header-field-length {yes | no}**—(Optional) Enables max-header-field-length.
  - **specify-max-request-length {yes | no}**—(Optional) Enables max-request-length.
  - **specify-max-uri-field-length {yes | no}**—(Optional) Enables max-uri-field-length.
- **no**—Removes an entry or selection setting.
- **regex**—Enables regular expression grouping:
  - **specify-arg-name-regex**—(Optional) Enables arg-name-regex.
  - **specify-header-regex** — Optional) Enables header-regex.
  - **specify-request-regex**—(Optional) Enables request-regex.
  - **specify-uri-regex**— Optional) Enables uri-regex.
- **service-ports** —A comma-separated list of ports or port ranges where the target service may reside.
- **swap-attacker-victim {true | false}**—Swaps the attacker and victim addresses and ports (source and destination) in the alert message and for any actions taken. The default is false for no swapping.

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 submode.

```
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 submode.

```
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 submode.

```
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 submode.

```
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.

For More Information

For detailed information about event actions, see Event Actions, page 7-8.

Example Meta Signature

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.

Caution A large number of Meta signatures could adversely affect overall sensor performance.

Note The Meta engine is different from other engines in that it takes alerts as input where most engines take packets as input.

The following options apply:

- **component-list**—Specifies the Meta component:
  - **edit**—Edits an existing entry in the list.
  - **insert name1**—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.
- **component-list-in-order {true | false}**—Whether or not to have the component list fire in order.
- **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. This action causes an alert to be written to Event Store, even if **produce-alert** is not selected.
- **log-pair-packets**—Starts IP logging of packets containing the attacker-victim address pair. This action causes an alert to be written to Event Store, even if **produce-alert** is not selected.
- **log-victim-packets**—Starts IP logging of packets containing the victim address. This action causes an alert to be written to Event Store, even if **produce-alert** is not selected.
- **produce-alert** —Writes the event to Event Store as an alert.
- **produce-verbose-alert**—Includes an encoded dump (possibly truncated) 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.
- **request-block-connection**—Sends a request to ARC to block this connection. You must have blocking devices configured to implement this action.
- **request-block-host**—Sends a request to ARC to block this attacker host. You must have blocking devices configured to implement this action.
- **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.
- **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.
- **reset-tcp-connection**—Sends TCP resets to hijack and terminate the TCP flow. **Reset TCP Connection** only works on TCP signatures that analyze a single connection. It does not work for sweeps or floods.
- **modify-packet-inline**— Modifies packet data to remove ambiguity about what the end point might do with the packet.

• **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.

• **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.

Note Signature 64000 subsignature 0 will fire when it sees the alerts from signature 2000 subsignature 0 and signature 3000 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.

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 c1) at the beginning of the list.

```
sensor(config-sig-sig-met)# component-list insert c1 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 2000
```

Step 7 Exit component list submode:.

```
sensor(config-sig-sig-met-com)# exit
```

Step 8 Insert another signature (named c2) at the end of the list.

```
sensor(config-sig-sig-met)# component-list insert c2 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 3000
```

Step 10 Verify the settings.

```
sensor(config-sig-sig-met-com)# exit
sensor(config-sig-sig-met)# show settings
   meta
   ----------------------------------------------
      event-action: produce-alert <defaulted>
      meta-reset-interval: 60 <defaulted>
      component-list (min: 1, max: 8, current: 2 - 2 active, 0 inactive)
      ----------------------------------------------
      ACTIVE list-contents
      ----------------------------------------------
         NAME: c1
         ----------------------------------------------
            component-sig-id: 2000
            component-subsig-id: 0 <defaulted>
            component-count: 1 <defaulted>
         ----------------------------------------------
         ----------------------------------------------
         NAME: c2
         ----------------------------------------------
            component-sig-id: 3000
            component-subsig-id: 0 <defaulted>
            component-count: 1 <defaulted>
         ----------------------------------------------
         ----------------------------------------------
      ----------------------------------------------
      meta-key
      ----------------------------------------------
         Axxx
         ----------------------------------------------
            unique-victims: 1 <defaulted>
         ----------------------------------------------
      ----------------------------------------------
      component-list-in-order: false <defaulted>
   ----------------------------------------------
```

```
sensor(config-sig-sig-met)#
```

Step 11 Exit signature definition submode.

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

Step 12 Press **Enter** to apply the changes or enter **no** to discard them.

For More Information

- For detailed information about event actions, see Event Actions, page 7-8.
- For more information on the Meta engine, see Meta Engine, page B-19.
- For more information on the Signature Event Action Processor, see Signature Event Action Processor, page 7-6.

CHAPTER 9

Configuring Anomaly Detection

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.

This chapter describes anomaly detection and its features and how to configure them. It contains the following topics:

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 6.1 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 out 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 again 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.

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 Turning Off Anomaly Detection, page 9-48.

Anomaly Detection Modes

Anomaly detection 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

  Although anomaly detection is in detect mode by default, it 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.

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 6-3.
- 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 more information on configuring zones, see Configuring the Internal Zone, page 9-11, Configuring the Illegal Zone, page 9-19, 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 and 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. Configure the anomaly detection zones and protocols.
4. By default, the anomaly detection operational mode is set to detect, although for the first 24 hours it 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.
5. 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.

   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.

   After the time period, the sensor saves the initial KB as a baseline of the normal activity of your network.
6. If you manually set anomaly detection to learning accept mode, switch back to detect mode.
7. 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 6-3.
- 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 9-19, 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 7-8.

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 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 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.

Table 9-1 Anomaly Detection Worm Signatures (continued)

Signature ID	Subsignature ID	Name	Description
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.
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 the procedure for assigning actions to signatures, see Assigning Actions to Signatures, page 8-15.

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.

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 MyAD
Editing new instance MyAD.
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 is 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 To 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
   MyAD        255    N/A
```

```
   ad1        141    vs1
sensor#
```

Step 6 To delete an anomaly detection policy:

```
sensor# configure terminal
sensor(config)# no service anomaly-detection MyAD
sensor(config)# exit
sensor#
```

Note You cannot delete the default anomaly detection policy, ad0.

Step 7 Confirm 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 To 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 9-9.
- For the procedures for configuring anomaly detection zones, see Configuring the Internal Zone, page 9-11, Configuring the Illegal Zone, page 9-19, and Configuring the External Zone, page 9-28.
- For the procedure for configuring learning accept mode, see Configuring Learning Accept Mode, page 9-37.
- For the procedure for working with KBs, see Working With KB Files, page 9-39.

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 options apply:

- **worm-timeout**—The amount time in seconds for the worm termination timeout.

  The range is 120 to 10,000,000 seconds. The default is 600 seconds.
- **ignore**—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**—Source IP addresses that you want anomaly detection to ignore during processing.
- **dest-ip-address-range**—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>].

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 10.89.30.108-10.89.30.191
```

Step 7 Verify the settings.

```
sensor(config-ano-ign)# show settings
   ignore
   -----------------------------------------------
      enabled: true default: true
      source-ip-address-range: 10.89.30.108-10.89.30.191 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

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 options apply:

- **enabled** {**true** | **false**}—Enables/disables the zone.
- **ip-address-range**—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.

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 10.89.130.72-10.89.130.108
```

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 9-12.
- 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 9-17.

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 options apply:

- **enabled** {**true** | **false**}—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.

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.

```
sensor(config-ano-int-tcp-dst)# override-scanner-settings yes
sensor(config-ano-int-tcp-dst-yes)#
```

You can use the default scanner values, or you can override them and configure your own scanner values.

Step 7 To add a histogram for the new scanner settings.

```
sensor(config-ano-int-tcp-dst-yes)# threshold-histogram low num-source-ips 100
```

Enter the number of destination IP addresses (low, medium, or high) and the number of source IP addresses you want associated with this histogram.

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 options apply:

- **enabled** {**true** | **false**}—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.

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.

```
sensor(config-ano-int-udp-dst)# override-scanner-settings yes
sensor(config-ano-int-udp-dst-yes)#
```

You can use the default scanner values, or you can override them and configure your own scanner values.

Step 7 To add a histogram for the new scanner settings.

```
sensor(config-ano-int-udp-dst-yes)# threshold-histogram low num-source-ips 100
```

Enter the number of destination IP addresses (low, medium, or high) and the number of source IP addresses you want associated with this histogram.

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 options apply:

- **enabled** {**true** | **false**}—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.

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.

```
sensor(config-ano-int-oth-pro)# override-scanner-settings yes
sensor(config-ano-int-oth-pro-yes)#
```

You can use the default scanner values, or you can override them and configure your own scanner values.

Step 7 To add a histogram for the new scanner settings.

```
sensor(config-ano-int-oth-pro-yes)# threshold-histogram high num-source-ips 75
```

Enter the number of destination IP addresses (low, medium, or high) and the number of source IP addresses you want associated with this histogram.

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

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** |**other**] command in service anomaly detection submode to enable the illegal zone, add IP addresses to the illegal zone, and specify protocols.

The following options apply:

- **enabled** [true | false]—Enables/disables the zone.
- **ip-address-range**—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.

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 10.89.130.72-10.89.130.108
```

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 UPD protocol, see Configuring UDP Protocol for the Illegal Zone, page 9-23.
- 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 options apply:

- **enabled** [true | false]—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.

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 To override the scanner values for that port:

```
sensor(config-ano-ill-tcp-dst)# override-scanner-settings yes
sensor(config-ano-ill-tcp-dst-yes)#
```

You can use the default scanner values, or you can override them and configure your own scanner values.

Step 7 To add a histogram for the new scanner settings:

```
sensor(config-ano-ill-tcp-dst-yes)# threshold-histogram low num-source-ips 100
```

Enter the number of destination IP addresses (low, medium, or high) and the number of source IP addresses you want associated with this histogram.

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 options apply:

- **enabled** [true | false]—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.

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 To override the scanner values for that port:

```
sensor(config-ano-ill-udp-dst)# override-scanner-settings yes
sensor(config-ano-ill-udp-dst-yes)#
```

You can use the default scanner values, or you can override them and configure your own scanner values.

Step 7 To add a histogram for the new scanner settings:

```
sensor(config-ano-ill-udp-dst-yes)# threshold-histogram low num-source-ips 100
```

Enter the number of destination IP addresses (low, medium, or high) and the number of source IP addresses you want associated with this histogram.

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 options apply:

- **enabled** [true | false]—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.

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 To override the scanner values for that protocol:

```
sensor(config-ano-ill-oth-pro)# override-scanner-settings yes
sensor(config-ano-ill-oth-pro-yes)#
```

You can use the default scanner values, or you can override them and configure your own scanner values.

Step 7 To add a histogram for the new scanner settings:

```
sensor(config-ano-ill-oth-pro-yes)# threshold-histogram high num-source-ips 75
```

Enter the number of destination IP addresses (low, medium, or high) and the number of source IP addresses you want associated with this histogram.

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 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

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 options apply:

- **enabled** [true | false]—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.

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 9-31.
- 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 options apply:

- **enabled** [true | false]—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.

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 To override the scanner values for that port:

```
sensor(config-ano-ext-tcp-dst)# override-scanner-settings yes
sensor(config-ano-ext-tcp-dst-yes)#
```

You can use the default scanner values, or you can override them and configure your own scanner values.

Step 7 To add a histogram for the new scanner settings:

```
sensor(config-ano-ext-tcp-dst-yes)# threshold-histogram low num-source-ips 100
```

Enter the number of destination IP addresses (low, medium, or high) and the number of source IP addresses you want associated with this histogram.

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 options apply:

- **enabled** [true | false]—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.

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 To override the scanner values for that port:

```
sensor(config-ano-ext-udp-dst)# override-scanner-settings yes
sensor(config-ano-ext-udp-dst-yes)#
```

You can use the default scanner values, or you can override them and configure your own scanner values.

Step 7 To add a histogram for the new scanner settings:

```
sensor(config-ano-ext-udp-dst-yes)# threshold-histogram low num-source-ips 100
```

Enter the number of destination IP addresses (low, medium, or high) and the number of source IP addresses you want associated with this histogram.

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 options apply:

- **enabled** [true | false]—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.

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 To override the scanner values for that protocol:

```
sensor(config-ano-ext-oth-pro)# override-scanner-settings yes
sensor(config-ano-ext-oth-pro-yes)#
```

You can use the default scanner values, or you can override them and configure your own scanner values.

Step 7 To add a histogram for the new scanner settings:

```
sensor(config-ano-ext-oth-pro-yes)# threshold-histogram high num-source-ips 75
```

Enter the number of destination IP addresses (low, medium, or high) and the number of source IP addresses you want associated with this histogram.

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 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 9-37
- Saving and Loading KBs Manually, page 9-41

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 Learning accept mode uses the sensor local time.

The following options 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**—Lets you specify 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.

Note You can load the KB using the **anomaly-detection load** command.

  - **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 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.

```
sensor(config-ano)# learning-accept-mode auto
sensor(config-ano-auto)#
```

Go to Step 4.

b. Specify that the KB is going to be manually saved and loaded.

```
sensor(config-ano)# learning-accept-mode manual
```

Step 4 Specify how you want the KB automatically accepted.

a. To save the KB so that you can inspect it and decide whether to load it.

```
sensor(config-ano-aut)# action save-only
```

Go to Step

b. To have the KB saved and loaded as the current KB according to the schedule you define.

```
sensor(config-ano-aut)# action rotate
```

Continue with Step 5.

Step 5 Schedule the automatic KB saves and loads:

- Calendar schedule

```
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
```

With this schedule the KB is saved and loaded every Monday at midnight.

- Periodic schedule

```
sensor(config-ano-aut)# schedule periodic-schedule
sensor(config-ano-aut-per)# start-time 24:00:00
sensor(config-ano-aut-per)# interval 24
```

With this schedule the KB is saved and loaded every 24 hours at midnight.

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.

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

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 options 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 anomaly detection 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.

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 To 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 To 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 options 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 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.

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 To 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 To rename a KB file:

```
sensor# rename ad-knowledge-base vs0 2006-Mar-16-10_00_00 My-KB
sensor#
```

Step 5 To remove a KB file from a specific virtual sensor:

```
sensor# erase ad-knowledge-base vs0 2006-Mar-16-10_00_00
sensor#
```

Step 6 To 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 To 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 created 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 4-42.
- For the procedure for adding TLS trusted hosts, see Adding TLS Trusted Hosts, page 4-46.

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 options apply:

- *virtual-sensor*—Name of the virtual sensor that contains the KB files you want to compare.
- *name1*—Name of the first existing KB file to compare.

- *name2*—Name of the second existing KB file to compare.
- **current**—The currently loaded KB.
- **initial**—The initial KB.
- **file**—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%.

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 options apply:

- *virtual-sensor*—Name of the virtual sensor that contains the KB files you want to compare.
- *name*—Name of the existing KB file.
- **current**—The currently loaded KB.
- **initial**—The initial KB.
- **file**—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*—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*—The protocol number. The valid values are 0 to 255.

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 To 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 To 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 To 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 To 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#
```

Turning Off Anomaly Detection

If 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.

CHAPTER 10

Configuring External Product Interfaces

This chapter explains how to configure external product interfaces. It contains the following sections:

Understanding External Product Interfaces

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.

Note In Cisco IPS 6.1, you can only add interfaces to the CSA MC.

Understanding CSA MC

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.

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.

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 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 CSA MC and the IPS sensor are maintained through SSL/TLS. The sensor initiates SSL/TLS communications with CSA MC. This communication is mutually authenticated. 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 4-46.

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 7-25 and Displaying and Clearing OS Identifications, page 7-29.
- For the procedure for adding trusted hosts, see Adding TLS Trusted Hosts, page 4-46.

Configuring CSA MC to Support the IPS Interface

You must configure CSA MC to send host posture events and quarantined IP address events to the sensor.

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*.

To configure 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.

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 CSA MC.

Adding External Product Interfaces and Posture ACLs

Caution In Cisco IPS 6.1, the only external product interfaces you can add are CSA MC interfaces. Cisco IPS 6.1 supports two CSA MC interfaces.

Use the **cisco-security-agents-mc-settings** *ip-address* command in service external product interfaces submode to add CSA MC as an external product interface.

The following options apply:

- **enabled {yes | no}**—Enables/disables the receipt of information from CSA MC.
- **host-posture-settings**—Specifies how host postures received from CSA MC are handled.
  - **allow-unreachable-postures {yes | no}**—Allow postures for hosts that are not reachable by CSA MC.

    A host is not reachable if 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 CSA MC are also not reachable by the IPS, for example if the IPS and CSA MC are on the same network segment.
  - **enabled {yes | no}**—Enables/disables receipt of host postures from CSA MC.
  - **posture-acls** {**edit | insert | move**} **name1 {begin | end | inactive | before | after}**—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}**—Permit or deny postures that match the specified network address.
  - **network-address** *address*—The network address, in the form x.x.x.x/nn, for postures to be permitted or denied.
- **password**—The password used to log in to CSA MC.
- **port** —The TCP port to connect to on CSA MC. The valid range is 1 to 65535. The default is 443.
- **username** —The username used to log in to CSA MC.
- **watchlist-address-settings**—Specifies how watch listed addresses received from CSA MC are handled.
  - **enabled {yes | no}**—Enables/disables receipt of watch list addresses from CSA MC.
  - **manual-rr-increase**—The number added to an event risk rating because the attacker has been manually watch-listed by CSA MC. The valid range is 0 to 35. The default is 25.
  - **packet-rr-increase**—The number added to an event risk rating because the attacker has been watch listed by CSA MC because of a sessionless packet-based policy violation. The valid range is 0 to 35. The default is 10.
  - **session-rr-increase**—The number added to an event risk rating because the attacker has been watch-listed by 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.

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 submode.

```
sensor# configure terminal
sensor(config)# service external-product-interface
```

Step 3 Add the CSA MC interface.

```
sensor(config-ext)# cisco-security-agents-mc-settings 10.89.146.25
sensor(config-ext-cis)#
```

Step 4 Enable receipt of information from CSA MC.

```
sensor(config-ext-cis)# enabled yes
```

Step 5 To 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 though 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. To change the percentage of the manual watch list risk rating from the default of 25:

```
sensor(config-ext-cis-wat)# manual-rr-increase 30
```

c. To change the percentage of the session-based watch list risk rating from the default of 25:

```
sensor(config-ext-cis-wat)# session-rr-increase 30
```

d. To change the percentage of the packet-based watch list risk rating 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 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 CSA MC are also not reachable by the IPS, for example if the IPS and 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 171.171.171.0/24
```

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: 10.89.146.25
      ---------------------------------------------
         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: 171.171.171.0/24
                       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 4-46.

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.
- 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 CSA MC using the browser.
- Check Event Store for CSA MC subscription errors.

For More Information

- For the procedure for adding trusted hosts, see Adding TLS Trusted Hosts, page 4-46.
- For the procedure for displaying events, see Clearing Events from Event Store, page 7-38.

CHAPTER 11

Configuring IP Logging

This chapter describes how to configure IP logging on the sensor. It contains the following sections:

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.

Caution Enabling IP logging slows down system performance.

Note 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 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 On IPS sensors with multiple processors (for example, the IPS 4260 and IPS 4270-20), 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.

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. To reset the parameters, use the **default** keyword.

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.`

The following options 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 you want the sensor to log packets.

  The valid value is 0 to 65535 minutes. The default is 30 minutes.
- **ip-log-bytes** —Identifies the maximum number of bytes you want logged.

  The valid value is 0 to 2147483647. The default is 0.

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

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.

```
sensor(config-sig-ip)# ip-log-packets 200
```

Step 4 Specify the duration you want the sensor to log packets.

```
sensor(config-sig-ip)# ip-log-time 60
```

Step 5 Specify the number of bytes you want logged.

```
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 11-7.
- For more information, see Assigning Actions to Signatures, page 8-15 and Configuring Event Action Overrides, page 7-16.

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.

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.

The following options apply:

- *name*—Virtual sensor on which to begin and end logging.
- *ip_address*—Logs packets containing the specified source and/or destination IP address.
- *minutes*—Duration the logging should be active. The valid range is 1 to 60 minutes. The default is 10 minutes.
- *numPackets*—Maximum number of packets to log. The valid range is 0 to 4294967295. The default is 1000 packets.
- *numBytes*—Maximum number of bytes to log. The valid range is 0 to 4294967295. A value of 0 indicates unlimited bytes.

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.

```
sensor# iplog vs0 10.16.0.0 duration 5
Logging started for virtual sensor vs0, IP address 10.16.0.0, 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 10.16.0.0.

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:       10.16.0.0
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 11-6.
- To log IP packets as an event associated with a signature, see Configuring Automatic IP Logging, page 11-2.
- To copy and view an IP log file, see Copying IP Log Files to Be Viewed, page 11-7.

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 options apply:

- *log_id*—(Optional) 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*—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.

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:           10.1.1.2
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:           10.2.3.1
IP Address 2:           10.2.3.4
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   10.1.1.2      started     N/A        2003/07/30
2342     vs0   10.2.3.1      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.

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.

Note 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.

The following options apply:

- *log_id*—Log ID of the logging session to stop. Use the **iplog-status** command to find the log ID.
- *name*—Virtual sensor on which to begin or end logging.

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 To stop a particular IP logging session:

a. Find the log ID of the session you want to stop.

```
sensor# iplog-status
Log ID:             1
IP Address 1:       10.16.0.0
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 To stop all IP logging sessions on a virtual sensor:

```
sensor# no iplog name vs0
```

Step 4 Verify that IP logging has been stopped.

```
sensor# iplog-status
Log ID:             1
IP Address 1:       10.16.0.0
Virtual Sensor:     vs0
Status:             completed
```

```
Event ID:           0
Bytes Captured:     0
Packets Captured:   0
sensor#
```

When the logs are stopped, the status shows them as completed.

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 options apply:

- *log_id*—The log ID of the logging session. You can retrieve the log ID using the **iplog-status** command.
- *destination_url*—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.

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:          10.1.1.2
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:          10.2.3.1
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@10.16.0.0/user/iplog1
Password: ******** Connected to 10.16.0.0 (10.16.0.0). 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/.

CHAPTER 12

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:

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.

Note Capturing live traffic off the interface does not disrupt any of the functionality of the sensor.

Caution Changing the interface configuration results in abnormal termination of any **packet** command running on that interface.

Caution Executing the **packet display** or **capture** command causes significant performance degradation.

Note On IPS sensors with multiple processors (for example, the IPS 4260 and IPS 4270-20), 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.

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 options apply:

- *interface_name*—Interface name, interface type (GigabitEthernet, FastEthernet, Management) followed by slot/port. You can only use an interface name that exists in the system.
- **snaplen**—(Optional) 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) 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**—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 the IDSM2 and 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 = username of user initiating capture

  id = CLI ID of the user

  cliCmd = command entered to perform the capture

Caution Executing the **packet display** command causes significant performance degradation.

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 To 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 options apply:

- *interface_name*—Logical interface name. You can only use an interface name that exists in the system.
- **snaplen**—Maximum number of bytes captured for each packet (optional). The valid range is 68 to 1600. The default is 0.

- **count**—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**—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

  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.

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 To 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 To 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 options apply:

- **packet-file**—Locally stored libpcap file that you captured using the **packet capture** command.
- *destination_url*—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@64.101.182.20/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#
```

CHAPTER 13

Configuring Attack Response Controller for Blocking and Rate Limiting

Note ARC is formerly known as Network Access Controller. Although the name has been changed, IDM, IME, and the CLI contain references to Network Access Controller, **nac**, and **network-access**.

This chapter provides information for setting up ARC to perform blocking and rate limiting on the sensor. It the following sections:

Understanding Blocking

ARC is responsible for managing network devices in response to suspicious events by blocking access from attacking hosts and networks. 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. ARC monitors the time for the block and removes the block after the time has expired.

Caution Blocking is not supported on the FWSM in multiple mode admin context.

Note 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, ARC may not apply blocks in a timely manner or may not be able to apply blocks at all.

For security appliances configured in multi-mode, IPS 6.1 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 security appliances. 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 check the Request Block Host or Request Block Connection check boxes as the event action for particular signatures, and add them to any event action overrides you have configured, so that SensorApp sends a block request to ARC when the signature is triggered. When ARC receives the block request from 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 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

Note ARC is formerly known as Network Access Controller. Although the name has been changed, IDM, IME, and the CLI contain references to Network Access Controller, **nac**, and **network-access**.

Tip To check the status of 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. To see the status of ARC, in IDM choose **Monitoring > Sensor Monitoring > Support Information > Statistics**.To see the status of ARC, in IME choose **Configuration > *sensor_name* > Sensor Monitoring > Support Information > Statistics**.

For More Information

- For the procedure to add Request Block Host or Request Block Connection event actions to a signatures, see Assigning Actions to Signatures, page 8-15.
- For the procedure for configuring overrides that add the Request Block Host or Request Block Connection event actions to alerts of specific risk rating, see Configuring Event Action Overrides, page 7-16.
- For more information on Pre- and Post-Block ACLs, see How the Sensor Manages Devices, page 13-21.

Understanding Rate Limiting

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. 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.

Tip To check the status of 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.

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.

Table 13-1 lists the supported rate limiting signatures and parameters.

Table 13-1 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
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

For More Information

- For the procedure for configuring rate limiting on a router, see Configuring Blocking and Rate Limiting Devices, page 13-20.
- For the procedure for configuring a sensor to be a master blocking sensor, see Configuring the Sensor to be a Master Blocking Sensor, page 13-28.

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, to block on 10 security appliances and 10 routers with one blocking interface/direction each, assign 10 to the sensor and assign the other 10 to a master blocking sensor.

Before you configure 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 13-28.

Supported Devices

Caution If the recommended limits are exceeded, ARC may not apply blocks in a timely manner or may not be able to apply blocks at all.

By default, ARC supports up to 250 devices in any combination. The following devices are supported for blocking by 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 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 ARC cannot perform rate limits on 7500 routers with VIP. 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 13-7
- Disabling Blocking, page 13-8
- Specifying Maximum Block Entries, page 13-10
- Specifying the Block Time, page 13-12
- Enabling ACL Logging, page 13-13

- Enabling Writing to NVRAM, page 13-14
- Logging All Blocking Events and Errors, page 13-15
- Configuring the Maximum Number of Blocking Interfaces, page 13-17
- Configuring Addresses Never to Block, page 13-18

Allowing the Sensor to Block Itself

Use the **allow-sensor-block [true | false]** command in the service network access submode to configure 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.

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.

```
sensor(config-net-gen)# allow-sensor-block true
```

By default, this value is **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: 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: 11.11.11.11
         -----------------------------------------------
      -----------------------------------------------
      never-block-networks (min: 0, max: 250, current: 1)
      -----------------------------------------------
         ip-address: 12.12.0.0/16
         -----------------------------------------------
```

```
      ---------------------------------------------
      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: 11.11.11.11
         ---------------------------------------------
      ---------------------------------------------
      never-block-networks (min: 0, max: 250, current: 1)
      ---------------------------------------------
         ip-address: 12.12.0.0/16
         ---------------------------------------------
      ---------------------------------------------
      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

Use the **block-enable {true | false}** command in the service network access submode to enable or disable blocking on the sensor.

Note For blocking to operate, you must set up devices to do the blocking.

By default, blocking is enabled on the sensor. If 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 ARC could be making a change at the same time on the same device. This could cause the device and/or 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, 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.

```
sensor(config-net-gen)# block-enable false
```

By default, this value is **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: 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: 11.11.11.11
         ----------------------------------------------
      ----------------------------------------------
      never-block-networks (min: 0, max: 250, current: 1)
      ----------------------------------------------
         ip-address: 12.12.0.0/16
         ----------------------------------------------
      ----------------------------------------------
      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: 11.11.11.11
         ----------------------------------------------
      ----------------------------------------------
      never-block-networks (min: 0, max: 250, current: 1)
      ----------------------------------------------
         ip-address: 12.12.0.0/16
         ----------------------------------------------
      ----------------------------------------------
      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 procedures to set up devices for blocking, see Configuring the Sensor to Manage Cisco Routers, page 13-22 and Configuring the Sensor to Manage Catalyst 6500 Series Switches and Cisco 7600 Series Routers, page 13-25.

Specifying Maximum Block Entries

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.

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.

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: 11.11.11.11
         -----------------------------------------------
      -----------------------------------------------
      never-block-networks (min: 0, max: 250, current: 1)
      -----------------------------------------------
         ip-address: 12.12.0.0/16
         -----------------------------------------------
      -----------------------------------------------
      block-hosts (min: 0, max: 250, current: 0)
      -----------------------------------------------
--MORE--
```

Step 6 To 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: 11.11.11.11
         ----------------------------------------------
      ----------------------------------------------
      never-block-networks (min: 0, max: 250, current: 1)
      ----------------------------------------------
         ip-address: 12.12.0.0/16
         ----------------------------------------------
      ----------------------------------------------
      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

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.

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.

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.

```
sensor(config-rul-gen)# global-block-timeout 60
```

The value is the time duration of the block event in minutes (0 to 10000000).

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 To disable ACL logging, use 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 To 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 submode.

```
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 submode.

```
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 To 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. You can specify a single host or an entire network.

The following options apply:

- *ip_address*—IP address of the device that should never be blocked.
- *ip_address/netmask*— 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.

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.

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 10.16.0.0
```

- For an entire network:

```
sensor(config-net-gen)# never-block-networks 10.0.0.0/8
```

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: 10.16.0.0
         ----------------------------------------------
         ip-address: 11.11.11.11
         ----------------------------------------------
      ----------------------------------------------
      never-block-networks (min: 0, max: 250, current: 2)
      ----------------------------------------------
         ip-address: 10.0.0.0/8
         ----------------------------------------------
         ip-address: 12.12.0.0/16
--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 more information on how to configure event action filters, see Configuring Event Action Overrides, page 7-16.

Configuring User Profiles

Use the **user-profiles** *profile_name* command in the service network access submode to set up user profiles for the other devices that the senor 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.

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.

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:

- Configuring the Sensor to Manage Catalyst 6500 Series Switches and Cisco 7600 Series Routers, page 13-25
- Configuring the Sensor to Manage Cisco Firewalls, page 13-27

How the Sensor Manages Devices

Note ACLs do not apply to rate limiting devices.

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 ARC reads the lines in the ACL and copies these lines to the beginning of the ACL.

3. Any active blocks
4. Either:
   - Post-Block ACL (if specified)

     This ACL must already exist on the device.

     Note 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.

   - **permit ip any any** (not used if a Post-Block ACL is specified)

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. ARC then reverses the process on the next cycle.

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.

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 13-6.
- For the procedure for configuring the sensor to be a master blocking sensor, see Configuring the Sensor to be a Master Blocking Sensor, page 13-28.

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

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.

Note Pre-Block and Post-Block ACLS do not apply to rate limiting.

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 ARC.

```
sensor(config-net)# router-devices ip_address
```

Step 4 Enter the logical device name that you created when you configured the user profile.

```
sensor(config-net-rou)# profile-name user_profile_name
```

ARC accepts anything you enter. It does not check to see if the user profile exists.

Step 5 Specify the method used to access the sensor.

```
sensor(config-net-rou)# communication {telnet | ssh-des | ssh-3des}
```

If unspecified, SSH 3DES is used.

Note If you are using DES or 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 sensor's address 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. For rate limiting only:

```
sensor(config-net-rou)# response-capabilities rate-limit
```

b. For 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: 10.89.127.97
   ----------------------------------------------
      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 13-19.
- For the procedure for adding the device to the known hosts list, see Adding Hosts to the SSH Known Hosts List, page 4-42.

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

You can configure 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 V ACL, the sensor inserts **permit ip any any** at the end of the new VACL.

Note IDSM-2 inserts **permit ip any any capture** 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 blocking using router ACLs, see Configuring Blocking and Rate Limiting Devices, page 13-20.

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 ARC.

```
sensor(config-net)# cat6k-devices ip_address
```

Step 4 Enter the user profile name that you created when you configured the user profile.

```
sensor(config-net-cat)# profile-name user_profile_name
```

Note ARC accepts anything you type. It does not check to see if the logical device exists.

Step 5 Specify the method used to access the sensor.

```
sensor(config-net-cat)# communication {telnet | ssh-des | ssh-3des}
```

If unspecified, SSH 3DES is used.

Note If you are using DES or 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 13-19.
- For the procedure for adding the device to the known hosts list, see Adding Hosts to the SSH Known Hosts List, page 4-42.

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 ARC.

```
sensor(config-net)# firewall-devices ip_address
```

Step 4 Enter the user profile name that you created when you configured the user profile.

```
sensor(config-net-fir)# profile-name user_profile_name
```

ARC accepts anything you type. It does not check to see if the logical device exists.

Step 5 Specify the method used to access the sensor.

```
sensor(config-net-fir)# communication {telnet | ssh-des | ssh-3des}
```

If unspecified, SSH 3DES is used.

Note If you are using DES or 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-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 13-19.
- For the procedure for adding the device to the known hosts list, see Adding Hosts to the SSH Known Hosts List, page 4-42.

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.

Note When you add a master blocking sensor, you reduce the number of blocking devices per sensor. For example, to block on 10 firewalls and 10 routers with one blocking interface/direction each, 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** *mbs_ip_address* command in the service network access submode to configure a master blocking sensor.

The following options apply:

- *mbs_ip_address*—IP address of sensor for forward block requests.
- **password**—Account password of sensor for forward block requests.
- **port**—Port of sensor for forward block requests.
- **tls {true | false}**—Set to true if the remote sensor requires TLS; otherwise set to false.
- **username**—Account name of sensor for forward block requests.

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.

```
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)#
```

If `enable-tls` is true, go to Step b.

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 mbs_ip_address port port_number
```

Example

```
sensor(config)# tls trusted-host ip-address 10.0.0.0 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 host certificate's fingerprints 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 submode.

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

Step 7 Add a master blocking sensor entry.

```
sensor(config-net-gen)# master-blocking-sensors mbs_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.

```
sensor(config-net-gen-mas)# port port_number
```

The default is 80/443 if not specified.

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** *mbs_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 4-5.

Configuring Host Blocking

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.

Note If you do not configure the amount of time for the host block, it is permanent.

Note Connection blocks and network blocks are not supported on security appliances. Security appliances only support host blocks with additional connection information.

The following options apply:

- *ip-address*—IP address of the host to be blocked.
- *minutes*—(Optional) Duration of the host block in minutes. The valid range is 0 to 70560 minutes.

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:.

```
sensor# block host 10.2.3.1 timeout 15
```

The host block ends in 15 minutes.

Step 3 To start a host block:

```
sensor# block host 10.2.3.1
```

The host block lasts until you remove it.

Step 4 To end the host block:

```
sensor# no block host 10.2.3.1
sensor#
```

Configuring Network Blocking

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.

Note If you do not configure the amount of time for the network block, it is permanent.

Note Connection blocks and network blocks are not supported on security appliances. Security appliances only support host blocks with additional connection information.

The following options apply:

- *ip-address/netmask*—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) Duration of the network block in minutes. The valid range is 0 to 70560 minutes.

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:.

```
sensor# block network 10.0.0.0/8 timeout 15
```

The network block ends in 15 minutes.

Step 3 To start a network block:

```
sensor# block network 10.0.0.0/8
```

The network block lasts until you remove it.

Step 4 To end the network block:

```
sensor# no block network 10.0.0.0/8
sensor#
```

Configuring Connection Blocking

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.

Note If you do not configure the amount of time for the connection block, it is permanent.

Note Connection blocks and network blocks are not supported on security appliances. Security appliances only support host blocks with additional connection information.

The following options apply:

- *source-ip-address*—Source IP address in a connection block.
- *destination-ip-address*—Destination IP address in a connection block.
- *port-number*—(Optional) Destination port number. The valid range is 0 to 65535.
- *type*—(Optional) Protocol type. The valid types are **tcp** or **udp**.
- *minutes*—(Optional) Duration of the connection block in minutes. The valid range is 0 to 70560 minutes.

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.

```
sensor# block connection 10.2.3.1 11.2.3.1 port 80 protocol tcp timeout 30
```

The connection block ends in 30 minutes.

Step 3 To start a connection block:

```
sensor# block connection 10.2.3.1 11.2.3.1
```

The connection block lasts until you remove it.

Step 4 To end the connection block:

```
sensor# no block connection 10.2.3.1
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 ARC.

```
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
```

The `Host` entry indicates which hosts are being blocked and how long the blocks are.

CHAPTER 14

Configuring SNMP

This chapter describes how to configure SNMP. It contains the following sections:

Understanding SNMP

Caution To have the sensor send SNMP traps, you must also select **request-snmp-trap** as the event action when you configure signatures.

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.

For More Information

For the procedure for assigning actions to signatures, see Assigning Actions to Signatures, page 8-15.

Configuring SNMP

Configure general SNMP parameters in the service notification submode.

The following options 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**—Remove an entry or selection setting.
- **read-only-community**—The read-only community name for the SNMP agent. The default is public.
- **read-write-community**—The read-write community name for the SNMP agent. The default is private.
- **snmp-agent-port**—The port the SNMP agent will listen on. The default SNMP port number is 161.
- **snmp-agent-protocol**—The protocol the SNMP agent will communicate with. The default protocol is UDP.
- **system-contact**—The contact information for this sensor. The system-contact option modifies the SNMPv2-MIB::sysContact.0 value.
- **system-location**—The location of the sensor. The system-location option modifies the SNMPv2-MIB::sysLocation.0 value.

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.

```
sensor(config-not)# read-only-community PUBLIC1
```

The read-only community name specifies the password for queries to the SNMP agent.

b. Assign the read-write community string.

```
sensor(config-not)# read-write-community PRIVATE1
```

The read-write community name specifies the password for **sets** to the SNMP agent.

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 senor, 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.

Configuring SNMP Traps

Caution To have the sensor send SNMP traps, you must also select **request-snmp-trap** as the event action when you configure signatures.

Configure the SNMP traps in the service notification submode.

The following options 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-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**—The community name used when sending traps if no name is specified when defining the trap destinations.
- **trap-destinations**—Defines the destinations to send error events and alert events generated from signature actions.
  - **trap-community-name**—The community name used when sending the trap. If no community name is specified the general trap community name is used.
  - **trap-port**—The port number to send the SNMP trap to.

To configure SNMP 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. 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 10.1.1.1
```

b. Enter the UDP port of the SNMP management station.

```
sensor(config-not-tra)# trap-port 162
```

The default is 162.

c. Enter the trap Community string.

```
sensor(config-not-tra)# trap-community-name AUSTIN_PUBLI
```

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.

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: 10.1.1.1
      ---------------------------------------------
         trap-community-name: AUSTIN_PUBLIC default:
         trap-port: 161 default: 162
      ---------------------------------------------
   -----------------------------------------------
   error-filter: warning|error|fatal default: error|fatal
   enable-detail-traps: true default: false
   enable-notifications: true default: false
   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: PUBLIC1 default: public
   system-location: AUSTIN default: Unknown
   system-contact: BUSINESS default: Unknown
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 8-15.

Supported MIBS

The following private MIBs are supported on the sensor:

- CISCO-CIDS-MIB
- 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.

CHAPTER 15

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

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
! ------------------------------
! Version 6.1(1)
! Current configuration last modified Thu Jul 14 21:49:58 2008
! ------------------------------
display-serial
! ------------------------------
service interface
exit
! ------------------------------
service analysis-engine
exit
! ------------------------------
service authentication
exit
! ------------------------------
```

```
service event-action-rules rules0
exit
! ------------------------------
service host
network-settings
host-ip 10.89.149.27/25,10.89.149.126
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 logger
exit
! ------------------------------
service network-access
user-profiles test
exit
exit
! ------------------------------
service notification
exit
! ------------------------------
service signature-definition sig0
signatures 60000 0
alert-severity medium
sig-fidelity-rating 75
sig-description
sig-name My Sig
sig-string-info My Sig Info
sig-comment Sig Comment
exit
engine string-tcp
event-action produce-alert
direction to-service
regex-string My Regex String
service-ports 23
exit
event-counter
event-count 1
event-count-key Axxx
specify-alert-interval no
exit
alert-frequency
summary-mode summarize
summary-interval 15
summary-key Axxx
specify-global-summary-threshold yes
global-summary-threshold 75
exit
exit
exit
exit
exit
! ------------------------------
service ssh-known-hosts
exit
! ------------------------------
service trusted-certificates
```

```
exit
! ------------------------------
service web-server
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 host submode.

```
sensor# configure terminal
sensor(config)# service host
sensor(config-hos)# show settings
   network-settings
   ---------------------------------------------
      host-ip: 10.89.149.27/25,10.89.149.126 default: 10.1.9.201/24,10.1.9.1
      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
630199473878411519325039111726689401947545491553904076580203933306118912925083 00
859403040311860144996325688124280680580895816141963373996230606249900570491030 55
901539559350860600086797768080736401860634357232523755752931263045580687043018 63
805621144374392890694566709220749958273902847616105915157520084051402436730831 89
778224699649345983670103893898882974908028841185437300762935897035359121619933 19
470931302986888300125472155726463496235394688386410649153139478068529040823519 55
132172731380999653830397161301532707152200465671078281289241976924173320339117 04
3 <defaulted>
            ---------------------------------------------
         ---------------------------------------------
         <protected entry>
         name: realm-trend.pub <defaulted>
         type
         ---------------------------------------------
            rsa-pubkey
            ---------------------------------------------
               length: 2048 <defaulted>
               exponent: 65537 <defaulted>
               modulus: 21765561422573021314159855351418723031625093380777053696
638172895270605709325510654898181907137456721482605270300606672083666066038026 79
304390667241433906264954793005501016181795846372870529364656921465726126513759 69
203545215856442216029442035208044042129754019708951199037567696011338536732967 66
452897957779734919840565870452145148200633669507313464000443084915946264347069 99
476086688228140148300633995342046470695090524434395253637065272552245107711222 35
801811504605447832514984814327059910100698443685257548784136694276397529508017 67
999053092352324562955800867242032979140959842243284443915822231384237991008381 91
9 <defaulted>
            ---------------------------------------------
         ---------------------------------------------
      ---------------------------------------------
   ---------------------------------------------
sensor(config-hos)# exit
sensor(config)# exit
sensor#
```

Step 8 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 9 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 10 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 11 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 12 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 13 Display the current configuration for the SSH known hosts submode.

```
sensor# configure terminal
sensor(config)# service ssh-known-hosts
sensor(config-ssh)# show settings
   rsa1-keys (min: 0, max: 500, current: 0)
   ---------------------------------------------
   ---------------------------------------------
sensor(config-ssh)# exit
sensor(config)# exit
sensor#
```

Step 14 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: MIICJDCCAY0CCPbSkgXUchJIMA0GCSqGSIb3DQEBBQUAMFcxCzAJBgNVBAYTA
lVTMRwwGgYDVQQKExNDaXNjbyBTeXN0ZW1zLCBJbmMuMRIwEAYDVQQLEwlTU00tSVBTMjAxFjAUBgNVB
AMTDTEwLjg5LjEzMC4xMDgwHhcNMDMwMTAzMDE1MjEwWhcNMDUwMTAzMDE1MjEwWjBXMQswCQYDVQQGE
wJVUzEcMBoGA1UEChMTQ2lzY28gU3lzdGVtcywgSW5jLjESMBAGA1UECxMJU1NNLUlQUzIwMRYwFAYDV
QQDEw0xMC44OS4xMzAuMTA4MIGfMA0GCSqGSIb3DQEBAQUAA4GNADCBiQKBgQCzldqLFG4MT4bfgh3mJ
fP/DCilnnaLfzHK9FdnhmWI4FY+9MVvAI7MOhAcuV6HYfyp6n6cYvH+Eswzl9uv7H5nouID9St9GI3Yr
SUtlIQAJ4QVL2DwWP230x6KdHrYqcj+Nmhc7AnnPypjidwGSfF+VetIJLEeRFh/mI2JcmwF2QIDAQABM
A0GCSqGSIb3DQEBBQUAA4GBAAUI2PLANTOehxvCfwd6UAFXvy8uifbjqKMC1jrrF+f9KGkxmR+XZvUaG
OS83FYDXlXJvB5Xyxms+Y01wGjzKKpxegBoan8OB8o193Ueszdpvz2xYmiEgywCDyVJRsw3hAFMXWMS5
XsBUiHtw0btHH0j7ElFZxUjZv12fGz8hlnY
   ---------------------------------------------
sensor(config-tru)# exit
sensor(config)# exit
sensor#
```

Step 15 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 options apply:

- *keyword*—Either the current-config or the backup-config.
  - **current-config**—The current running configuration. This configuration becomes persistent as the commands are entered.
  - **backup-config**—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*—Any regular expression found in the **more** command output.

Note The *regular-expression* option is case sensitive and allows for complex matching requirements.

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 10.89.149.185/25,10.89.149.254
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 5.0(0.27)
! Current configuration last modified Fri Feb 11 15:10:57 2005
! ------------------------------
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 10.89.149.185/25,10.89.149.254
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 options 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*—Any regular expression found in the **show settings** command output.

Note The *regular_expression* option is case sensitive and allows for complex matching requirements.

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: 10.89.149.124
         ----------------------------------------------
            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: 10.89.149.185/25,10.89.149.254 default: 10.1.9.201/24,10.1.9.1
sensor(config-hos)#
```

Displaying the Contents of a Logical File

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 options apply:

- *keyword*—Either the current-config or the backup-config.
  - **current-config**—The current running configuration. This configuration becomes persistent as the commands are entered.
  - **backup-config**—The storage location for the configuration backup file.

Note Operators and viewers can only display the current configuration. Only administrators can view hidden fields such as passwords.

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.

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.

```
! ------------------------------
! Version 6.1(1)
! Current configuration last modified Thu Jul 14 21:49:58 2008
! ------------------------------
display-serial
! ------------------------------
service interface
exit
! ------------------------------
service analysis-engine
```

```
exit
! ------------------------------
service authentication
exit
! ------------------------------
service event-action-rules rules0
exit
! ------------------------------
service host
network-settings
host-ip 10.89.149.27/25,10.89.149.126
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 logger
exit
! ------------------------------
service network-access
user-profiles test
exit
exit
! ------------------------------
service notification
exit
! ------------------------------
service signature-definition sig0
signatures 60000 0
alert-severity medium
sig-fidelity-rating 75
sig-description
sig-name My Sig
sig-string-info My Sig Info
sig-comment Sig Comment
exit
engine string-tcp
event-action produce-alert
direction to-service
regex-string My Regex String
service-ports 23
exit
event-counter
event-count 1
event-count-key Axxx
specify-alert-interval no
exit
alert-frequency
summary-mode summarize
summary-interval 15
summary-key Axxx
specify-global-summary-threshold yes
global-summary-threshold 75
exit
exit
exit
exit
exit
```

```
! ------------------------------
service ssh-known-hosts
exit
! ------------------------------
service trusted-certificates
exit
! ------------------------------
service web-server
exit
sensor#
```

For More Information

For the procedure for using the **terminal** command, see Modifying Terminal Properties, page 16-17.

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.

Options

The following options 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
- 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 also 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 HTTP and HTTPS prompt for a password if a username is required to access the website. If you use HTTPS protocol, 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 SSHM known host list, see Adding Hosts to the SSH Known Hosts List, page 4-42.
- For the procedure for adding the remote host to the TLS trusted hosts list, see Adding TLS Trusted Hosts, page 4-46.

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. To restore this configuration, 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.

```
sensor# copy current-config backup-config
```

The current configuration is saved in a backup file.

Step 3 Display the backup configuration file.

```
sensor# more backup-config
```

The backup configuration file is displayed.

Step 4 You can either merge the backup configuration with the current configuration, or you can overwrite the current configuration:

- To merge the backup configuration into the current configuration:

  ```
  sensor# copy backup-config current-config
  ```

- To 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 options 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.

CHAPTER 16

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:

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

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.

Note Administrators may need to disable the password recovery feature for security reasons.

Table 16-1 lists the password recovery methods according to platform.

Table 16-1 Password Recovery Methods According to Platform

Platform	Description	Recovery Method
4200 series sensors	Standalone IPS appliances	GRUB prompt or ROMMON
AIM IPS NME IPS	Router IPS modules	Bootloader command
AIP SSM	ASA 5500 series adaptive security appliance modules	ASA CLI command
IDSM2	Switch IPS module	Password recovery image file

Password Recovery for Appliances

This section describes the two ways to recover the password for appliances. It contains the following topics:

Using the GRUB Menu

For 4200 series 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.

Note You must have a terminal server or direct serial connection to the appliance to use the GRUB menu to recover the password.

To recover the password on appliances, follow these steps:

Step 1 Reboot the appliance.

The following menu appears:

```
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**. You can change the password the next time you log in to the CLI.

Using ROMMON

For the IPS 4240 and the IPS 4255 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.

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-4240-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
```

Password Recovery for the AIM IPS

To recover the password for the AIM IPS, use the **clear password** command. You must have console access to the AIM IPS and administrative access to the router.

To recover the password for the AIM IPS, follow these steps:

Step 1 Log in to the router.

Step 2 Enter privileged EXEC mode on the router.

```
router> enable
```

Step 3 Confirm the module slot number in your router.

```
router# show run | include ids-sensor
interface IDS-Sensor0/0
router#
```

Step 4 Session in to the AIM IPS.

```
router# service-module ids-sensor slot/port session
```

Example

```
router# service-module ids-sensor 0/0 session
```

Step 5 Press **Control-shift-6** followed by **x** to navigate to the router CLI.

Step 6 Reset the AIM IPS from the router console.

```
router# service-module ids-sensor 0/0 reset
```

Step 7 Press **Enter** to return to the router console.

Step 8 When prompted for boot options, enter ******* quickly.

You are now in the bootloader.

Step 9 Clear the password.

```
ServicesEngine boot-loader# clear password
```

The AIM IPS reboots. The password is reset to **cisco**. Log in to the CLI with username **cisco** and password **cisco**. You can then change the password.

Password Recovery for the AIP SSM

You can reset the password to the default (**cisco**) for the AIP SSM 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 7.2.2 or later.

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.
```

Resetting the Password Using the CLI

To reset the password on the AIP SSM, follow these steps:

Step 1 Log into the adaptive security appliance and enter the following command to verify the module slot number:

```
asa# show module
Mod Card Type                                      Model              Serial No.
--- ---------------------------------------------- ------------------ -----------
  0 ASA 5510 Adaptive Security Appliance           ASA5510            JMX1135L097
  1 ASA 5500 Series Security Services Module-40    ASA-SSM-40         JAF1214AMRL

Mod MAC Address Range                 Hw Version   Fw Version   Sw Version
--- --------------------------------- ------------ ------------ ---------------
  0 001b.d5e8.e0c8 to 001b.d5e8.e0cc  2.0          1.0(11)2     8.4(3)
  1 001e.f737.205f to 001e.f737.205f  1.0          1.0(14)5     7.0(7)E4

Mod SSM Application Name           Status           SSM Application Version
--- ------------------------------ ---------------- --------------------------
  1 IPS                            Up               7.0(7)E4

Mod Status             Data Plane Status     Compatibility
--- ------------------ --------------------- -------------
  0 Up Sys             Not Applicable
  1 Up                 Up
```

Step 2 Reset the password for module 1.

```
asa# hw-module module 1 password-reset
Reset the password on module in slot 1? [confirm]
```

Step 3 Press **Enter** to confirm.

```
Password-Reset issued for slot 1.
```

Step 4 Verify the status of the module. Once the status reads Up, you can session to the AIP SSM.

```
asa# show module 1
Mod Card Type                                    Model              Serial No.
--- -------------------------------------------- ------------------ -----------
  1 ASA 5500 Series Security Services Module-40  ASA-SSM-40         JAF1214AMRL

Mod MAC Address Range                 Hw Version   Fw Version   Sw Version
--- --------------------------------- ------------ ------------ ---------------
  1 001e.f737.205f to 001e.f737.205f  1.0          1.0(14)5     7.0(7)E4

Mod SSM Application Name           Status           SSM Application Version
--- ------------------------------ ---------------- --------------------------
  1 IPS                            Up               7.0(7)E4

Mod Status             Data Plane Status     Compatibility
--- ------------------ --------------------- -------------
  1 Up                 Up
```

Step 5 Session to the AIP SSM.

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

Step 6 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 7 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.
aip_ssm#
```

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.

Password Recovery for the IDSM2

To recover the password for the IDSM2, you must install a special password recovery image file. This installation only resets the password, all other configuration remains intact. The password recovery image is version-dependent and can be found on the Cisco Download Software site. For IPS 6.*x*, download WS-SVC-IDSM2-K9-a-6.0-password-recovery.bin.gz. For IPS 7.*x*, download WS-SVC-IDSM2-K9-a-7.0-password-recovery.bin.gz.

FTP is the only supported protocol for image installations, so make sure you put the password recovery image file on an FTP server that is accessible to the switch. You must have administrative access to the Cisco 6500 series switch to recover the password on the IDSM2.

During the password recovery image installation, the following message appears:

```
Upgrading will wipe out the contents on the hard disk.
Do you want to proceed installing it [y|n]:
```

This message is in error. Installing the password recovery image does not remove any configuration, it only resets the login account.

Once you have downloaded the password recovery image file, follow the instructions to install the system image file but substitute the password recovery image file for the system image file. The IDSM2 should reboot into the primary partition after installing the recovery image file. If it does not, enter the following command from the switch:

```
hw-module module module_number reset hdd:1
```

Note The password is reset to **cisco**. Log in to the CLI with username **cisco** and password **cisco**. You can then change the password.

For More Information

- For the procedures for reimaging the IDSM2, see Installing the IDSM2 System Image, page 22-26.
- For more information on downloading Cisco IPS software, see Obtaining Cisco IPS Software, page 21-1.

Password Recovery for the NME IPS

To recover the password for the NME IPS, use the **clear password** command. You must have console access to the NME IPS and administrative access to the router.

To recover the password for the NME IPS, follow these steps:

Step 1 Log in to the router.

Step 2 Enter privileged EXEC mode on the router.

```
router> enable
```

Step 3 Confirm the module slot number in your router.

```
router# show run | include ids-sensor
interface IDS-Sensor1/0
router#
```

Step 4 Session in to the NME IPS.

```
router# service-module ids-sensor slot/port session
```

Example

```
router# service-module ids-sensor 1/0 session
```

Step 5 Press **Control-shift-6** followed by **x** to navigate to the router CLI.

Step 6 Reset the NME IPS from the router console.

```
router# service-module ids-sensor 1/0 reset
```

Step 7 Press **Enter** to return to the router console.

Step 8 When prompted for boot options, enter `***` quickly.

You are now in the bootloader.

Step 9 Clear the password.

```
ServicesEngine boot-loader# clear password
```

The NME IPS reboots. The password is reset to **cisco**. Log in to the CLI with username **cisco** and password **cisco**. You can then change the password.

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, 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 IDM

To disable password recovery in IDM or IME, follow these steps:

Step 1 Log in to 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, such as the AIM IPS and the NME IPS bootloader, ROMMON, and the maintenance partition for the IDSM2, 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.
- When performing password recovery on the IDSM2, you see the following message: `Upgrading will wipe out the contents on the storage media`. You can ignore this message. Only the password is reset when you use the specified password recovery image.

Clearing the Sensor Databases

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.

Caution We do not recommend that you use this 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 following options apply:

- *virtual-sensor*—Name of a virtual sensor configured on the sensor.
- **all**— Clear all the 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.

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.

Configuring Health Status Information

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.

The following options 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.
- **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 IME. Disable event retrieval policy if you are not doing external event monitoring.

- **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.
- **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.
- **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

The health status categories are rated by red and green with red being critical.

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 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 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 sensor health and security monitoring.

```
sensor(config-hea)# enable-monitoring true
sensor(config-hea)#
```

Step 6 Set the event retrieval thresholds.

```
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 heartbeat events to be emitted at the specified interval of seconds.

```
sensor(config-hea)# heartbeat-events enable 20000
sensor(config-hea)#
```

Step 8 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 9 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 10 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 11 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 12 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 13 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 14 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 15 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 16 Exit health monitoring submode.

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

Step 17 Press **Enter** to apply the changes or enter **no** to discard them.

Showing Sensor Overall Health Status

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 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.

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

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 of a completed banner login:

```
This message will be displayed on login.
Thank you
login: cisco
Password:****
```

Step 5 To remove the banner login:

```
sensor(config)# no banner login
```

The banner no longer appears at login.

Terminating CLI Sessions

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 options apply:

- *cli_id*—CLI ID number associated with the login session. Use the **show users** command to find the CLI ID number.
- **message**—Message to send to the receiving user.

Caution You can only clear CLI login sessions with the **clear line** command. You cannot clear service logins with this command.

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

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.

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.

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
```

Displaying and Clearing Events

This section describes how to display and clear events from Event Store, and how to reenable generation of status events as a consequence of sensor health monitoring control transactions. It contains the following topics:

- Displaying Events, page 16-18
- Clearing Events from Event Store, page 16-21
- Generating Status Events from Health Monitoring Control Transactions, page 16-21

Displaying Events

Note The Event Store has a fixed size of 30 MB for all platforms except for AIP SSC-5, which has a fixed size of 10 MB.

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.

Note Events are displayed as a live feed. To cancel the request, press **Ctrl-C.**

The following options 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 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**—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 ARC (block) requests.

  Note ARC is formerly known as NAC. This name change has not been completely implemented throughout IDM, IME, and the CLI for Cisco IPS 6.1.
- **status**—Displays status events.
- **past**—Displays events starting in the past for the specified hours, minutes, and seconds.
- *hh:mm:ss*—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**.

To display events from Event Store, follow these steps:

Step 1 Log in to the CLI.

Step 2 Display all events starting now.

```
sensor# show events
evError: eventId=1041472274774840147 severity=warning vendor=Cisco
  originator:
    hostId: sensor2
    appName: cidwebserver
    appInstanceId: 12075
  time: 2008/01/07 04:41:45 2008/01/07 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: 2008/01/07 04:41:45 2008/01/07 04:41:45 UTC
```

```
  errorMessage: name=errTransport WebSession::sessionTask(6) TLS connection exception:
handshake incomplete.
```

The feed continues showing all events until you press **Ctrl-C**.

Step 3 Display the block requests beginning at 10:00 a.m. on February 9, 2008.

```
sensor# show events NAC 10:00:00 Feb 9 2008
evShunRqst: eventId=1106837332219222281 vendor=Cisco
  originator:
    deviceName: Sensor1
    appName: NetworkAccessControllerApp
    appInstance: 654
  time: 2008/02/09 10:33:31 2008/08/09 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, 2008.

```
sensor# show events error warning 10:00:00 Feb 9 2008
evError: eventId=1041472274774840197 severity=warning vendor=Cisco
  originator:
    hostId: sensor
    appName: cidwebserver
    appInstanceId: 12160
  time: 2008/01/07 04:49:25 2008/01/07 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: 2008/03/02 14:15:59 2008/03/02 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: 2008/01/08 02:41:00 2008/01/08 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: 2008/01/08 02:41:00 2008/01/08 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 Event Store.

To clear events from Event Store, follow these steps:

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

Step 2 Clear 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.

Generating Status Events from Health Monitoring Control Transactions

Sensor health monitoring generates control transactions every 5 seconds or so. By default, the status events generated from control transactions are filtered to reduce the large amount of status events. You can have status events generated from control transaction again by using **status-event-logging-categories** command.

To reenable status events generated from health monitoring control transactions, follow these steps:

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

Step 2 Enter service logger submode.

```
sensor(config)# service logger
sensor(config-log)#
```

Step 3 Enter event store submode.

```
sensor(config-log)# event-store
sensor(config-log-eve)#
```

Step 4 Enable event generation from control transactions.

```
sensor(config-log-eve)# status-event-logging-categories controlTransaction enabled true
sensor(config-log-eve)#
```

Step 5 Exit event store submode.

```
sensor(config-log-eve)# exit
sensor(config-log)#
```

Step 6 Exit service logger submode.

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

Step 7 Press **Enter** to apply the changes or enter **no** to discard them.

Setting 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 16-22
- Manually Setting the Clock, page 16-23

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 16-2 lists the system clock flags.

Table 16-2 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.

```
sensor# show clock detail
20:09:43 UTC Thu Apr 03 2008
Time source is NTP
Summer time starts 03:00:00 UTC Sun Mar 09 2008
Summer time stops 01:00:00 UTC Sun Nov 02 2008
```

This 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 2008
No time source
Summer time starts 03:00:00 UTC Sun Mar 09 2008
Summer time stops 01:00:00 UTC Sun Nov 02 2008
```

This indicates that no time source is configured.

Manually Setting the Clock

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.

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.

The **clock set** command does not apply to the following platforms:

- AIM IPS
- AIP SSM
- IDSM2
- NME IPS

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 2008
```

Note The time format is 24-hour time.

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 options apply:

- *virtual_sensor*—(Optional) The virtual sensor whose denied attackers list should be cleared.
- *ip_address*—(Optional) The IP address to clear.

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.

```
sensor# show statistics denied-attackers
Denied Attackers and hit count for each.
   10.20.4.2 = 9
   10.20.5.2 = 5
```

The statistics show that there are two IP addresses being denied at this time.

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 10.1.1.1
Warning: Executing this command will delete ip address 10.1.1.1 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 To clear only the statistics.

```
sensor# show statistics virtual-sensor clear
```

Step 11 Verify that you have cleared the statistics.

```
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
```

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.

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** | **authentication** | **event-server** | **event-store** | **external-product-interface** |**host** | **logger** | **network-access** | **notification** | **sdee-server** | **transaction-server** | **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**] 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.

To display statistics for the sensor, follow these steps:

Step 1 Log in to the CLI.

Step 2 Display the statistics for Analysis Engine.

```
sensor# show statistics analysis-engine
Analysis Engine Statistics
   Number of seconds since service started = 1421127
   Measure of the level of current resource utilization = 0
   Measure of the level of maximum resource utilization = 0
   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 = 0
      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
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-4240#
```

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 Event Server.

```
sensor# show statistics event-server
General
   openSubscriptions = 0
   blockedSubscriptions = 0
Subscriptions
sensor#
```

Step 7 Display the statistics for Event Store.

```
sensor# show statistics event-store
Event 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 queries issued = 0
      The number of times the event store circular buffer has wrapped = 0
   Number of events of each type currently stored
      Debug events = 0
```

```
      Status events = 9904
      Log transaction events = 0
      Shun request events = 61
      Error events, warning = 67
      Error events, error = 83
      Error events, fatal = 0
      Alert events, informational = 60
      Alert events, low = 1
      Alert events, medium = 60
      Alert events, high = 0
sensor#
```

Step 8 Display the statistics for the host.

```
sensor# show statistics host
General Statistics
   Last Change To Host Config (UTC) = 16:11:05 Thu Feb 10 2008
   Command Control Port Device = FastEthernet0/0
Network Statistics
   fe0_0     Link encap:Ethernet HWaddr 00:0B:46:53:06:AA
             inet addr:10.89.149.185 Bcast:10.89.149.255 Mask:255.255.255.128
             UP BROADCAST RUNNING MULTICAST MTU:1500 Metric:1
             RX packets:1001522 errors:0 dropped:0 overruns:0 frame:0
             TX packets:469569 errors:0 dropped:0 overruns:0 carrier:0
             collisions:0 txqueuelen:1000
             RX bytes:57547021 (54.8 Mib) TX bytes:63832557 (60.8 MiB)
             Interrupt:9 Base address:0xf400 Memory:c0000000-c0000038
NTP Statistics
   status = Not applicable
Memory Usage
   usedBytes = 500592640
   freeBytes = 8855552
   totalBytes = 509448192
Swap Usage
   Used Bytes = 77824
   Free Bytes = 600649728

   Total Bytes = 600727552
CPU Statistics
   Usage over last 5 seconds = 0
   Usage over last minute = 1
   Usage over last 5 minutes = 1
Memory Statistics
   Memory usage (bytes) = 500498432
   Memory free (bytes) = 894976032
Auto Update Statistics
   lastDirectoryReadAttempt = 15:26:33 CDT Tue Jun 17 2008
    =   Read directory: http://tester@198.133.219.243//cisco/ciscosecure/ips/6.x/sigup/
    =   Success
   lastDownloadAttempt = 15:26:33 CDT Tue Jun 17 2008
    =   Download: http://bmarquardt@198.133.219.243//cisco/ciscosecure/ips/6.x/sigup/IPS-
sig-S338-req-E1.pkg
    =   Error: httpResponse status returned:  Unauthorized
   lastInstallAttempt = N/A
   nextAttempt = 16:26:30 CDT Tue Jun 17 2008

sensor#
```

Step 9 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 10 Display the statistics for 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 = 10.89.150.219
      NATAddr = 0.0.0.0
      Communications = ssh-des
   NetDevice
      Type = PIX
      IP = 10.89.150.250
      NATAddr = 0.0.0.0
      Communications = telnet
   NetDevice
      Type = Cisco
      IP = 10.89.150.158
      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 = 10.89.150.138
      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 = 10.89.150.171
      AclSupport = Does not use ACLs
      Version = 6.3
      State = Active
      Firewall-type = PIX
   NetDevice
      IP = 10.89.150.219
      AclSupport = Does not use ACLs
      Version = 7.0
      State = Active
      Firewall-type = ASA
   NetDevice
      IP = 10.89.150.250
      AclSupport = Does not use ACLs
      Version = 2.2
      State = Active
      Firewall-type = FWSM
   NetDevice
      IP = 10.89.150.158
      AclSupport = uses Named ACLs
      Version = 12.2
      State = Active
   NetDevice
      IP = 10.89.150.138
      AclSupport = Uses VACLs
      Version = 8.4
      State = Active
   BlockedAddr
      Host
         IP = 22.33.4.5
         Vlan =
         ActualIp =
         BlockMinutes =
      Host
         IP = 21.21.12.12
         Vlan =
         ActualIp =
         BlockMinutes =
      Host
         IP = 122.122.33.4
         Vlan =
         ActualIp =
         BlockMinutes = 60
         MinutesRemaining = 24
      Network
         IP = 111.22.0.0
         Mask = 255.255.0.0
         BlockMinutes =
sensor#
```

Step 11 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 12 Display the statistics for the SDEE server.

```
sensor# show statistics sdee-server
General
   Open Subscriptions = 0
   Blocked Subscriptions = 0
   Maximum Available Subscriptions = 5
   Maximum Events Per Retrieval = 500
Subscriptions
sensor#
```

Step 13 Display the statistics for the transaction server.

```
sensor# show statistics transaction-server
General
   totalControlTransactions = 35
   failedControlTransactions = 0
sensor#
```

Step 14 Display the statistics for a virtual sensor.

```
sensor# show statistics virtual-sensor vs0
   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 =
      General Statistics for this Virtual Sensor
         Number of seconds since a reset of the statistics = 1421711
         Measure of the level of resource utilization = 0
         Total packets processed since reset = 0
         Total IP 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 =
         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 packets with bad IP checksums processed since reset = 0
         Total packets with bad layer 4 checksums processed since reset = 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.


      The Signature Database Statistics.
         The Number of each type of node active in the system (can not be reset
            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 constraint
      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 dgrams limit since last reset = 0
   Fragments too small since last reset = 0
   Too many fragments per dgram 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
   Current Streams = 0
   Current Streams Closed = 0
   Current Streams Closing = 0
   Current Streams Embryonic = 0
   Current Streams Established = 0
   Current Streams Denied = 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
         SigEvent Action Override Stage Statistics
            Number of Alerts received to Action Override Processor = 0
            Number of Alerts where an override was applied = 0
            Actions Added
               deny-attacker-inline = 0
               deny-attacker-victim-pair-inline = 0
               deny-attacker-service-pair-inline = 0
               deny-connection-inline = 0
               deny-packet-inline = 0
               modify-packet-inline = 0
               log-attacker-packets = 0
               log-pair-packets = 0
               log-victim-packets = 0
               produce-alert = 0
               produce-verbose-alert = 0
               request-block-connection = 0
               request-block-host = 0
               request-snmp-trap = 0
               reset-tcp-connection = 0
               request-rate-limit = 0
         SigEvent Action Filter Stage Statistics
            Number of Alerts received to Action Filter Processor = 0
            Number of Alerts where an action was filtered = 0
            Number of Filter Line matches = 0
            Number of Filter Line matches causing decreased DenyPercentage = 0
            Actions Filtered
               deny-attacker-inline = 0
               deny-attacker-victim-pair-inline = 0
               deny-attacker-service-pair-inline = 0
               deny-connection-inline = 0
               deny-packet-inline = 0
               modify-packet-inline = 0
               log-attacker-packets = 0
               log-pair-packets = 0
               log-victim-packets = 0
               produce-alert = 0
               produce-verbose-alert = 0
               request-block-connection = 0
               request-block-host = 0
               request-snmp-trap = 0
               reset-tcp-connection = 0
               request-rate-limit = 0
         SigEvent Action Handling Stage Statistics.
            Number of Alerts received to Action Handling Processor = 0
            Number of Alerts where produceAlert was forced = 0
            Number of Alerts where produceAlert was off = 0
            Actions Performed
               deny-attacker-inline = 0
               deny-attacker-victim-pair-inline = 0
               deny-attacker-service-pair-inline = 0
               deny-connection-inline = 0
               deny-packet-inline = 0
               modify-packet-inline = 0
               log-attacker-packets = 0
               log-pair-packets = 0
               log-victim-packets = 0
               produce-alert = 0
               produce-verbose-alert = 0
--MORE--
```

Step 15 Display the statistics for Web Server.

```
sensor# show statistics web-server
listener-443
   number of server session requests handled = 61
   number of server session requests rejected = 0
   total HTTP requests handled = 35
   maximum number of session objects allowed = 40
   number of idle allocated session objects = 10
   number of busy allocated session objects = 0
crypto library version = 6.0.3
sensor#
```

Step 16 Clear the statistics for an application, for example, the logging application.

```
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
```

The statistics were retrieved and cleared.

Step 17 Verify that the statistics have been cleared.

```
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#
```

The statistics all begin from 0.

Displaying Tech Support Information

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.

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.

```
sensor# show tech-support page
```

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.

Step 3 To send the output (in HTML format) to a file, follow these steps:

a. Enter the following command, followed by a valid destination.

```
sensor# show tech-support destination-url destination_url
```

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`.

For 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
```

The `password:` prompt appears.

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.

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 6.1(1)E1

Host:
    Realm Keys          key1.0
Signature Definition:
    Signature Update    S323.0                  2008-03-24
    Virus Update        V1.2                    2005-11-24
OS Version:             2.4.30-IDS-smp-bigphys
Platform:               IPS-4240-K9
Serial Number:          P3000000652
No license present
Sensor up-time is 4 days.
Using 1421475840 out of 1984548864 bytes of available memory (71% usage)
system is using 17.7M out of 29.0M bytes of available disk space (61% usage)
application-data is using 41.0M out of 166.8M bytes of available disk space (26%
 usage)
boot is using 40.4M out of 68.6M bytes of available disk space (62% usage)


MainApp          M-2008_APR_16_21_44   (Release)   2008-04-16T22:25:36-0500   Running
AnalysisEngine   M-2008_APR_16_21_44   (Release)   2008-04-16T22:25:36-0500   Running
CLI              M-2008_APR_16_21_44   (Release)   2008-04-16T22:25:36-0500


Upgrade History:

  IPS-K9-6.1-1-E1   21:44:00 UTC Wed Apr 16 2008

Recovery Partition Version 1.1 - 6.1(1)E1

Host Certificate Valid from: 23-Apr-2008 to 24-Apr-2010

sensor#
```

Note If the `--MO RE--` 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 Thu Apr 24 16:21:25 2008
! ------------------------------
! Version 6.1(1)
! Host:
!     Realm Keys          key1.0
! Signature Definition:
!     Signature Update    S323.0   2008-03-24
!     Virus Update        V1.2     2005-11-24
! ------------------------------
service interface
exit
! ------------------------------
service authentication
exit
! ------------------------------
service event-action-rules rules0
exit
! ------------------------------
service host
network-settings
host-ip 10.89.147.45/25,10.89.147.126
telnet-option enabled
access-list 0.0.0.0/0
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 analysis-engine
exit
sensor#
```

Diagnosing Network Connectivity

Use the **ping** *ip_address* [**count**] command to diagnose basic network connectivity.

Caution No command interrupt is available for this command. It must run to completion.

To diagnose basic network connectivity, follow these steps:

Step 1 Log in to the CLI.

Step 2 Ping the address you are interested in.

```
sensor# ping ip_address count
```

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.

Example of a successful ping:

```
sensor# ping 10.89.146.110 6
PING 10.89.146.110 (10.89.146.110): 56 data bytes
64 bytes from 10.89.146.110: icmp_seq=0 ttl=61 time=0.3 ms
64 bytes from 10.89.146.110: icmp_seq=1 ttl=61 time=0.1 ms
64 bytes from 10.89.146.110: icmp_seq=2 ttl=61 time=0.1 ms
64 bytes from 10.89.146.110: icmp_seq=3 ttl=61 time=0.2 ms
64 bytes from 10.89.146.110: icmp_seq=4 ttl=61 time=0.2 ms
64 bytes from 10.89.146.110: icmp_seq=5 ttl=61 time=0.2 ms

--- 10.89.146.110 ping statistics ---
6 packets transmitted, 6 packets received, 0% packet loss
round-trip min/avg/max = 0.1/0.1/0.3 ms
```

Example of an unsuccessful ping:

```
sensor# ping 172.21.172.1 3
PING 172.21.172.1 (172.21.172.1): 56 data bytes

--- 172.21.172.1 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, follow 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 To 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 modules, see the following individual procedures:

- Rebooting, Resetting, and Shutting Down the AIM IPS, page 17-17
- Reloading, Shutting Down, Resetting, and Recovering the AIP SSM, page 18-14
- Resetting the IDSM2, page 19-41
- Rebooting, Resetting, and Shutting Down the NME IPS, page 20-11

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.

```
sensor# show history
clear line
configure terminal
show history
```

Step 3 Show the history of the commands you have used in network access mode.

```
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.

PEP information provides an easy way to obtain the hardware version and serial number through the CLI.

Note The **show inventory** command does not apply to IPS modules, such as the AIM IPS, AIP SSM, IDSM2, or NME IPS.

To display PEP information, follow these steps:

Step 1 Log in to the CLI.

Step 2 Display the PEP information.

```
sensor# show inventory

Name: "Chassis", DESCR: "IPS 4255 Intrusion Prevention Sensor"
PID: IPS-4255-K9, VID: V01 , SN: JAB0815R017

Name: "Power Supply", DESCR: ""
PID: ASA-180W-PWR-AC, VID: V01 , SN: 123456789AB
sensor#

sensor# show inventory

Name: "Module", DESCR: "ASA 5500 Series Security Services Module-20"
PID: ASA-SSM-20, VID: V01 , SN: JAB0815R036
sensor#

sensor-4240# show inventory

Name: "Chassis", DESCR: "IPS 4240 Appliance Sensor"
PID: IPS-4240-K9, VID: V01 , SN: P3000000653
sensor-4240#
```

You can use this information when dealing with the TAC.

Tracing the Route of an IP Packet

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.

Caution There is no command interrupt available for this command. It must run to completion.

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 10.1.1.1
traceroute to 10.1.1.1 (10.1.1.1), 4 hops max, 40 byte packets
 1  10.89.130.1 (10.89.130.1)  0.267 ms  0.262 ms  0.236 ms
 2  10.89.128.17 (10.89.128.17)  0.24 ms *  0.399 ms
 3  * 10.89.128.17 (10.89.128.17)  0.424 ms *
 4  10.89.128.17 (10.89.128.17)  0.408 ms *  0.406 ms
sensor#
```

Step 3 To have the route take more hops than the default of 4, use the *count* option.

```
sensor# trace 10.1.1.1 8
traceroute to 10.1.1.1 (10.1.1.1), 8 hops max, 40 byte packets
 1  10.89.130.1 (10.89.130.1)  0.35 ms  0.261 ms  0.238 ms
 2  10.89.128.17 (10.89.128.17)  0.36 ms *  0.344 ms
 3  * 10.89.128.17 (10.89.128.17)  0.465 ms *
 4  10.89.128.17 (10.89.128.17)  0.319 ms *  0.442 ms
 5  * 10.89.128.17 (10.89.128.17)  0.304 ms *
 6  10.89.128.17 (10.89.128.17)  0.527 ms *  0.402 ms
 7  * 10.89.128.17 (10.89.128.17)  0.39 ms *
 8  10.89.128.17 (10.89.128.17)  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: 10.89.147.61
   ---------------------------------------------
      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: 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: 2)
-----------------------------------------------
   ip-address: 10.89.147.10
   ---------------------------------------------
```

```
            communication: telnet default: ssh-3des
            nat-address: 0.0.0.0 <defaulted>
            profile-name: insidePix
         ---------------------------------------------
         ip-address: 10.89.147.82
         ---------------------------------------------
            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: 10.89.147.61
      ---------------------------------------------
      router-devices (min: 0, max: 250, current: 1)
      ---------------------------------------------
         ip-address: 10.89.147.54
      ---------------------------------------------
      firewall-devices (min: 0, max: 250, current: 2)
```

```
   ----------------------------------------------
      ip-address: 10.89.147.10
      ip-address: 10.89.147.82
   ----------------------------------------------
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: 10.89.147.61
         profile-name: cat
      ip-address: 10.89.147.54
         profile-name: r7200
      ip-address: 10.89.147.10
         profile-name: insidePix
      ip-address: 10.89.147.82
         profile-name: test
sensor(config-net)#
```

CHAPTER 17

Configuring the AIM IPS

Note All IPS platforms allow ten concurrent log in sessions.

This chapter describes how to configure the AIM IPS and get it ready to receive IPS traffic. After that you are ready to configure intrusion prevention. It contains the following sections:

AIM IPS Configuration Sequence

Perform the following tasks to configure the AIM IPS:

1. Set up the interfaces.
2. Log in to the AIM IPS.
3. Initialize the AIM IPS.

   Run the **setup** command to initialize the AIM IPS.
4. Create the service account.

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 new password if the administrator password is lost. Analyze your situation to decide if you want a service account existing on the system.

5. Perform the other initial tasks, such as adding users, trusted hosts, and so forth.
6. Configure intrusion prevention.
7. Perform administrative tasks to keep your AIM IPS running smoothly.
8. Upgrade the IPS software with new signature updates and service packs.
9. Reimage the boot helper and bootloader when needed.

For More Information

- For the procedure for setting up interfaces, see Setting Up Interfaces on the AIM IPS and the Router, page 17-4.
- For the procedure for logging in to the AIM IPS, see Establishing Sessions, page 17-13.
- For the procedure for running the **setup** command on the AIM IPS, see Advanced Setup for the AIM IPS, page 3-12.
- For the procedure for creating the service account, see Creating the Service Account, page 4-14.
- For the procedures for setting up the sensor, see Chapter 4, "Setting Up the Sensor."
- For the procedures for configuring intrusion prevention, see Chapter 9, "Configuring Anomaly Detection," Chapter 7, "Configuring Event Action Rules," Chapter 8, "Defining Signatures," and Chapter 13, "Configuring Attack Response Controller for Blocking and Rate Limiting."
- For the procedures to keep your sensor running smoothly, see Chapter 16, "Administrative Tasks for the Sensor."
- For more information on obtaining Cisco IPS software, see Obtaining Cisco IPS Software, page 21-1.
- For the procedure for reimaging the AIM IPS, see Installing the AIM IPS System Image, page 22-21.

Verifying Installation and Finding the Serial Number

Use the **show inventory** command in privileged EXEC mode to verify the installation of the AIM IPS.

Note You can also use this command to find the serial number of your AIM IPS for use in troubleshooting with TAC. The serial number appears in the PID line, for example, SN: FOC11372M9X.

To verify the installation of the AIM IPS, follow these steps:

Step 1 Log in to the router.

Step 2 Enter privileged EXEC mode on the router.

```
router> enable
```

Step 3 Verify that the AIM IPS is part of the router inventory.

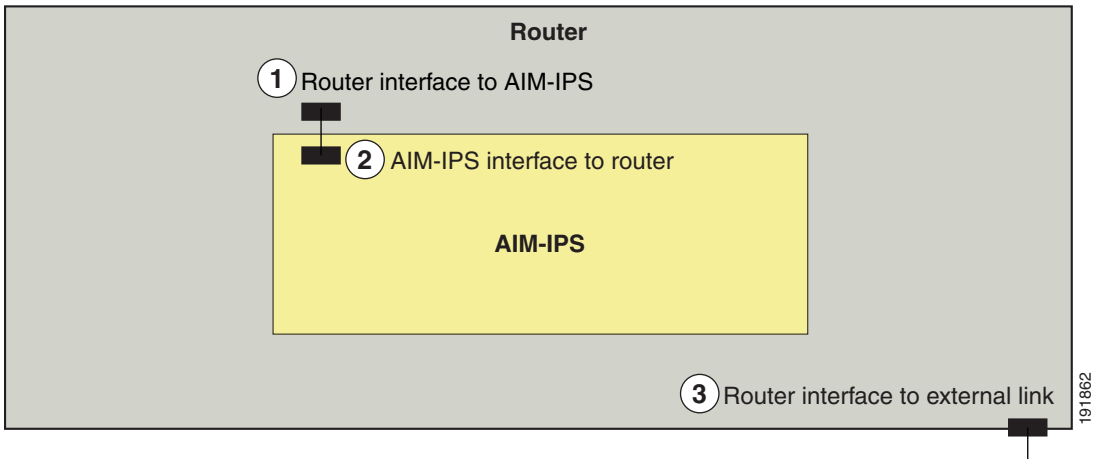
```
router# show inventory
NAME: "3825 chassis", DESCR: "3825 chassis"
PID: CISCO3825 , VID: V01 , SN: FTX1009C3KT
```

```
NAME: "Cisco Intrusion Prevention System AIM in AIM slot: 1", DESCR: "Cisco Intrusion
Prevention"
PID: AIM-IPS-K9 , VID: V01 , SN: FOC11372M9X

router#
```

Understanding the Hardware Interfaces

Figure 17-1 shows the router and AIM IPS interfaces used for internal communication. You can configure the router interfaces through the Cisco IOS CLI and the AIM IPS interfaces through the IPS CLI, IDM, IME, or CSM.

Figure 17-1 AIM IPS and Router Interfaces



1	Router interface to the AIM IPS (IDS-Sensor 0/1) Uses the Cisco OS CLI to configure the IP address of the router interface that connects to the AIM IPS. This router IP address is used as the default router IP address when you configure Cisco IPS on the AIM IPS.
2	The AIM IPS interface to router (GigabitEthernet0/1) Configure the command and control interface using the IPS CLI, IDM, IME, or CSM.
3	Router interface to external link.

Note You need two IP addresses to configure the AIM IPS. The AIM IPS has a command and control IP address that you configure through the Cisco IPS CLI. You also assign an IP address to the router for its internal interface (IDS-Sensor 0/x) to the AIM IPS. This IP address belongs to the router itself and is used for routing traffic to the command and control interface of the AIM IPS. It is used as the default router IP address when you set up the AIM IPS command and control interface.

Setting Up Interfaces on the AIM IPS and the Router

This section describes how to set up interfaces on the AIM IPS and the router, and contains the following topics:

Interface Configuration Sequence

Follow this sequence to set up interfaces on the AIM IPS and the router:

1. Configure the IPS command and control interface on the router, and the AIM IPS IP address, mask, and gateway using one of the following methods:
   - An unnumbered IP address on the IDS-Sensor interface

     Note Using an unnumbered IP address on the IDS-Sensor interface is the preferred method for configuring interfaces on the module and router.

   - A routable IP address
   - Default module IP address with NAT
   - User-configured IP address with NAT
2. Enable the monitoring interface and specify whether it is promiscuous or inline, assign the ACL to the interface, specify how you want the router to handle traffic if the module fails, and create a monitoring ACL (optional).
3. Save the configuration.

For More Information

- For the procedure for configuring an unnumbered IP address on the IDS-Sensor interface, see Using an Unnumbered IP Address Interface, page 17-5.
- For the procedure for configuring a routable IP address, see Using a Routable IP Address Interface, page 17-7.
- For the procedure for configuring the default module IP address using NAT, see Using a Default IP Address and NAT, page 17-9.
- For the procedure for configuring the IP address with NAT, see Using a User-Configured IP Address and NAT, page 17-10.
- For the procedure for enabling the monitoring interface, see Configuring Monitoring on the Router Interface, page 17-12.

ARC and NAT

If you use NAT to establish management access to the AIM IPS, ARC on the AIM IPS does not know the external IP address of the AIM IPS. To make sure that management access to the AIM IPS is not interrupted by devices that the AIM IPS is managing, you must state the NAT address of the AIM IPS every time you add a blocking device.

For More Information

- For more information on ARC, see Chapter 13, "Configuring Attack Response Controller for Blocking and Rate Limiting."
- For the procedures for configuring the AIM IPS NAT address every time you add a blocking device, see the following procedures:
  - Configuring the Sensor to Manage Cisco Routers, page 13-22
  - Configuring the Sensor to Manage Catalyst 6500 Series Switches and Cisco 7600 Series Routers, page 13-25
  - Configuring the Sensor to Manage Cisco Firewalls, page 13-27

Using an Unnumbered IP Address Interface

Note Using an unnumbered IP address on the IDS-Sensor interface is the preferred method for configuring interfaces on the AIM IPS and the router.

To configure the interface using an unnumbered IP address interface, follow these steps:

Step 1 Log in to the router.

Step 2 Enter privileged EXEC mode on the router.

```
router> enable
```

Step 3 Confirm the module slot number in your router.

```
router# show run | include ids-sensor
interface IDS-Sensor0/1
router#
```

Step 4 Configure the IPS command and control interface on the router using the **ip unnumbered** command on the IDS-Sensor interface to specify the router interface that provides external connectivity:

a. Make sure the IDS-Sensor interface is not shut down.

```
router# configure terminal
router(config)# interface ids-sensor 0/1
router(config-if)# no shutdown
```

b. Specify the external router interface.

```
router(config-if)# ip unnumbered other_router_interface
router(config-if)# exit
router(config)#
```

Note The IDS-Sensor interface shares the IP address between the two router interfaces (the IDS-Sensor interface and the other specified interface).

Note The IP address of the sensor and the *other_router_interface* IP address must be on the same subnet.

c. Enter a route to send traffic to the IP address of the AIM IPS to the IDS-Sensor interface.

```
router(config)# ip route sensor_ip_address 255.255.255.255 ids-sensor 0/1
router(config)#
```

d. Exit configuration mode.

```
router(config)# exit
router#
```

Step 5 Configure the IP address, mask, and gateway:

Note You can also configure these parameters by initializing the AIM IPS with the **setup** command.

Note The AIM IPS IP address defaults to 192.168.1.2/24,192.168.1.1.

a. Session to the AIM IPS.

```
router# service-module ids-sensor 0/1 session
Trying 192.168.1.2, 2322 ... Open


sensor login:
```

b. Log in to the CLI.

c. Enter global configuration mode.

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

d. Enter service host mode.

```
sensor(config)# service host
sensor(config-hos)#
```

e. Assign the command and control interface and the gateway.

```
sensor(config-hos)# network-settings
sensor(config-hos-net)# host-ip ip_address/mask,gateway
sensor(config-hos-net)#
```

Note The gateway should be the IP address of the *other_router_interface* that you set up in Step 4b.

f. Exit network settings mode.

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

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

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

h. Exit the session to the AIM IPS.

Step 6 Write the configuration to NVRAM.

```
router# write memory
Building configuration
[OK]
```

For More Information

- For the procedure for using the **setup** command to initialize the AIM IPS, see Advanced Setup for the AIM IPS, page 3-12.
- For more information on sessioning from the router to the AIM IPS and exiting sessions, see Opening and Closing a Session, page 17-14.

Using a Routable IP Address Interface

To configure the interface using a routable IP address interface, follow these steps:

Step 1 Log in to the router.

Step 2 Enter privileged EXEC mode on the router.

```
router> enable
```

Step 3 Confirm the module slot number in your router.

```
router# show run | include ids-sensor
interface IDS-Sensor0/1
router#
```

Step 4 Configure the IPS command and control interface on the router using the **ip unnumbered** command on the IDS-Sensor interface to specify the router interface that provides external connectivity:

a. Make sure the IDS-Sensor interface is not shut down.

```
router# configure terminal
router(config)# interface ids-sensor 0/1
router(config-if)# no shutdown
```

b. Configure an IP address for the IDS-Sensor interface.

```
router(config-if)# ip address 192.168.1.2 255.255.255.0
router(config0if)#
```

Use 192.168.1.2 (default IP address for the default gateway on the AIM IPS). You cannot session to the AIM IPS if its interface does not have an IP address.

c. Enter a route to send traffic to the IP address of the AIM IPS to the IDS-Sensor interface.

```
router(config)# ip route sensor_ip_address 255.255.255.255 ids-sensor 0/1
router(config)#
```

d. Exit configuration mode.

```
router(config)# exit
router#
```

Step 5 Configure the AIM IPS IP address, mask, and gateway:

Note The AIM IPS IP address defaults to 192.168.1.2/24,192.168.1.1.

a. Session to the AIM IPS.

```
router# service-module ids-sensor 0/1 session
Trying 192.168.1.2, 2322 ... Open


sensor login:
```

b. Log in to the CLI.

c. Enter global configuration mode.

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

d. Enter service host mode.

```
sensor(config)# service host
sensor(config-hos)#
```

e. Assign the command and control interface and the gateway.

```
sensor(config-hos)# network-settings
sensor(config-hos-net)# host-ip ip_address/mask,gateway
sensor(config-hos-net)#
```

f. Exit network settings mode:

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

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

h. Exit the session to the AIM IPS.

Step 6 Write the configuration to NVRAM.

```
router# write memory
Building configuration
[OK]
```

For More Information

- For the procedure for using the **setup** command to initialize the AIM IPS, see Advanced Setup for the AIM IPS, page 3-12.
- For more information on sessioning from the router to the AIM IPS and exiting sessions, see Opening and Closing a Session, page 17-14.

Using a Default IP Address and NAT

To configure the interfaces using the default IP address and NAT, follow these steps:

Step 1 Log in to the router.

Step 2 Enter privileged EXEC mode on the router.

```
router> enable
```

Step 3 Confirm the module slot number in your router.

```
router# show run | include ids-sensor
interface IDS-Sensor0/1
router#
```

Step 4 Configure the IPS command and control interface on the router using the default sensor IP address and have the router perform NAT:

a. Make sure the IDS-Sensor interface is not shut down.

```
router# configure terminal
router(config)# interface ids-sensor 0/1
router(config-if)# no shutdown
```

b. Configure an IP address for the IDS-Sensor interface (should match the default gateway that is configured on the AIM IPS).

```
router(config-if)# ip address 192.168.1.2 255.255.255.0
router(config0if)#
```

Use 192.168.1.2 (default IP address for the default gateway on the AIM IPS). You cannot session to the AIM IPS if its interface does not have an IP address.

c. Set up a NAT address for the AIM IPS (the AIM IPS default IP address is 192.168.1.2).

```
router(config-if)# ip nat inside
router(config-if)# exit
router(config)# interface router_nat_outside_interface
router(config-if)# ip nat outside
router(config-if)# exit
router(config)# ip nat inside source static 192.168.1.2 aim_external_ip_address
router(config-if)# exit
```

Note The *aim_external_ip_address* and the *router_nat_outside_interface* IP addresses must be on the same subnet. The IP address of the AIM IPS must be on a separate subnet.

d. Exit configuration mode.

```
router(config-if)# exit
router(config)# exit
router#
```

Step 5 Configure the AIM IPS IP address, mask, and gateway:

Note The AIM IPS IP address defaults to 192.168.1.2/24,192.168.1.1.

a. Session to the AIM IPS.

```
router# service-module ids-sensor 0/1 session
Trying 192.168.1.2, 2322 ... Open


sensor login:
```

b. Log in to the CLI.

c. Enter global configuration mode.

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

d. Enter service host mode.

```
sensor(config)# service host
sensor(config-hos)#
```

e. Assign the command and control interface and the gateway.

```
sensor(config-hos)# network-settings
sensor(config-hos-net)# host-ip ip_address/mask,gateway
sensor(config-hos-net)#
```

f. Exit network settings mode.

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

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

h. Exit the session to the AIM IPS.

Step 6 Write the configuration to NVRAM.

```
router# write memory
Building configuration
[OK]
```

For More Information

- For the procedure for using the **setup** command to initialize the AIM IPS, see Advanced Setup for the AIM IPS, page 3-12.
- For more information on sessioning from the router to the AIM IPS and exiting sessions, see Opening and Closing a Session, page 17-14.
- For more information on how ARC and NAT operate on the AIM IPS, see ARC and NAT, page 17-5.

Using a User-Configured IP Address and NAT

To configure the interfaces using a user-configured IP address and NAT, follow these steps:

Step 1 Log in to the router.

Step 2 Enter privileged EXEC mode on the router.

```
router> enable
```

Step 3 Confirm the module slot number in your router.

```
router# show run | include ids-sensor
interface IDS-Sensor0/1
router#
```

Step 4 Configure the IPS command and control interface on the router using the default sensor IP address and have the router perform NAT:

a. Make sure the IDS-Sensor interface is not shut down.

```
router# configure terminal
router(config)# interface ids-sensor 0/1
router(config-if)# no shutdown
```

b. Configure an IP address for the IDS-Sensor interface.

```
router(config-if)# ip address user_configured_ip_address gateway
router(config0if)#
```

You cannot session to the AIM IPS if its interface does not have an IP address.

c. Set up a NAT address for the AIM IPS.

```
router(config-if)# ip nat inside
router(config-if)# exit
router(config)# interface router_nat_outside_interface
router(config-if)# ip nat outside
router(config-if)# exit
router(config)# ip nat inside source static AIM_ip_address AIM_external_ip_address
router(config-if)# exit
```

d. Exit configuration mode.

```
router(config-if)# exit
router(config)# exit
router#
```

Step 5 Configure the AIM IPS IP address, mask, and gateway:

Note The AIM IPS IP address defaults to 192.168.1.2/24,192.168.1.1.

a. Session to the AIM IPS.

```
router# service-module ids-sensor 0/1 session
Trying 192.168.1.2, 2322 ... Open


sensor login:
```

b. Log in to the CLI.

c. Enter global configuration mode.

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

d. Enter service host mode.

```
sensor(config)# service host
sensor(config-hos)#
```

e. Assign the command and control interface and the gateway.

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

```
sensor(config-hos-net)# host-ip ip_address/mask,gateway
sensor(config-hos-net)#
```

f. Exit network settings mode.

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

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

h. Exit the session to the router.

Step 6 Write the configuration to NVRAM.

```
router# write memory
Building configuration
[OK]
```

For More Information

- For the procedure for using the **setup** command to initialize the AIM IPS, see Advanced Setup for the AIM IPS, page 3-12.
- For more information on sessioning from the router to the AIM IPS and exiting sessions, see Opening and Closing a Session, page 17-14.
- For more information on how ARC and NAT operate on the AIM IPS, see ARC and NAT, page 17-5.

Configuring Monitoring on the Router Interface

Note You must add the AIM IPS internal interface to the virtual sensor (vs0) so that traffic can be monitored.

To configure the router interface to be monitored, follow these steps:

Step 1 Log in to the router.

Step 2 Enter privileged EXEC mode on the router.

```
router> enable
```

Step 3 (Optional) Configure a monitoring access list on the router.

```
router(config)# access-list 101 permit tcp any eq www any
```

You can set up a standard access list and apply it to filter what type of traffic you want to inspect. A matched ACL causes traffic not to be inspected for that ACL. This example bypasses inspection of HTTP traffic only. Refer to your Cisco IOS Command Reference for more information on the options for the **access-list** command.

Step 4 Enable monitoring on the interface in either inline or promiscuous mode and associate the access list.

```
router(config)# interface monitored_interface
router(config-if)# ids-service-module monitoring [inline | promiscuous] access-list 101
router(config-if)# exit
router(config)#
```

Note Associating the access list with the interface further controls what traffic is sent to the AIM IPS.

Step 5 (For inline mode) Confirm the module slot number in your router.

```
router# show run | include ids-sensor
interface IDS-Sensor0/1
router#
```

Step 6 (For inline mode) Specify how the router handles traffic inspection during a module failure.

```
router(config)# interface ids-sensor 0/1
router(config-if)# service-module [fail-close | fail-open]
router(config-if)#
```

The default is fail-open.

Note The **fail-close** option means that if the AIM IPS fails, then the router does not let traffic pass. The **fail-open** option means if the AIM IPS fails, the router lets traffic pass, but it is not inspected by the IPS.

Step 7 Exit configuration mode.

```
router(config-if)# exit
router(config)# exit
router#
```

Step 8 Write the configuration to NVRAM.

```
router# write memory
Building configuration
[OK]
```

For More Information

- For more information on promiscuous mode, see Configuring Promiscuous Mode, page 5-15.
- For more information on inline mode, see Configuring Inline Interface Mode, page 5-16.
- For the procedure for adding the AIM IPS internal interface to the virtual sensor (vs0) so that traffic can be monitored, see Advanced Setup for the AIM IPS, page 3-12.

Establishing Sessions

Because the AIM IPS does not have an external console port, console access to the AIM IPS is enabled when you issue the **service-module ids-sensor** *slot/port* **session** command on the router, or when you initiate a Telnet connection into the router with the slot number corresponding to the AIM IPS port number. The lack of an external console port means that the initial bootup configuration is possible only through the router.

When you issue the **service-module ids-sensor** *slot/port* **session** command, you create a console session with the AIM IPS, in which you can issue any IPS configuration commands. After completing work in the session and exiting the IPS CLI, you are returned to the Cisco IOS CLI.

The **session** command starts a reverse Telnet connection using the IP address of the IDS-Sensor interface. The IDS-Sensor interface is an interface between the AIM IPS and the router. You must assign an IP address to the IDS-Sensor interface before invoking the **session** command. Assigning a routable IP address can make the IDS-Sensor interface itself vulnerable to attacks, because the AIM IPS is visible on the network through that routable IP address, meaning you can communicate with the AIM IPS outside the router. To counter this vulnerability, assign an unnumbered IP address to the IDS-Sensor interface. Then the AIM IPS IP address is only used locally between the router and the AIM IPS, and is isolated for the purposes of sessioning in to the AIM IPS.

Note Before you install your application software or reimage the module, opening a session brings up the bootloader. After you install the software, opening a session brings up the application.

Caution If you session to the module and perform large console transfers, character traffic may be lost unless the host console interface speed is set to 115200/bps or higher. Use the **show running config** command to check that the speed is set to 115200/bps.

For More Information

For the procedure for setting up an unnumbered IP address, see Using an Unnumbered IP Address Interface, page 17-5.

Opening and Closing a Session

Note You must initialize the AIM IPS (run the **setup** command) from the router. After networking is configured, SSH and Telnet are available.

Use the **service-module ids-sensor** *slot/port* **session** command to establish a session from the AIM IPS to the module. Press **Ctrl-Shift-6**, then **x**, to return a session prompt to a router prompt, that is, to go from the AIM IPS prompt back to the router prompt. Press **Enter** on a blank line to go back to the session prompt, which is also the router prompt. You should only suspend a session to the router if you will be returning to the session after executing router commands. If you do not plan on returning to the AIM IPS session, you should close the session rather than suspend it.

When you close a session, you are logged completely out of the AIM IPS CLI and a new session connection requires a username and password to log in. A suspended session leaves you logged in to the CLI. When you connect with the **session** command, you can go back to the same CLI without having to provide your username and password.

Note Telnet clients vary. In some cases, you may have to press **Ctrl-6 + x**. The control character is specified as ^^, Ctrl-^, or ASCII value 30 (hex 1E).

Caution If you use the **disconnect** command to leave the session, the session remains running. The open session can be exploited by someone wanting to take advantage of a connection that is still in place.

To open and close sessions to the AIM IPS, follow these steps:

Step 1 Log in to the router.

Step 2 Check the status of the AIM IPS to make sure it is running.

```
router# service-module ids-sensor 0/1 status
Service Module is Cisco IDS-Sensor0/1
Service Module supports session via TTY line 322
Service Module is in Steady state
Getting status from the Service Module, please wait..
Cisco Systems Intrusion Prevention System Network Module
  Software version:  6.1(1)E1
  Model:             AIM-IPS
  Memory:            443508 KB
  Mgmt IP addr:      10.89.148.196
  Mgmt web ports:    443
  Mgmt TLS enabled:  true


router#
```

Step 3 Open a session from the router to the AIM IPS.

```
router# service-module ids-sensor 0/1 session
Trying 10.89.148.196, 2322 ... Open
```

Step 4 Exit, or suspend and close the module session:

- `sensor# exit`

Note If you are in submodes of the IPS CLI, you must exit all submodes. Enter **exit** until the sensor login prompt appears.

Caution Failing to close a session properly makes it possible for others to exploit a connection that is still in place. Remember to enter **exit** at the `router#` prompt to close the Cisco IOS session completely.

- To suspend and close the session to the AIM IPS, press **Ctrl-Shift** and press **6**. Release all keys, and then press **x**.

Note When you are finished with a session, you need to return to the router to establish the association between a session (the IPS application) and the router interfaces you want to monitor.

Step 5 Disconnect from the router.

```
router# disconnect
```

Step 6 Press **Enter** to confirm the disconnection.

```
router# Closing connection to 10.89.148.196 [confirm] <Enter>
```

For More Information

For the procedure for initializing the AIM IPS, see Advanced Setup for the AIM IPS, page 3-12.

Displaying the Status of the AIM IPS

Use the **service-module ids-sensor** *slot/port* **status** command in privileged EXEC mode to display the status and statistics of the AIM IPS.

To display the status of the AIM IPS, follow these steps:

Step 1 Log in to the router.

Step 2 Enter privileged EXEC mode on the router.

```
router> enable
```

Step 3 Display the status of the AIM IPS.

```
router# service-module ids-sensor 0/1 status
Service Module is Cisco IDS-Sensor0/1
Service Module supports session via TTY line 322
Service Module is in Steady state
Service Module is in fail close
Cisco Systems Intrusion Prevention System Network Module
  Software version:  6.1(1)E2
  Model:             AIM-IPS
  Memory:            443508 KB
  Mgmt IP addr:      10.89.148.196
  Mgmt web ports:    443
  Mgmt TLS enabled:  true


router#
```

Enabling and Disabling Heartbeat Reset

Use the **service-module ids-sensor** *slot/port* **heartbeat reset {enable | disable}** command in privileged EXEC mode to reset the heartbeat of the AIM IPS.

When the AIM IPS is booted in failsafe mode or is undergoing an upgrade, you can use the **service-module ids heartbeat-reset** command to prevent a reboot during the process. If you leave the heartbeat reset enabled during an upgrade, you may lose the AIM IPS heartbeat.

When the AIM IPS heartbeat is lost, the router applies a fail-open or fail-close configuration option to the AIM IPS and stops sending traffic to the AIM IPS, and sets the AIM IPS to error state. The router performs a hardware reset on the AIM IPS and monitors the AIM IPS until the heartbeat is reestablished.

Note Disabling the heartbeat reset prevents the router from resetting the module during system image installation if the process takes too long.

To reset the heartbeat of the AIM IPS, follow these steps:

Step 1 Log in to the router.

Step 2 Enter privileged EXEC mode on the router.

```
router> enable
```

Step 3 Verify the status of heartbeat reset.

```
router# service-module ids-sensor 0/1 status
Service Module is Cisco IDS-Sensor 0/1
Service Module supports session via TTY line 194
Service Module heartbeat-reset is enabled
```

Step 4 To disable the heartbeat on the AIM IPS:

```
router# service-module ids-sensor 0/1 heartbeat-reset disable
```

Step 5 To reenable the heartbeat on the AIM IPS:

```
router# service-module ids-sensor 0/1 heartbeat-reset enable
```

Rebooting, Resetting, and Shutting Down the AIM IPS

This section describes when and how the AIM IPS shuts down. It contains the following topics:

AIM IPS Status Monitoring

The AIM IPS uses RBCP to monitor its status. RBCP is monitored by the main application on the AIM IPS, not by SensorApp. If the main application on the AIM IPS fails, the RBCP heartbeat responses do not return from the AIM IPS. When the router determines that the AIM IPS has failed, a **reload** command is issued through RBCP to reboot the Linux kernel on the AIM IPS. In the period during the attempt to bring the AIM IPS back up, the router works in the mode determined by the failover operation configured.

In some cases, SensorApp may stop processing, but the main application on the AIM IPS continues to process RBCP packets. In this case, packets are processed according to the bypass settings set for the AIM IPS by the IPS CLI, IDM, or IME.

There are two situations in which the AIM IPS shuts down:

- A hardware or software error forces it to fail. The router can detect this through the loss of the RBCP heartbeat.
- **Reload** or **shutdown** command.

For More Information

- For more information on SensorApp, see SensorApp, page A-22.
- For more information on software bypass, see Configuring Inline Bypass Mode, page 5-33.

Rebooting, Resetting, and Shutting Down the AIM IPS

Use the **service-module ids-sensor** *slot/port* [**reload | reset | shutdown**] command in privileged EXEC mode to reboot, reset, and shut down the AIM IPS.

To reboot, reset, and shut down the AIM IPS, follow these steps:

Step 1 Log in to the router.

Step 2 Enter privileged EXEC mode on the router.

```
router> enable
```

Step 3 To gracefully halt and reboot the operating system on the AIM IPS:

```
router# service-module ids-sensor 0/1 reload
Do you want to proceed with the reload? [confirm]
```

Step 4 To reset the hardware on the AIM IPS:

```
router# service-module ids-sensor 0/1 reset
Use reset only to recover from shutdown or failed state
Warning: May lose data on the hard disc!
Do you want to reset?[confirm]
```

Note The AIM IPS has a compact flash device that functions as a permanent storage device rather than a hard-disk drive.

Caution Data loss occurs only if you issue the **reset** command without first shutting down the AIM IPS. You can use the **reset** command safely in other situations.

Step 5 To shut down applications running on the AIM IPS:

```
router# service-module ids-sensor 0/1 shutdown
Trying 10.10.10.1, 2129 . . . Open
%SERVICEMODULE-5-SHUTDOWN2:Service module IDS-Sensor1/0 shutdown complete
```

New and Modified Commands

Note All other Cisco IOS software commands are documented in the Cisco IOS Release 12.4(20)T command reference at Cisco.com, http://www.cisco.com/en/US/products/ps6441/index.html.

This section describes the following new and modified Cisco IOS commands, and specific commands that are used to configure the AIM IPS. This section contains the following topics:

- service-module ids-sensor, page 17-21
- service-module ids-bootmode, page 17-24

interface ids-sensor

To configure the IPS sensor interface and enter config-if mode, use the **interface ids-sensor** command in config mode. To specify how the router handles traffic inspection during a module failure, use the **service-module** command in config-if mode. The default is fail open.

interface ids-sensor *slot/port*

ip {**address** | **unnumbered**}

service-module {***fail-close*** | ***fail-open***}

Syntax Description

slot	Number of the router chassis slot for the AIM IPS.
/port	Port number of the AIM IPS. **Note** The slash mark is required between the *slot* argument and the *unit* argument.
ids-sensor	The IPS interface for the sensor.
ip address	Sets the IP address of an interface.
ip unnumbered	Enables IP address processing without an explicit IP address.
service-module fail-close	The AIM IPS drops all the traffic.
service-module fail-open	The AIM IPS passes all the traffic through, but does not perform traffic inspection (default).

Caution Although there are 57 subcommands associated with the **ip** command, the only two supported for the modules are **ip address** and ip **unnumbered**. Enabling any of the other subcommands can result in unpredictable behavior.

Command Defaults

Command Modes

Config

Config-if

Command History

Release	Modification
12.4(20)T	This command was introduced.

Usage Guidelines

The **interface ids-sensor** *slot/port* command lets you enter config-if mode and configure the IPS sensor slot and port. On the AIM IPS, the slot value is 0 and the port number value is specified by identifying the physical location where the module is installed on the router.

Examples

The following example demonstrates how to use the **interface ids-sensor** command to enter config-if mode on an AIM IPS in slot 0, port1:

```
router(config)# interface ids-sensor 0/1
router(config-if)#
```

The following example demonstrates how to use the **interface ids-sensor** command with the **ip unnumbered** subcommand to specify the router command and control interface:

```
router(config)# interface ids-sensor 0/1
router(config-if)# ip unnumbered router_command_and_control_interface
router(config-if)#
```

The following example demonstrates how to use the **service-module fail-open** command to configure the module to pass all traffic through the module when the hardware fails, but not to perform traffic inspection:

```
router(config)# interface ids-sensor 0/1
router(config-if)# service-module fail-open
router(config-if)#
```

Related Commands

Command	Description
interface *interface_name*	Lets you specify which interface should be monitored.

interface *interface_name*

To enter config-if mode, configure the interface for monitoring in promiscuous or inline mode, and apply a standard or extended ACL to inline monitoring, use the **interface** *interface_name* command in config mode.

interface *interface_name* **ids-service-module monitoring** {**promiscuous** | **inline**} **access-list** *number*

Syntax Description

interface_name	The name of the router interface to be monitored.
ids-service-module	Configures IPS on the interface.
monitoring	Specifies how the AIM IPS inspects traffic.
promiscuous	Specifies whether the AIM IPS inspects traffic in promiscuous mode.
inline	Specifies whether the AIM IPS inspects traffic in inline mode.
access-list	Specifies that you are applying a numbered or extended ACL to the inspected interface.
number	Number of the ACL.

Command Defaults

Command Modes

Config

Config-if

Command History

Release	Modification
12.4(20)T	This command was introduced.

Usage Guidelines

The **interface** *interface_name* command lets you enter config-if mode and configure the router to operate in inline or promiscuous mode for that interface.

Examples

The following example demonstrates how to use the **interface** command to enter config-if mode and configure monitoring for GigabitEthernet0/0 using ACL 101:

```
router(config)# interface GigabitEthernet0/0
router(config-if)# ids-service-module monitoring inline access-list 101
router(config-if)#
```

Related Commands

Command	Description
interface ids-sensor	Configures the IPS interface.

service-module ids-sensor

Caution When you reload the router, the AIM IPS also reloads. To ensure that there is no loss of data on the AIM IPS, make sure you shut down the module using the **shutdown** command before you use the **reload** command to reboot the router.

To prevent the Cisco IOS software from rebooting the AIM IPS when the heartbeat is lost, to reboot, reset, enable console access to, shut down, see the statistics, and monitor the status of a module, use the **service-module ids-sensor** command in privileged EXEC mode.

service-module ids-sensor *slot*/*port* {**heartbeat-reset** {**enable** | **disable**} **reload** | **reset** | **session** | **shutdown** | **status**}

Syntax Description

slot	Number of the router chassis slot for the AIM IPS.
/*port*	Port number of the AIM IPS. **Note** The slash mark is required between the *slot* argument and the *unit* argument.

heartbeat-reset	Enables or disables the heartbeat reset. The default is enabled. **Note** Disabling the heartbeat reset prevents the router from resetting the AIM IPS during system image installation if the process takes too long.
reload	Performs a graceful halt and reboot of the operating system on the AIM IPS.
reset	Resets the hardware on the AIM IPS. This command is usually used to recover from a shutdown.
session	Enables console access to the AIM IPS from the router.
shutdown	Shuts down the IPS application running on the AIM IPS.
statistics	Provides AIM IPS statistics.
status	Provides information about the status of the IPS software.

Defaults

Command Modes

Privileged EXEC

Command History

Release	Modification
12.4(15)XY	This command was introduced.
12.4(20)T	This command was introduced.

Usage Guidelines

When the AIM IPS is booted in failsafe mode or is undergoing an upgrade, you can use the **service-module ids heartbeat-reset** command to prevent a reboot during the process. If you leave the heartbeat reset enabled during an upgrade, you may lose the AIM IPS heartbeat.

When the AIM IPS heartbeat is lost, the router applies a fail-open or fail-close configuration option to the AIM IPS and stops sending traffic to the AIM IPS, and sets the AIM IPS to error state. The router performs a hardware reset on the AIM IPS and monitors the AIM IPS until the heartbeat is reestablished.

If a confirmation prompt is displayed, press **Enter** to confirm the action or **n** to cancel.

Examples

The following example demonstrates how to disable or enable the reset action when the heartbeat is lost on an AIM IPS in slot 0, port1:

```
router# service-module ids-sensor 0/1 heartbeat-reset {disable | enable}
```

The following example demonstrates how to enable the heartbeat on the AIM IPS:

```
router# service-module ids-sensor 0/1 heartbeat-reset enable
```

The following example demonstrates how todisplay the status of the heartbeat reset by using the **service-module ids** *slot/port* **status** command:

```
router# service-module ids-sensor 0/1 status
Service Module is Cisco IDS-Sensor 0/1
Service Module supports session via TTY line 194
```

```
Service Module heartbeat-reset is enabled
```

The following example demonstrates how to gracefully halt and reboot the operating system on the AIM IPS:

```
router# service-module ids-sensor 0/1 reload

Do you want to proceed with reload?[confirm]
```

The following example demonstrates how to reset the hardware on an AIM IPS. A warning is displayed.

```
router# service-module ids-sensor 0/1 reset

Use reset only to recover from shutdown or failed state
Warning: May lose data on the NVRAM, nonvolatile file system or unsaved configuration!

Do you want to reset?[confirm]
```

The following example demonstrates how to enable console access to the AIM IPS operating system:

```
router# service-module ids-sensor 0/1 session
```

The following example demonstrates how to shut down IPS applications running on the AIM IPS:

```
router# service-module ids-sensor 0/1 shutdown

Trying 10.10.10.1, 2129 ... Open
%SERVICEMODULE-5-SHUTDOWN2:Service module IDS-Sensor 0/1 shutdown complete
```

The following example demonstrates how to display IPS software statistics:

```
router# service-module ids-sensor 0/1 statistics
Module Reset Statistics:
  CLI reset count = 1
  CLI reload count = 0
  Registration request timeout reset count = 1
  Error recovery timeout reset count = 1
  Module registration count = 7

The last IOS initiated event was a cli reset at 20:18:36.038 UTC Tue Jan 16 2007
```

The following example demonstrates how to display the status of the IPS software on the AIM IPS:

```
router# service-module ids-sensor 0/1 status

Service Module is Cisco IDS-Sensor0/1
Service Module supports session via TTY line 33
Service Module is in Steady state
Getting status from the Service Module, please wait...
Service Module Version information received, Major ver = 1, Minor ver= 1

Cisco Systems Intrusion Prevention System Network Module
  Software version:  6.1(1)E1
  Model:             AIM-IPS
  Memory:            890996 KB
  Mgmt IP addr:      10.1.9.201
  Mgmt web ports:    443
  Mgmt TLS enabled:  true
```

Related Commands

Command	Description
ids-service-module monitoring	Enables IPS monitoring on a specified interface.

service-module ids-bootmode

To enter failsafe or normal boot mode for the AIM IPS, use the **service-module ids-sensor bootmode** command in privileged EXEC mode.

service-module ids-sensor *slot*/*port* **bootmode** {**failsafe** | **normal**}

Syntax Description

slot/	Number of the router chassis slot for the AIM IPS. The slash mark (/) is required between the *slot* argument and the *port* argument.
port	Port number of the AIM IPS.
failsafe	Enters failsafe boot mode on the AIM IPS .
normal	Enters normal boot mode on the AIM IPS.

Defaults

None

Command Modes

Privileged EXEC (#)

Command History

Release	Modification
12.4(15)XY	This command was introduced.
12.4(20)T	This command was integrated into Cisco IOS Release 12.4(20)T.

Usage Guidelines

If a confirmation prompt is displayed, press **Enter** to confirm the action, or press **n** to cancel.

Examples

The following example demonstrates how to enter failsafe boot mode on an AIM IPS in slot 0, port 1:

```
router# service-module ids-sensor 0/1 bootmode failsafe
```

The following example demonstrates how to enter normal boot mode on the AIM IPS:

```
router# service-module ids-sensor 0/1 bootmode normal
```

Related Commands

Command	Description
ids-service-module monitoring	Enables IDS monitoring on a specified interface.

CHAPTER 18

Configuring the AIP SSM

Note All IPS platforms allow ten concurrent log in sessions.

This chapter contains procedures that are specific to configuring the AIP SSM. It contains the following sections:

AIP SSM Configuration Sequence

Perform the following tasks to configure the AIP SSM:

1. Log in to the AIP SSM.
2. Initialize the AIP SSM.

   Run the **setup** command to initialize the AIP SSM.
3. Verify the AIP SSM initialization.
4. (Optional) If you have Cisco Adaptive Security Appliance Software 7.2.3 or later, configure multiple virtual sensors.
5. Configure adaptive security appliance to send IPS traffic to the AIP SSM.
6. Perform other initial tasks, such as adding users, trusted hosts, and so forth.
7. Configure intrusion prevention.
8. Perform miscellaneous tasks to keep your AIP SSM running smoothly.

9. Upgrade the IPS software with new signature updates and service packs.
10. Reimage the AIP SSM when needed.

For More Information

- For the procedure for logging in to the AIP SSM, see Chapter 2, "Logging In to the Sensor."
- For the procedure for running the **setup** command, see Advanced Setup for the AIP SSM, page 3-15.
- For the procedure for verifying the AIP SSM initialization, see Verifying AIP SSM Initialization, page 18-2.
- For the procedure for creating virtual sensors, see Creating Virtual Sensors, page 18-3.
- For the procedure configuring ASA to send traffic to the AIP SSM, see Sending Traffic to the AIP SSM, page 18-9.
- For the procedures for setting up the sensor, see Chapter 4, "Setting Up the Sensor."
- For the procedures for configuring intrusion prevention, see Chapter 7, "Configuring Event Action Rules," Chapter 8, "Defining Signatures," Chapter 9, "Configuring Anomaly Detection,"and Chapter 13, "Configuring Attack Response Controller for Blocking and Rate Limiting,"
- For the procedures for keeping your AIP SSM running smoothly, see Chapter 16, "Administrative Tasks for the Sensor."
- For more information on how to obtain Cisco IPS software, see Chapter 21, "Obtaining Software."
- For the procedure for reimaging the AIP SSM, see Installing the AIP SSM System Image, page 22-24.

Verifying AIP SSM Initialization

You can use the **show module** *slot* **details** command to verify that you have initialized the AIP SSM 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 AIP SSM.

```
asa# show module 1 details
ASA 5500 Series Security Services Module-10
Model:              ASA-SSM-10
Hardware version:   1.0
Serial Number:      JAB09370212
Firmware version:   1.0(10)0
Software version:   6.0(4)E1
MAC Address Range:  0012.d948.fe73 to 0012.d948.fe73
App. name:          IPS
App. Status:        Up
App. Status Desc:
App. version:       6.0(4)E1
```

```
Data plane Status:  Up
Status:             Up
Mgmt IP addr: 171.69.36.171
Mgmt web ports:     443
Mgmt TLS enabled:   true
asa#
```

Step 3 Confirm the information.

Creating Virtual Sensors

Caution Cisco Adaptive Security Appliance Software 7.2.3 or later supports virtualization.

This section describes how to create virtual sensors on the AIP SSM, and contains the following topics:

- Virtual Sensor Configuration Sequence, page 18-3
- Creating Virtual Sensors on the AIP SSM, page 18-4
- Assigning Virtual Sensors to Adaptive Security Appliance Contexts, page 18-6

The AIP SSM and Virtualization

The AIP SSM has one interface, GigabitEthernet0/1. 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 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, and the **fail-open** policy is enforced.

Virtual Sensor Configuration Sequence

Follow this sequence to create virtual sensors on the AIP SSM and to assign them to adaptive security device contexts:

1. Configure up to four virtual sensors on the AIP SSM.
2. Assign the AIP SSM interface, GigabitEthernet0/1, to one of the virtual sensors.
3. Assign virtual sensors to different contexts on the adaptive security device.
4. Use MPF to direct traffic to the targeted virtual sensor.

Creating Virtual Sensors on the AIP SSM

Use the **virtual-sensor** *name* command in service analysis engine submode to create virtual sensors on the AIP SSM.

Note You can create four virtual sensors.

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 interface GigabitEthernet0/1 to one virtual sensor.

The following options apply:

- **anomaly-detection**—Anomaly detection parameters
  - **anomaly-detection-name** *name*—Name of the anomaly detection policy
  - **operational-mode**—Anomaly detection mode (**inactive**, **learn**, **detect**)
- **description**—Description of the virtual sensor
- **event-action-rules**—Name of the event action rules policy
- **signature-definition**—Name of the signature definition policy
- **physical-interfaces**—Name of the physical interface
- **no**—Removes an entry or selection

To create a virtual sensor on the AIP SSM, 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 Assign the interface to one virtual sensor.

```
sensor(config-ana-vir)# physical-interface GigabitEthernet0/1
```

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: GigabitEthernet0/1
         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 7-11.
- For the procedure for creating and configuring signature definitions, Working With Signature Definition Policies, page 8-1.

Assigning Virtual Sensors to Adaptive Security Appliance Contexts

After you create virtual sensors on the AIP SSM, you must assign them to a security context on the adaptive security appliance.

The following options apply:

- **[no] allocate-ips** *sensor_name* [*mapped_name*] [**default**]—Allocates a virtual sensor to a security context. Supported mode are multiple mode, system context, and context submode.

  Note You cannot allocate the same AIP SSM twice in a context.

  - *sensor_name*—Name of the AIP SSM. You receive a warning message if the name is not valid.
  - *mapped_name*—Name by which the security context knows the AIP SSM.

    Note The mapped name is used the hide the real name of the AIP SSM 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 AIP SSM name is used. You cannot reuse a mapped name for two different AIP SSMs 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 AIP SSM as the default. All legacy IPS configurations that do not specify a virtual sensor are mapped to this AIP SSM.

Caution You can only configure one default AIP SSM per context. You must turn off the default flag of an existing default AIP SSM before you can designate another AIP SSM as the default.

  - **clear configure allocate-ips**—Removes the configuration.
  - **allocate-ips?**—Displays the list of configured AIP SSMs.

- **show ips [detail]**—Displays all available virtual sensors. Supported modes are EXEC mode, single or multiple, system or user modes.
  - **detail**—Adds the virtual sensor ID number.

    Note In single mode, the command shows the names of all available virtual sensors. In multiple mode user context, the command shows the mapped names of all virtual sensors that have been allocated to this context. In multiple mode system context, the command shows the names of all virtual sensors and with the detail keyword, the sensor ID number, allocated context, and mapped name are displayed.

- **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, two new lines are added to show the real and mapped names of virtual sensors allocated to this context.

After you create virtual sensors on the AIP SSM, you must assign them to a security context on the adaptive security appliance.

The following options apply:

- **[no] allocate-ips** *sensor_name* [*mapped_name*] [**default**]—Allocates a virtual sensor to a security context. Supported mode are multiple mode, system context, and context submode.

  Note You cannot allocate the same AIP SSM twice in a context.

  - *sensor_name*—Name of AIP SSM configured on the AIP SSM. You receive a warning message if the name is not valid.
  - *mapped_name*—Name by which the security context knows the AIP SSM.

    Note The mapped name is used the hide the real name of the AIP SSM 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 AIP SSM name is used. You cannot reuse a mapped name for two different AIP SSMs 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 AIP SSM as the default. All legacy IPS configurations that do not specify a virtual sensor are mapped to this AIP SSM.

Caution You can only configure one default AIP SSM per context. You must turn off the default flag of an existing default AIP SSM before you can designate another AIP SSM as the default.

  - **clear configure allocate-ips**—Removes the configuration.
  - **allocate-ips?**—Displays the list of configured AIP SSMs.

- **show ips [detail]**—Displays all available virtual sensors. Supported modes are EXEC mode, single or multiple, system or user modes.
  - **detail**—Adds the virtual sensor ID number.

    Note In single mode, the command shows the names of all available virtual sensors. In multiple mode user context, the command shows the mapped names of all virtual sensors that have been allocated to this context. In multiple mode system context, the command shows the names of all virtual sensors and with the detail keyword, the sensor ID number, allocated context, and mapped name are displayed.

- **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, two new lines are added to show the real and mapped names of virtual sensors allocated to this context.

The following procedure demonstrates how to add three security contexts in multiple mode and how to assign virtual sensors to these security contexts.

Note 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.

To assign the AIP SSM virtual sensors to adaptive security appliance contexts in multiple mode, follow these steps:

Step 1 Log in to the adaptive security appliance.

Step 2 Display the list of available virtual sensors.

```
asa# show ips detail
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)#
```

Sending Traffic to the AIP SSM

This section describes how to configure the AIP SSM to receive IPS traffic from the adaptive security appliance (inline or promiscuous mode) and contains the following sections:

- ASA and the AIP SSM, page 18-9
- Configuring the Adaptive Security Appliance to Send IPS Traffic to the AIP SSM, page 18-10

ASA and the AIP SSM

The adaptive security appliance diverts packets to AIP SSM just before the packet exits the egress interface (or before VPN encryption occurs, if configured) and after other firewall policies are applied. For example, packets that are blocked by an access list are not forwarded to AIP SSM.

You can configure the AIP SSM to inspect traffic in inline or promiscuous mode and in fail-open or fail-over mode.

On the adaptive security appliance, to identify traffic to be diverted to and inspected by AIP SSM:

1. Create or use an existing ACL.
2. Use the **class-map** command to define the IPS traffic class.
3. Use the **policy-map** command to create an IPS policy map by associating the traffic class with one or more actions.
4. Use the **service-policy** command to create an IPS security policy by associating the policy map with one or more interfaces.

   You can use the adaptive security appliance CLI or ASDM to configure IPS traffic inspection.

Configuring the Adaptive Security Appliance to Send IPS Traffic to the AIP SSM

Note For more information on these commands, refer to "Using Modular Policy Framework," in *Cisco Security Appliance Command Line Configuration Guide*.

The following options apply:

- **access-list** *word*—Configures an access control element; *word* is the access list identifier (up to 241 characters).
- **class-map** *class_map_name*—Defines the IPS traffic class.
- **match**—Identifies the traffic included in the traffic class.

  A traffic class map contains a **match** command. When a packet is matched against a class map, the match result is either a match or a no match.

  - **access-list**—Matches an access list.
  - **any**—Matches any packet.
- **policy-map** *policy_map_name*—Creates an IPS policy map by associating the traffic class with one or more actions.
- **ips {inline | promiscuous} {fail-open | fail-close} {sensor** *sensor_name*}—Assigns traffic from the security appliance to a specified virtual sensor on the AIP SSM. If no virtual sensor is specified, traffic is assigned to the default virtual sensor. Supported modes are single or multi mode, user context, config mode, and policy map class submode.

  - **inline**—Places the AIP SSM directly in the traffic flow.

    No traffic can continue through the adaptive security appliance without first passing through and being inspected by the AIP SSM. This mode is the most secure because every packet is analyzed before being permitted through. Also, the AIP SSM can implement a blocking policy on a packet-by-packet basis. This mode, however, can affect throughput.

- **promiscuous**—Sends a duplicate stream of traffic to the AIP SSM.

  This mode is less secure, but has little impact on traffic throughput. Unlike when in inline mode, the AIP SSM cannot block traffic by instructing the adaptive security appliance to block the traffic or by resetting a connection on the adaptive security appliance.
- **fail-close**—Sets the adaptive security appliance to block all traffic if the AIP SSM is unavailable.
- **fail-open**—Sets the adaptive security appliance to permit all traffic through, uninspected, if the AIP SSM is unavailable.

Note The adaptive security appliance fail-open/fail-close behavior depends on low-level heartbeats, which are turned off when the AIP SSM is shut down or reset. If the AIP SSM fails, the adaptive security appliance cannot detect this failure because the heartbeats are still received. For inline inspection of traffic, use IPS bypass mode to drop or permit traffic through.

- **sensor** *sensor_name*—Name of the allocated virtual sensor. If the sensor name was mapped, the mapped name is used. Otherwise, the real sensor name is used.

- **service-policy** *service_policy_name* {**global** | **interface** *interface_name*}—Creates an IPS security policy by associating the policy map with one or more interfaces.
  - **global**—Applies the policy map to all interfaces.

    Only one global policy is allowed. You can override the global policy on an interface by applying a service policy to that interface. You can only apply one policy map to each interface.
  - **interface**—Applies the policy to one interface.

    You can assign a different policy for each interface.

To allocate virtual sensors and send traffic from the adaptive security appliance to the AIP SSM for the IPS to inspect, follow these steps:

Step 1 Log in to the adaptive security appliance.

Step 2 Enter configuration mode.

```
asa# configure terminal
```

Step 3 Create an IPS access list.

```
asa(config)# access-list IPS permit ip any any
```

Step 4 Define the IPS traffic class.

```
asa(config)# class-map class_map_name
asa(config-cmap)# match [access-list | any]
```

Step 5 Define the IPS policy map.

```
asa(config-cmap)# policy-map policy_map_name
```

Step 6 Identify the class map from Step 5 to which you want to assign an action.

```
asa(config-pmap)# class class_map_name
```

Step 7 Assign traffic to the AIP SSM.

```
asa(config-pmap-c)# ips {inline | promiscuous} {fail-close | fail-open}
```

Step 8 Define the IPS service policy.

```
asa(config-pmap-c)# service-policy policymap_name {global | interface interface_name}
```

Step 9 Verify the settings.

```
asa(config-pmap-c)# show running-config
!
class-map my_ips_class
match access-list IPS
class-map all_traffic
 match access-list all_traffic
class-map inspection_default
 match default-inspection-traffic
!
!
policy-map my-ids-policy
 class my-ips-class
  ips promiscuous fail-close
!
service-policy my-ids-policy global
```

Step 10 Exit and save the configuration.

```
asa(config-pmap-c)# exit
asa(config-pmap)# exit
asa(config)# exit
asa#
```

The following example diverts all IP traffic to AIP SSM in inline mode, and blocks all IP traffic should AIP SSM fail for any reason:

```
hostname(config)# access-list IPS permit ip any any
hostname(config)# class-map my-ips-class
hostname(config-cmap)# match access-list IPS
hostname(config-cmap)# policy-map my-ids-policy
hostname(config-pmap)# class my-ips-class
hostname(config-pmap-c)# ips inline fail-close
hostname(config-pmap-c)# service-policy my-ids-policy global
```

For More Information

For more information on bypass mode, see Adaptive Security Appliance, the AIP SSM, and Bypass Mode, page 18-12.

Adaptive Security Appliance, the AIP SSM, and Bypass Mode

The following conditions apply to bypass mode configuration, the adaptive security appliance, and the AIP SSM.

The SensorApp Fails OR a Configuration Update is Taking Place

The following occurs when bypass is set to Auto or Off on the AIP SSM:

- Bypass Auto—Traffic passes without inspection.
- Bypass Off—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 AIP SSM.
- If set to fail-close, the adaptive security appliance stops passing traffic until the AIP SSM is restarted or completes reconfiguration.

Note When bypass is set to On, traffic passes without inspection regardless of the state of the SensorApp.

The AIP SSM Is Rebooted or Not Responding

The following occurs according to how the adaptive security appliance is configured for failover:

- 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 AIP SSM.
  - If set to fail-close, the adaptive security appliance stops passing traffic until the AIP SSM is restarted.

For More Information

For more information on bypass mode, see Configuring Inline Bypass Mode, page 5-33.

AIP SSM and the Normalizer Engine

The majority of the features in the Normalizer engine are not used on the AIP SSM, 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 more information on the Normalizer engine, see Normalizer Engine, page B-22.

Reloading, Shutting Down, Resetting, and Recovering the AIP SSM

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 devices 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 AIP SSM directly from the adaptive security appliance:

- **hw-module module** *slot_number* **reload**

  This command reloads the software on the AIP SSM without doing a hardware reset. It is effective only when the AIP SSM is in the Up state.

- **hw-module module** *slot_number* **shutdown**

  This command shuts down the software on the AIP SSM. It is effective only when the AIP SSM is in Up state.

- **hw-module module** *slot_number* **reset**

  This command performs a hardware reset of the AIP SSM. It is applicable when the card is in the Up/Down/Unresponsive/Recover states.

- **hw-module module** *slot_number* **password-reset**

  This command restores the cisco CLI account password 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. You can change the parameter or keep the existing setting by pressing Enter.

  – **hw-module module** *slot_number* **recover boot**

    This command initiates recovery of the AIP SSM. It is applicable only when the AIP SSM is in the Up state.

– **hw-module module** *slot_number* **recover stop**

This command stops recovery of the AIP SSM. It is applicable only when the AIP SSM is in the Recover state.

Caution If the AIP SSM recovery needs to be stopped, you must issue the **hw-module module 1 recover stop** command within 30 to 45 seconds after starting the AIP SSM recovery. Waiting any longer can lead to unexpected consequences. For example, the AIP SSM may come up in the Unresponsive state.

– **hw-module module 1 recover configure**

Use this command to configure parameters for module recovery. The essential parameters are the IP address and recovery image TFTP URL location.

Example:

```
aip-ssm# hardware-module module 1 recover configure
Image URL [tftp://10.89.146.1/IPS-SSM-K9-sys-1.1-a-5.1-1.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 AIP SSM, see Installing the AIP SSM System Image, page 22-24.

CHAPTER 19

Configuring the IDSM2

Note All IPS platforms allow ten concurrent log in sessions.

Note Catalyst 6500 Series Switch is used generically to refer to both the 6500 series switches and the 7600 series routers.

This chapter contains procedures that are specific to configuring the IDSM2. Once you set up the IDSM2 to receive traffic from the network, you can configure it for intrusion prevention. It contains the following sections:

IDSM2 Configuration Sequence

Perform the following tasks to configure the IDSM2:

1. Configure the Catalyst 6500 series switch for command and control access to the IDSM2.
2. Log in to the IDSM2.
3. Configure the switch to send traffic to be monitored to the IDSM2.

4. Initialize the IDSM2.

   Run the **setup** command to initialize the IDSM2. During setup, you can configure the interfaces of the IDSM2.
5. Create the service account.
6. Perform the other initial tasks, such as adding users, trusted hosts, and so forth.
7. Configure intrusion prevention.
8. Perform miscellaneous tasks to keep the IDSM2 running smoothly.
9. Upgrade the IPS software with new signature updates and service packs.
10. Reimage the application partition and the maintenance partition when needed.

For More Information

- For the procedure to session to the IDSM2, see Logging In to the IDSM2, page 2-6.
- For the procedure, see Advanced Setup for the IDSM2, page 3-20.
- For the procedure for configuring command and control access to the IDSM2, see Configuring the Catalyst 6500 Series Switch for Command and Control Access to the IDSM2, page 19-5.
- To change the interface configuration, see Chapter 5, "Configuring Interfaces."
- For the procedure, see Creating the Service Account, page 4-14.
- For the procedures, see Chapter 4, "Setting Up the Sensor."
- For the procedures for configuring intrusion prevention, see Chapter 7, "Configuring Event Action Rules," Chapter 8, "Defining Signatures," Chapter 9, "Configuring Anomaly Detection," and Chapter 13, "Configuring Attack Response Controller for Blocking and Rate Limiting."
- For the procedures for administrative tasks for the IDSM2, see Chapter 16, "Administrative Tasks for the Sensor" and Administrative Tasks for the IDSM2, page 19-39.
- For more information on how to obtain the most recent IPS software, see Chapter 21, "Obtaining Software."
- For the procedures for reimaging the application and maintenance partitions, see Installing the IDSM2 System Image, page 22-26.

Verifying the IDSM2 Installation

Use the **show module** command to verify that the switch acknowledges the IDSM2 and has brought it online.

To verify the installation, follow these steps:

Step 1 Log in to the console.

Step 2 For Catalyst software:

```
console> (enable) show module
Mod Slot Ports Module-Type                Model               Sub Status
--- ---- ----- ------------------------- ------------------- --- --------
1   1    2     1000BaseX Supervisor      WS-X6K-SUP1A-2GE    yes ok
15  1    1     Multilayer Switch Feature WS-F6K-MSFC         no  ok
2   2    48    10/100BaseTX Ethernet     WS-X6248-RJ-45      no  ok
3   3    48    10/100/1000BaseT Ethernet WS-X6548-GE-TX      no  ok
4   4    16    1000BaseX Ethernet        WS-X6516A-GBIC      no  ok
```

```
6   6    8     Intrusion Detection Mod   WS-SVC-IDSM2        yes ok

Mod Module-Name          Serial-Num
--- -------------------- -----------
1                        SAD041308AN
15                       SAD04120BRB
2                        SAD03475400
3                        SAD073906RC
4                        SAL0751QYN0
6                        SAD062004LV

Mod MAC-Address(es)                         Hw     Fw          Sw
--- ------------------------------------ ------ ---------- -----------------
1   00-d0-c0-cc-0e-d2 to 00-d0-c0-cc-0e-d3 3.1    5.3.1      8.4(1)
    00-d0-c0-cc-0e-d0 to 00-d0-c0-cc-0e-d1
    00-30-71-34-10-00 to 00-30-71-34-13-ff
15  00-30-7b-91-77-b0 to 00-30-7b-91-77-ef 1.4    12.1(23)E2 12.1(23)E2
2   00-30-96-2b-c7-2c to 00-30-96-2b-c7-5b 1.1    4.2(0.24)V 8.4(1)
3   00-0d-29-f6-01-98 to 00-0d-29-f6-01-c7 5.0    7.2(1)     8.4(1)
4   00-0e-83-af-15-48 to 00-0e-83-af-15-57 1.0    7.2(1)     8.4(1)
6   00-e0-b0-ff-3b-80 to 00-e0-b0-ff-3b-87 0.102  7.2(0.67)  5.0(0.30)

Mod Sub-Type                Sub-Model           Sub-Serial  Sub-Hw Sub-Sw
--- ----------------------- ------------------- ----------- ------ ------
1   L3 Switching Engine     WS-F6K-PFC          SAD041303G6 1.1
6   IDS 2 accelerator board WS-SVC-IDSUPG       .           2.0
console> (enable)
```

Step 3 For Cisco IOS software:

```
router# show module
Mod Ports Card Type                                  Model              Serial No.
--- ----- ------------------------------------- ------------------ -----------
  1   48  48 port 10/100 mb RJ-45 ethernet       WS-X6248-RJ-45     SAD0401012S
  2   48  48 port 10/100 mb RJ45                 WS-X6348-RJ-45     SAL04483QBL
  3   48  SFM-capable 48 port 10/100/1000mb RJ45 WS-X6548-GE-TX     SAD073906GH
  6   16  SFM-capable 16 port 1000mb GBIC        WS-X6516A-GBIC     SAL0740MMYJ
  7    2  Supervisor Engine 720 (Active)         WS-SUP720-3BXL     SAD08320L2T
  9    1  1 port 10-Gigabit Ethernet Module      WS-X6502-10GE      SAD071903BT
 10    3  Anomaly Detector Module                WS-SVC-ADM-1-K9    SAD084104JR
 11    8  Intrusion Detection System             WS-SVC-IDSM2       SAD05380608
 13    8  Intrusion Detection System             WS-SVC-IDSM-2      SAD072405D8

Mod MAC addresses                        Hw    Fw           Sw           Status
--- ---------------------------------- ------ ------------ ------------ -------
  1  00d0.d328.e2ac to 00d0.d328.e2db   1.1   4.2(0.24)VAI 8.5(0.46)ROC Ok
  2  0003.6c14.e1d0 to 0003.6c14.e1ff   1.4   5.4(2)       8.5(0.46)ROC Ok
  3  000d.29f6.7a80 to 000d.29f6.7aaf   5.0   7.2(1)       8.5(0.46)ROC Ok
  6  000d.ed23.1658 to 000d.ed23.1667   1.0   7.2(1)       8.5(0.46)ROC Ok
  7  0011.21a1.1398 to 0011.21a1.139b   4.0   8.1(3)       12.2(PIKESPE Ok
  9  000d.29c1.41bc to 000d.29c1.41bc   1.3   Unknown      Unknown      PwrDown
 10  000b.fcf8.2ca8 to 000b.fcf8.2caf   0.101 7.2(1)       4.0(0.25)    Ok
 11  00e0.b0ff.3340 to 00e0.b0ff.3347   0.102 7.2(0.67)    5.0(1)       Ok
 13  0003.feab.c850 to 0003.feab.c857   4.0   7.2(1)       5.0(1)       Ok

Mod Sub-Module                  Model               Serial        Hw     Status
--- --------------------------- ------------------ ------------ ------- -------
  7 Policy Feature Card 3       WS-F6K-PFC3BXL     SAD083305A1   1.3    Ok
  7 MSFC3 Daughterboard         WS-SUP720          SAD083206JX   2.1    Ok
 11 IDS 2 accelerator board     WS-SVC-IDSUPG      .             2.0    Ok
 13 IDS 2 accelerator board     WS-SVC-IDSUPG      0347331976    2.0    Ok

Mod Online Diag Status
--- -------------------
```

```
  1 Pass
  2 Pass
  3 Pass
  6 Pass
  7 Pass
  9 Unknown
 10 Not Applicable
 11 Pass
 13 Pass
router#
```

Note It is normal for the status to read `other` when the IDSM2 is first installed. After the IDSM2 completes the diagnostics routines and comes online, the status reads `ok`. Allow up to 5 minutes for the IDSM2 to come online.

For More Information

For information on enabling a full memory test after verifying the IDSM2 installation, see Enabling Full Memory Tests, page 19-40.

Minimum Supported IDSM2 Configurations

Note The following matrix is not intended to recommend any particular version, but rather lists the earliest supported versions.

Table 19-1 lists the minimum supported configurations for the IDSM2.

Table 19-1 Minimum Catalyst 6500 Software Version for IDSM2 Feature Support

Catalyst/IDSM2 Feature	Catalyst Software				Cisco IOS Software			
	Sup1	Sup2	Sup32	Sup720	Sup1	Sup2	Sup32	Sup720
SPAN	7.5(1)	7.5(1)	8.4(1)	8.1(1)	12.1(19)E1	12.1(19)E1 12.2(18)SXF1	12.2(18)SXF1	12.2(14)SX1
VACL capture[1]	7.5(1)	7.5(1)	8.4(1)	8.1(1)	12.1(19)E1	12.1(19)E1 12.2(18)SXF1	12.2(18)SXF1	12.2(14)SX1
ECLB with VACL capture[2]	8.5(1)	8.5(1)	8.5(1)	8.5(1)	N/A	12.2(18)SXF4	12.2(18)SXF1	12.2(18)SXE1
Inline interface pairs	8.4(1)	8.4(1)	8.4(1)	8.4(1)	N/A	12.2(18)SXF4	12.2(18)SXF4	12.2(18)SXE1
ECLB with inline interface pairs	8.5(1)	8.5(1)	8.5(1)	8.5(1)	N/A	12.2(18)SXF4	12.2(18)SXF4	12.2(18)SXF4
Inline VLAN pairs	8.4(1)	8.4(1)	8.4(1)	8.4(1)	N/A	12.2(18)SXF4	12.2(18)SXF4	12.2(18)SXF4
ECLB with inline VLAN pairs	8.5(1)	8.5(1)	8.5(1)	8.5(1)	N/A	12.2(18)SXF4	12.2(18)SXF4	12.2(18)SXF4

1. Requires PFC2/3 or MSFC2/3.
2. Requires PFC2/3 or MSFC2/3.

Configuring the Catalyst 6500 Series Switch for Command and Control Access to the IDSM2

You must configure the Catalyst 6500 series switch to have command and control access to the IDSM2. This section describes how to configure the switch to have command and control access, and contains the following topics:

Catalyst Software

To configure the Catalyst 6500 series switch to have command and control access to the IDSM2, follow these steps:

Step 1 Log in to the console.

Step 2 Enter privileged mode.

```
console> enable
```

Step 3 Put the command and control port into the correct VLAN.

```
console> (enable) set vlan command_and_control_vlan_number
idsm2_slot_number/command_and_control_port_number
```

Example

```
console> (enable) set vlan 147 6/2
VLAN 147 modified.
VLAN 146 modified.
VLAN  Mod/Ports
---- -----------------------
147   2/5,2/16-18
      6/2
```

The command and control port number is always 2.

Step 4 Session to the IDSM2 and ping a network IP address.

```
console> session slot_number
idsm-2# ping network_ip_address
```

Example

```
console> (enable) session 6
Trying IDS-6...
Connected to IDS-6.
Escape character is '^]'.

login: cisco
Password:
Last login: Thu Mar 3 09:40:53 from 127.0.0.11
***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.
idsm-2# ping 10.89.149.126
PING 10.89.149.126 (10.89.149.126): 56 data bytes
64 bytes from 10.89.149.126: icmp_seq=0 ttl=255 time=0.3 ms
64 bytes from 10.89.149.126: icmp_seq=1 ttl=255 time=0.3 ms
64 bytes from 10.89.149.126: icmp_seq=2 ttl=255 time=0.3 ms
64 bytes from 10.89.149.126: icmp_seq=3 ttl=255 time=0.3 ms
--- 10.89.149.126 ping statistics ---
4 packets transmitted, 4 packets received, 0% packet loss
round-trip min/avg/max = 0.3/0.3/0.3 ms
idsm-2# exit
console> (enable)
```

Step 5 Initialize the IDSM2.

Step 6 Ping the default router of the IDSM2.

Step 7 Verify the management station can ping, SSH or Telnet, and web browse to the IDSM2.

For More Information

For the procedure for using the **setup** command to initialize the IDSM2, see Advanced Setup for the IDSM2, page 3-20.

Cisco IOS Software

To configure the Catalyst 6500 series switch to have command and control access to the IDSM2, follow these steps:

Step 1 Log in to the console.

Step 2 Enter global configuration mode.

```
router# configure terminal
```

Step 3 Put the command and control port into the correct VLAN.

```
router (config)# intrusion-detection module module_number management-port access-vlan
vlan_number
```

Example

```
router (config)# intrusion-detection module 11 management-port access-vlan 146
```

Step 4 Verify that you have connectivity by sessioning in to the IDSM2 and pinging a network IP address.

```
router# session slot module_number processor 1
idsm-2# ping network_ip_address
```

Example

```
router# session slot 11 processor 1
The default escape character is Ctrl-^, then x.
You can also type 'exit' at the remote prompt to end the session
Trying 127.0.0.91 ... Open

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.
idsm-2# ping 10.89.149.254
PING 10.89.149.254 (10.89.149.254): 56 data bytes
64 bytes from 10.89.149.254: icmp_seq=0 ttl=255 time=0.2 ms
64 bytes from 10.89.149.254: icmp_seq=1 ttl=255 time=0.2 ms
64 bytes from 10.89.149.254: icmp_seq=2 ttl=255 time=0.2 ms
64 bytes from 10.89.149.254: icmp_seq=3 ttl=255 time=0.2 ms
--- 10.89.149.254 ping statistics ---
4 packets transmitted, 4 packets received, 0% packet loss
round-trip min/avg/max = 0.2/0.2/0.2 ms
idsm-2# exit
[Connection to 127.0.0.91 closed by foreign host]
router#
```

Step 5 Initialize the IDSM2.

For More Information

For the procedure for using the **setup** command to initialize the IDSM2, see Advanced Setup for the IDSM2, page 3-20.

The IDSM2 Sensing Modes

The IDSM2 supports three sensing modes:

- Promiscuous mode—When the IDSM2 was introduced, promiscuous mode was the only sensing mode supported on the IDSM2 and it is the default sensing mode for both data ports.

  In promiscuous mode, the IDSM2 passively monitors network traffic copied to its data ports by the Catalyst switch. The data ports operate as 802.1q trunks and you can configure the two data ports to trunk the same or different VLANs. The Catalyst switch uses either SPAN or VACL capture to copy specific traffic to the data ports. You can send the same or different traffic to the two data ports.

Because the IDSM2 is passive in this mode, it cannot drop packets to block a network intrusion attempt, but you can configure it to send TCP resets to both sides of the network connection to try to break the connection.

Note Because the Catalyst switch does not forward traffic received from a capture destination port, the IDSM2 cannot send TCP resets over the data ports to try to block an intrusion. Therefore, a separate reset port available only in promiscuous mode is reserved for this purpose.

- Inline mode—Beginning with IPS 5.0(1), you can configure the IDSM2 to be an active network device in inline interface pair mode. The two data ports operate together to bridge two VLANs through the IDSM2. You configure each data port as an access port and assign a different VLAN to each data port. The IDSM2 bridges the two VLANs by forwarding traffic between the two data ports. It inspects the traffic it receives on each data port and can either forward the packets to the other data port or drop the packet if it detects intrusion. You must configure the switch for inline mode, and then create the inline interface pairs on the IDSM2.
- Inline VLAN pair mode—Beginning with IPS 5.1(1), you can configure the IDSM2 in inline VLAN pair mode. The IDSM2 performs VLAN bridging between pairs of VLANs within the same data port operating as an 802.1q trunk. The IDSM2 inspects the traffic it receives on each VLAN in a VLAN pair and can either forward the packets on the other VLAN in the pair (on the same data port on which the packet was received) or drop the packet if an intrusion is detected. You can configure the IDSM2 to simultaneously bridge up to 255 VLAN pairs on each data port. The IDSM2 replaces the VLAN ID field in the 802.1q header of each packet with the ID of the VLAN on which the packet is forwarded. It drops any packets received on VLANs that are not assigned to an inline VLAN pair.

  Note You are responsible for coordinating the IPS and switch configuration to make sure each of the VLANS associated with an inline VLAN pair is also an allowed VLAN for the data port trunk.

You can mix sensing modes on the IDSM2. For example, you can configure one data port for promiscuous mode and the other data port for inline VLAN pair mode. But because the IDSM2 only has two data ports and inline mode requires the use of both data ports as a pair, you cannot mix inline mode with either of the other two modes.

For More Information

- For more information on promiscuous mode, see Configuring Promiscuous Mode, page 5-15.
- For more information on TCP reset, see TCP Reset Interfaces, page 5-4.
- For the procedure for configuring SPAN, see Configuring SPAN, page 19-10.
- For the procedure for configuring VACL capture, see Configuring VACL Capture, page 19-14.
- For more information on inline mode, see Configuring Inline Interface Mode, page 5-16.
- For more information con configuring the switch for inline mode, see Configuring the Catalyst Series 6500 Switch for the IDSM2 in Inline Mode, page 19-19.
- For more information on configuring the IDSM2 in line mode, see Configuring Inline Interface Pairs, page 5-16.
- For more information on inline VLAN pair mode, see Configuring Inline VLAN Pair Mode, page 5-20.

- For the procedure for configuring the switch for inline VLAN pair mode, see Configuring the Catalyst Series 6500 Switch for the IDSM2 for Inline VLAN Pair Mode, page 19-22.
- For the procedure for configuring the IDSM2 inline VLAN pair mode, see Configuring Inline VLAN Pairs, page 5-21.

Configuring the Catalyst Series 6500 Switch for the IDSM2 in Promiscuous Mode

This section describes how to configure the switch so that the IDSM2 operates in promiscuous mode, and contains the following topics:

Understanding the Switch, the IDSM2, and Promiscuous Mode

Traffic is captured for promiscuous analysis on the IDSM2 through SPAN or VACL capture (if you are running the Cisco IOS Firewall on the MSFC, you cannot use VACLs, but you can use the **mls ip ids** command). Port 1 (GigabitEthernet0/1) is used as the TCP reset port, port 2 (GigabitEthernet0/2) is the command and control port, and ports 7 and 8 (GigabitEthernet0/7 and GigabitEthernet0/8) are the monitoring ports. You can configure both monitoring ports to be either SPAN destination ports or VACL capture ports.

Caution If you configure both ports as monitoring ports, make sure that they are configured to monitor different traffic.

Caution You should not configure an IDSM2 data port as both a SPAN destination port and a VACL capture port, because the IDSM2 will not receive traffic. This dual configuration (SPAN and VACL) causes problems on the switch and traffic is not sent properly.

Note Before Catalyst Software 8.4(3), the IDSM2 data ports defaulted to trunking all VLANs. In Catalyst Software 8.4(3) and later, the IDSM2 data ports default to trunking no VLANs. Make sure that the IDSM2 ports are trunking the proper VLANs, especially if you are upgrading from pre-8.4(3) to 8.4(3) or later.

Using the TCP Reset Interface

The IDSM2 has a TCP reset interface—port 1. The IDSM2 has a specific TCP reset interface because it cannot send TCP resets on its sensing ports.

If you have reset problems with the IDSM2, and the switch is running Catalyst software, try the following:

- If the sensing ports are access ports (a single VLAN), you need to configure the reset port to be in the same VLAN.
- If the sensing ports are dot1q trunk ports (multi-VLAN), the sensing ports and reset port all must have the same native VLAN, and the reset port must trunk all the VLANs being trunked by both the sensing ports.

Note In Cisco IOS when the IDSM2 is in promiscuous mode, the IDSM2 ports are always dot1q trunk ports (even when monitoring only 1 VLAN), and the TCP reset port is automatically set to a trunk port and is not configurable.

Configuring SPAN

The IDSM2 can analyze Ethernet VLAN traffic from Ethernet or Fast Ethernet SPAN source ports, or you can specify an Ethernet VLAN as the SPAN source. This section describes how to configure SPAN, and contains the following topics:

- Catalyst Software, page 19-10
- Cisco IOS Software, page 19-12

Catalyst Software

Note The IDSM2 port numbers are 7 or 8 only.

Use the **set span** command in privileged mode to enable SPAN to the IDSM2.

The following options apply:

- **disable**—Disables port monitoring.
- *module/port*—Source module and port numbers.
- *vlan*—Source VLAN numbers.
- *module/port*—Destination module and port numbers.
- **both**—Both receiving and transmitting traffic.
- **filter**—Applies filter to VLAN.
- **inpkts**—Enables/disables destination port incoming packets.
- **learning**—Enables/disables MAC address learning.
- **multicast**—Enables/disables multicast traffic.
- **rx**—Receiving traffic.

- **session**— Session number for SPAN session.
- **tx** —Transmitting traffic.

To enable SPAN on the IDSM2, follow these steps:

Step 1 Log in to the console.

Step 2 Enter privileged mode.

```
console> enable
```

Step 3 Enable SPAN to the IDSM2:

- From a source port:

```
console> (enable) set span 3/3 13/7
Destination     : Port 13/7
Admin Source    : Port 3/3
Oper Source     : Port 3/3
Direction       : transmit/receive
Incoming Packets: disabled
Learning        : enabled
Multicast       : enabled
Filter          : -

Session Number  : 1

console> (enable)
```

Note Use the **filter** keyword to monitor traffic on specific VLANs on source trunk ports.

- From a VLAN:

```
console> (enable) set span 650 13/7 rx
Destination     : Port 13/7
Admin Source    : VLAN 650
Oper Source     : Port 11/1,13/1
Direction       : receive
Incoming Packets: disabled
Learning        : enabled
Multicast       : enabled
Filter          : -

Session Number  : 1


console> (enable)
```

Step 4 Show the SPAN sessions.

```
console> (enable) show span
Destination     : Port 13/7
Admin Source    : VLAN 650
Oper Source     : Port 11/1,13/1
Direction       : receive
Incoming Packets: disabled
Learning        : enabled
Multicast       : enabled
Filter          : -

Session Number  : 1
```

```
Total local span sessions:  1
console> (enable)
```

Step 5 To disable the SPAN session that is sending traffic to the IDSM2:

```
console> (enable) set span disable session 1
This command will disable your span session.
Do you want to continue (y/n) [n]? y
Disabled Port 13/7 to monitor receive traffic of VLAN 650
console> (enable)
```

For More Information

For more information on SPAN, refer to the appropriate *Catalyst 6500 Series Switch Command Reference*.

Cisco IOS Software

Note Use 1 or 2 for the IDSM2 data port numbers.

Use the **monitor session** command in global configuration mode to enable SPAN on the IDSM2.

The following options apply:

- **interface**—SPAN source interface
- **remote**—SPAN source Remote
- **vlan**— SPAN source VLAN
- **GigabitEthernet**— GigabitEthernet IEEE 802.3z
- **Port-channel**— Ethernet Channel of interfaces
- **,**— Specify another range of interfaces
- **-**— Specify a range of interfaces
- **both**— Monitor received and transmitted traffic
- **rx**— Monitor received traffic only
- **tx**— Monitor transmitted traffic only
- **intrusion-detection-module**— SPAN destination intrusion detection module
- **destination**— SPAN destination interface or VLAN
- **filter**— SPAN filter VLAN
- **source**— SPAN source interface, VLAN
- **type**— Type of monitor session

To enable SPAN on the IDSM2, follow these steps:

Step 1 Log in to the console.

Step 2 Enter global configuration mode.

```
router# configure terminal
```

Step 3 Set the source interfaces for the monitor session.

```
router(config)# monitor session (session_number) source interface interface/port_number
[, | - | rx | tx | both]
```

Example

```
router(config)# monitor session 1 source interface GigabitEthernet2/23 both
```

Step 4 Enable an IDSM2 data port as a SPAN destination.

```
router(config)# monitor session (session_number) destination intrusion-detection-module
module_number data-port data_port_number
```

Example

```
router(config)# monitor session 1 destination intrusion-detection-module 9 data-port 1
```

Step 5 Make sure autostate is included for the data port.

```
router(config)# intrusion-detection module module_number data-port data_port_number
autostate include
```

Example

```
router(config)# intrusion-detection module 9 data-port 1 autostate include
```

This allows the switch virtual interface to stay up if the data port is the only port in the VLAN. The default is **no include**.

Step 6 (Optional) Enable PortFast for the data port.

```
router(config)# intrusion-detection module module_number data-port data_port_number
portfast
```

Example

```
router(config)# intrusion-detection module 9 data-port 1 portfast
```

The default is disabled.

Step 7 (Optional) To disable the monitor session.

```
router(config)# no monitor session session_number
```

Step 8 (Optional) To filter the SPAN session so that only certain VLANs are seen from switch port trunks.

```
router(config)# monitor session (session_number) {filter vlan {vlan_ID} [, | - ]}
```

Example

```
router(config)# monitor session 1 filter vlan 146
```

Step 9 Exit configuration mode.

```
router(config)# exit
```

Step 10 To show current monitor sessions:

```
router# show monitor session session_number
```

Example

```
router# show monitor session 1
    Session 1
    ---------
    Type                   : Local Session
    Source Ports           :
```

```
    Both                : Gi2/23
Destination Ports       : intrusion-detection-module 9 data-port 1
```

For More Information

For more information on SPAN, autostate, and PortFast, refer to the appropriate *Catalyst 6500 Series Cisco IOS Command Reference*.

Configuring VACL Capture

You can set VACLs to capture traffic for IPS from a single VLAN or from multiple VLANs or from FLexWAN2 ports on the 7600 router when using Cisco IOS software. This section describes how to configure VACL capture, and contains the following topics:

- Catalyst Software, page 19-14
- Cisco IOS Software, page 19-16

Catalyst Software

Note Port 1 is set as the TCP reset port. Ports 7 and 8 are the sensing ports and can be configured as security ACL capture ports. By default, in Catalyst Software 8.4(1) and earlier releases, ports 7 and 8 are configured as trunk ports and trunk all VLANs on which a security ACL has been applied with the capture feature. To monitor traffic from specific VLANs only, you need to clear the VLANs that you do not want to monitor so that they are not trunked to ports 7 and 8.

Use the **set security acl** command to configure security ACL capture ports.

The following options apply:

- **ACL**—Sets security ACL features
  - **capture-port**—Sets ports for ACL capture
  - **cram**—Sets security ACL cram
  - **ip** —Sets IP security ACL features
  - **ipx**—Sets IPX security ACL features
  - **mac**—Sets MAC security ACL features
  - **map**— Sets security ACL to VLAN mapping
- **permit**—Specifies packets to forward
- **deny**—Specifies packets to reject
- **redirect**—Specifies packets to redirect to ports
- **before**—Inserts ACE before a specified ace in editbuffer
- **capture**—Makes a copy of this flow in capture ports
- **modify**— Modifies a specified ACE in editbuffer

To configure VACLs to capture IPS traffic on VLANs, follow these steps:

Step 1 Log in to the console.

Step 2 Enter privileged mode.

```
console> enable
```

Step 3 Create the VACL to capture traffic. Specify what traffic is permitted, denied, and captured.

```
console> (enable) set security acl ip acl_name permit ip [permit (...) | deny (...)]
capture
```

Note Only permitted traffic can be captured. To permit traffic but not capture it, do no use the **capture** keyword

Example

```
console> (enable) set security acl ip CAPTUREALL permit ip any any capture
CAPTUREALL editbuffer modified. Use 'commit' command to apply changes.
```

Step 4 Commit the VACL.

```
console> (enable) commit security acl CAPTUREALL
ACL commit in progress.
```

Committing the VACL writes the VACL and associated ACEs to NVRAM.

Step 5 Map the VACL to the VLANs.

```
console> (enable) set security acl map acl_name vlan_number
```

Example

```
console> (enable) set security acl map CAPTUREALL 650
Mapping in progress.

ACL CAPTUREALL successfully mapped to VLAN 650.
```

Step 6 Configure the IDSM2 ports (port 7 or 8) to be capture ports.

```
console> (enable) set security acl capture module_number/port_number
```

Example

```
console> (enable) set security acl capture 2/7
Successfully set 2/7 to capture ACL traffic.
```

For More Information

For more information on trunk ports and ACLs, refer to the appropriate *Catalyst 6500 Series Switch Command Reference*.

Cisco IOS Software

Use the following commands to configure VACLs to capture IPS traffic on VLANs.

The following options apply:

- **ip-access-list**—Named access list
  - **extended**—Extended Access List
  - **hardware**—Enable Hardware Fragment Handling
  - **log-update**—Control access list log updates
  - **logging**—Control access list logging
  - **resequence**—Resequence Access List
  - **standard**—Standard Access List

To configure VACLs to capture IPS traffic on VLANs, follow these steps:

Step 1 Log in to the console.

Step 2 Enter global configuration mode.

```
router# configure terminal
```

Step 3 Define the ACL.

```
router(config)# ip access-list [standard | extended] acl_name
```

Example

```
router(config)# ip access-list standard CAPTUREALL
router(config-std-nacl)# exit
```

Step 4 Define the VLAN access map.

```
router(config)# vlan access-map map_name [0-65535]
```

Step 5 Configure a match clause in a VLAN access map sequence.

```
router(config-access-map)# match [ip address {1-199 | 1300-2699 | acl_name}]
```

Step 6 Configure an action clause in the VLAN access map sequence to accompany the preceding match clause.

```
router(config-access-map)# action forward capture
```

Step 7 Apply the VLAN access-map to the specified VLANs.

```
router(config)# vlan filter map_name vlan-list vlan_list
```

Step 8 Configure the IDSM2 data ports to capture the captured-flagged traffic.

```
router(config)# intrusion-detection module module_number data-port data_port_number
capture allowed-vlan capture_vlans
```

Note When the switch is routing traffic, you should configure the IDSM2 to monitor all VLANs being routed. If you apply the VACL to a FlexWan2 port, you need to configure the IDSM2 to monitor all VLANs.

Step 9 Enable the capture function on the IDSM2.

```
router(config)# intrusion-detection module module_number data-port data_port_number
capture
```

This example shows the output from the **show run** command:

```
router# show run
intrusion-detection module 4 data-port 1 capture allowed-vlan 450,1002-1005
intrusion-detection module 4 data-port 1 capture
.
.
.
vlan access-map CAPTUREALL 10
match ip address MATCHALL
action forward capture
.
.
.
ip access-list extended MATCHALL
permit ip any any
router#
```

Step 10 Make sure autostate is included for the data port.

```
router(config)# intrusion-detection module module_number data-port data_port_number
autostate include
```

Example

```
router(config)# intrusion-detection module 4 data-port 1 autostate include
```

This allows the switch virtual interface to stay up if the data port is the only port in the VLAN. The default is **no include**.

Step 11 (Optional) Enable PortFast for the data port.

```
router(config)# intrusion-detection module module_number data-port data_port_number
portfast
```

Example

```
router(config)# intrusion-detection module 4 data-port 1 portfast
```

The default is disabled.

For More Information

For more information on autostate and PortFast, refer to the appropriate *Catalyst 6500 Series Cisco IOS Command Reference*.

Configuring the mls ip ids Command

This section describes how to use the **mls ip ids** command to capture IPS traffic. It contains the following topics:

- Catalyst Software, page 19-18
- Cisco IOS Software, page 19-19

Catalyst Software

When you are running the Cisco IOS Firewall on the MSFC, you cannot use VACLs to capture traffic for the IDSM2, because you cannot apply VACLs to a VLAN in which you have applied an IP inspect rule for the Cisco IOS Firewall. However, you can use the **mls ip ids** command to designate which packets are captured. Packets that are permitted by the ACL are captured. Those denied by the ACL are not captured. The **permit/deny** parameter does not affect whether a packet is forwarded to destination ports. Packets coming into that router interface are checked against the IPS ACL to determine if they should be captured. The **mls ip ids** command is applied as part of the MSFC configuration instead of the supervisor configuration. The **mls ip ids** command only captures incoming traffic. Use the **mls ip ids** command on both the client-side router interface and server-side router interface, so that both directions of the connection are captured.

To use the **mls ip ids** command to capture IPS traffic, follow these steps:

Step 1 Log in to the MSFC.

Step 2 Enter privileged mode.

```
console> enable
```

Step 3 Enter configuration mode.

```
Router# configure terminal
```

Step 4 Configure an ACL to designate which packets will be captured.

```
Router(config)# ip access-list extended word
```

Step 5 Select the interface that carries the packets to be captured.

```
Router(config)# interface interface_name
```

Step 6 Apply the ACL created in Step 4 to the interface selected in Step 5.

```
Router(config-if)# mls ip ids word
```

Step 7 Log in to the supervisor engine.

Step 8 Enter privileged mode.

```
console> enable
```

Step 9 On the supervisor engine, add the IDSM2 monitoring port (port 7 or 8) to the VACL capture list.

```
console> (enable) set security acl capture module_number/port_number
```

Caution For the IDSM2 to capture all packets marked by the **mls ip ids** command, port 7 or 8 of the IDSM2 must be a member of all VLANs to which those packets are routed.

Cisco IOS Software

When you are using ports as router interfaces rather than switch ports, there is no VLAN on which to apply a VACL.

You can use the **mls ip ids** command to designate which packets are captured. Packets that are permitted by the ACL are captured. Those denied by the ACL are not captured. The **permit/deny** parameter does not affect whether a packet is forwarded to destination ports. Packets coming into that router interface are checked against the IPS ACL to determine if they should be captured.

To use the **mls ip ids** command to capture IDS traffic, follow these steps:

Step 1 Log in to the console.

Step 2 Enter global configuration mode.

```
router# configure terminal
```

Step 3 Configure an ACL to designate which packets will be captured.

```
router(config)# ip access-list extended word
```

Step 4 Select the interface that carries the packets to be captured.

```
router(config)# interface interface_name
```

Step 5 Specify the capture VLANs.

```
router(config)# intrusion-detection module module_number data-port data_port_number
capture allowed-vlan capture_vlans
```

Example

```
router(config)# intrusion-detection module 4 data-port 1 capture allowed-vlan 165
```

Step 6 Apply the ACL created in Step 4 to the interface selected in Step 5.

```
router(config-if)# mls ip ids word
```

Caution For the IDSM2 to capture all packets marked by the **mls ip ids** command, data port 1 or data port 2 of the IDSM2 must be a member of all VLANs to which those packets are routed.

Configuring the Catalyst Series 6500 Switch for the IDSM2 in Inline Mode

This section describes how to configure the switch so that the IDSM2 operates in inline mode, and contains the following topics:

Understanding the Switch, the IDSM2, and Inline Mode

You can use IDM or the CLI to configure the IDSM2 to operate in inline mode between two separate VLANs (one VLAN for each side of the IDSM2). To prepare the IDSM2 for inline mode, you must configure the switch as well as the IDSM2. Configure the switch first, then configure the IDSM2 interfaces for inline mode.

For More Information

For the procedure for configuring the IDSM2 to run in inline mode, see Configuring Inline Interface Pairs, page 5-16.

Catalyst Software

You configure the IDSM2 monitoring ports as trunk ports for inline operation for Catalyst software 8.4(1) or later with Supervisor Engine 1a, Supervisor Engine 2, Supervisor Engine 32, or Supervisor Engine 720. Because the native VLAN is the same as the sole VLAN being trunked, the traffic is not 802.1q encapsulated.

Caution Before Catalyst software 8.4.(3), the default configuration for the IDSM2 ports 7 and 8 is to trunk all VLANs 1 to 4094. If you clear the IDSM2 configuration (**clear configuration** *module_number*), the IDSM2 trunks all VLANs. If the IDSM2 interfaces are configured for inline, spanning tree loops will likely be created and a storm will occur. A storm is numerous packets looping and never reaching their destination.

To configure the monitoring ports on the IDSM2 for inline operation, follow these steps:

Step 1 Log in to the console.

Step 2 Enter privileged mode.

```
console> enable
```

Step 3 Set the native VLAN for each IDSM2 monitoring port:

```
console (enable)> set vlan vlan_number slot_number/port_number
```

Example

```
console (enable)> set vlan 651 9/7
console (enable)> set vlan 652 9/8
```

Step 4 Clear all VLANs from each IDSM2 monitoring port (port 7 and 8).

```
console (enable)> clear trunk slot_number/port_number vlan_range
```

Example

```
console (enable)> clear trunk 9/7 1-4094
console (enable)> clear trunk 9/8 1-4094
```

Step 5 Configure the IDSM2 monitoring ports 7 and 8 to trunk the single native VLAN configured in Step 3.

```
console (enable)> set trunk slot_number/port_number vlan_number
```

Example

```
console (enable)> set trunk 9/7 651
```

```
console (enable)> set trunk 9/8 652
```

Step 6 Pair the interfaces from Step 3 on the IDSM2.

For More Information

For the procedure for pairing inline interface pairs, see Configuring Inline Interface Pairs, page 5-16.

Cisco IOS Software

Configure the IDSM2 monitoring ports as access ports for inline operation. To configure inline VLANs, follow these steps:

Step 1 Log in to the console.

Step 2 Enter global configuration mode.

```
router# configure terminal
```

Step 3 Select the VLANs the IDSM2 will link.

Step 4 Configure each IDSM2 data port to be on a single VLAN.

```
router(config)# intrusion-detection module slot_number data-port {1 | 2} access-vlan
vlan_number
router(config)# exit
```

Example

```
router(config)# intrusion-detection module 13 data-port 1 access-vlan 661
router(config)# intrusion-detection module 13 data-port 2 access-vlan 662
router(config)# exit
```

Step 5 Verify the configuration:

Note In these examples, the IDSM2 in slot 13 is inline between VLANs 661 and 662. The IDSM2 data port 1 is on VLAN 661 and data port 2 is on VLAN 662.

a. Verify the IDSM2 intrusion detection settings.

```
router# show run | include intrusion-detection
intrusion-detection module 13 management-port access-vlan 147
intrusion-detection module 13 data-port 1 access-vlan 661
intrusion-detection module 13 data-port 2 access-vlan 662
router#
```

b. Verify that the IDSM2 data port 1 is an access port on VLAN 661.

```
router# show intrusion-detection module slot_number data-port data_port_number state
```

Example

```
router# show intrusion-detection module 13 data-port 1 state
Intrusion-detection module 13 data-port 1:

Switchport: Enabled
Administrative Mode: static access
Operational Mode: static access
```

```
Administrative Trunking Encapsulation: dot1q Operational Trunking Encapsulation:
native Negotiation of Trunking: Off Access Mode VLAN: 661 (inline-vlan-1) Trunking
Native Mode VLAN: 1 (default) Trunking VLANs Enabled: NONE Pruning VLANs Enabled:
2-1001 Vlans allowed on trunk:661 Vlans allowed and active in management domain: 661
Vlans in spanning tree forwarding state and not pruned: 661
Administrative Capture Mode: Disabled
Administrative Capture Allowed-vlans: <empty>
```

c. Verify the VLAN number.

```
router# show vlan id vlan-number
```

Example

```
router# show vlan id 661
VLAN Name                             Status    Ports
---- -------------------------------- --------- -------------------------------
661  ward-attack3                     active    Gi3/2, Gi13/d1

VLAN Type  SAID       MTU   Parent RingNo BridgeNo Stp  BrdgMode Trans1 Trans2
---- ----- ---------- ----- ------ ------ -------- ---- -------- ------ ------
661  enet  100661     1500  -      -      -        -    -        0      0

Remote SPAN VLAN
----------------
Disabled

Primary Secondary Type              Ports
------- --------- ----------------- ------------------------------------------
router#
```

Step 6 Pair the interfaces from Step 4 on the IDSM2.

For More Information

For the procedure for pairing inline interface pairs, see Configuring Inline Interface Pairs, page 5-16.

Configuring the Catalyst Series 6500 Switch for the IDSM2 for Inline VLAN Pair Mode

This section describes how to configure the switch so that the IDSM2 operates in inline VLAN pair mode, and contains the following topics:

Understanding the Switch, the IDSM2, and Inline VLAN Pair Mode

You can use IDM or the CLI to configure the IDSM2 to operate in inline VLAN pair mode. To prepare the IDSM2 for inline VLAN pair mode, you must configure the switch as well as the IDSM2. Configure the switch first, then configure the IDSM2 interfaces for inline VLAN pair mode.

For More Information

For the procedure for configuring the IDSM2 to run in inline VLAN pair mode, see Configuring Inline VLAN Pairs, page 5-21.

Catalyst Software

You configure the IDSM2 monitoring ports as trunk ports for inline VLAN pair mode for Catalyst software 8.4(1) or later with Supervisor Engine 1a, Supervisor Engine 2, Supervisor Engine 32, or Supervisor Engine 720.

To configure the monitoring ports on the IDSM2 for inline VLAN pair mode, follow these steps:

Step 1 Log in to the console.

Step 2 Enter privileged mode.

```
console> enable
```

Step 3 Clear all VLANs from the IDSM2 monitoring port.

```
console (enable)> clear trunk slot_number/port_number 1-4094
```

Example

```
console (enable)> clear trunk 9/7 1-4094
```

Note Before Catalyst software 8.4.(3), the value for the VLAN range when clearing VLANs from the IDSM2 monitoring port was 1-1005, 1024-4094. In later versions you can clear the entire VLAN range, 1-4094.

Step 4 Configure the IDSM2 monitoring port to trunk the VLANs to be paired.

```
console (enable)> set trunk slot_number/port_number vlans_to_be_paired
```

Example

```
console (enable)> set trunk 9/7 651,652
```

Step 5 Set the native VLAN for the IDSM2 monitoring port to a value other than the paired VLANs used in Step 4.

```
console (enable)> set vlan vlan-number slot_number/port_number
```

Example:

```
console (enable)> set vlan 1 9/7
```

The default native VLAN is VLAN 1.

Step 6 Repeat Step 4 for other VLANs to be paired on the IDSM2 monitoring port.

Step 7 To configure the other monitoring port, repeat Steps 3 through 6.

Step 8 Pair the VLANs from Step 4 on the IDSM2.

For More Information

For the procedure for configuring the IDSM2 to run in inline VLAN pair mode, see Configuring Inline VLAN Pairs, page 5-21.

Cisco IOS Software

Configure the IDSM2 monitoring ports as trunk ports for inline VLAN pair operation.

To configure inline VLAN pairs, follow these steps:

Step 1 Log in to the console.

Step 2 Enter global configuration mode.

```
router# configure terminal
```

Step 3 Configure one IDSM2 data port to trunk the VLANs to be paired.

```
router(config)# intrusion-detection module slot_number data-port data_port_number trunk
allowed-vlan vlans_to_be_paired
router(config)# exit
```

Example

```
router(config)# intrusion-detection module 13 data-port 1 trunk allowed-vlan 661,662
router(config)# exit
```

Step 4 Verify the configuration:

Note In these examples, data port 1 of the IDSM2 in slot 13 is trunking VLANs 661 and 662.

a. Verify the IDSM2 intrusion detection settings.

```
router# show run | include intrusion-detection
intrusion-detection module 13 management-port access-vlan 147
intrusion-detection module 13 data-port 1 trunk allowed-vlan 661,662
router#
```

b. Verify that the IDSM2 data port is trunking the proper VLANs.

```
router# show intrusion-detection module slot_number data-port data_port_number state
```

Example:

```
router# show intrusion-detection module 13 data-port 1 state
Intrusion-detection module 13 data-port 1:

Switchport: Enabled
Administrative Mode: trunk
Operational Mode: trunk
Administrative Trunking Encapsulation: dot1q
Operational Trunking Encapsulation: dot1q
Negotiation of Trunking: Off
Access Mode VLAN: 1 (default)
Trunking Native Mode VLAN: 1 (default)
Trunking VLANs Enabled: 661,662
Pruning VLANs Enabled: 2-1001
Vlans allowed on trunk: 661-662
Vlans allowed and active in management domain: none
Vlans in spanning tree forwarding state and not pruned:
```

```
   none
Administrative Capture Mode: Disabled
Administrative Capture Allowed-vlans: empty
Autostate mode: excluded
Portfast mode: default

router#
```

Step 5 Pair the VLANs from Step 3 on the IDSM2.

For More Information

For the procedure for configuring the IDSM2 to run in inline VLAN pair mode, see Configuring Inline VLAN Pairs, page 5-21.

Configuring EtherChannel Load Balancing

This section describes how to configure ECLB on the IDSM2, and contains the following topics:

Understanding EtherChannel Load Balancing

Supervisor Engines in the Catalyst 6500 series chassis recognize IDSM2 devices that are running IPS 5.*x* and greater as EtherChannel devices. This lets you install up to eight IDSM2 devices in the same chassis.

The IDSM2 in the Catalyst 6500 series switch has eight internal ports. Only four of these ports are used. Port 1 is a TCP/IP reset port. Port 2 is the command and control port. Ports 7 and 8 are the sensing ports for Catalyst software and data ports 1 and 2 for Cisco IOS software. The other ports are not used.

The backplane is 1000 Mbps, which is why the IDSM2 shows 1000 Mbps even though it can only handle about 600 Mbps of performance. ECLB allows up to eight IDSM2 devices to participate in the load balancing on either port 7 or port 8.

EtherChannel and the Three Sensing Modes

EtherChannel provides load balancing and failover between multiple IDSM2s in all three sensing modes. The IDSM2 does not participate in EtherChannel protocols, such as LACP or PAgP. Cisco IOS only allows load balancing using **src-dst-ip** algorithm so that all packets between a given pair of IP addresses are always mapped to the same channel. Catalyst software uses the **ip both** algorithm. This is necessary so the IDSM2 can correctly track the connections between two hosts.

Caution You cannot mix IDSM2 data ports with other port types in an EtherChannel group. You must configure all data ports in an EtherChannel group identically.

EtherChannel and the IDSM2 operate in the following way in the three sensing modes:

- EtherChannel and promiscuous mode—When the IDSM2 operates in promiscuous mode, the two data ports operate independently of each other. If you configure the switch so that a data port has two or more IDSM2s in a group, the switch distributes traffic between the IDSM2s. This balances the traffic between multiple IDSM2s. You should rebalance the channel when a data port goes to the `errDisabled` state, or the IDSM2 is shut down, powered down, or reset.
- EtherChannel and inline mode—When you configure multiple IDSM2s for inline mode, you can load balance the traffic between the IDSM2s by putting data port 1 of each IDSM2 into one channel group and data port 2 of each IDSM2 into another channel group.

Caution To make sure that the same traffic is assigned to the two data ports on each IDSM2, you must assign the same EtherChannel index to both data ports on each of the IDSM2s even though they are in different EtherChannel groups.

- EtherChannel and inline VLAN pair mode—When the IDSM2 is in inline on-a-stick mode, the two data ports operate independently of each other. The same restrictions apply as for promiscuous mode.

Enabling ECLB

This section describes how to enable ECLB for Cisco IOS and Catalyst software. It contains the following sections:

- Catalyst Software, page 19-26
- Cisco IOS Software, page 19-28

Catalyst Software

This section describes how to enable ECLB for the three sensing modes in Catalyst software. It contains the following topics:

- ECLB in Promiscuous and VLAN Pair mode, page 19-26
- ECLB in Inline Mode, page 19-27

ECLB in Promiscuous and VLAN Pair mode

For promiscuous mode and inline VLAN pair mode, add the single port (port 7 or port 8) from each IDSM2 to an EtherChannel.

To configure the monitoring ports on the IDSM2 for ECLB in promiscuous or inline VLAN pair mode, follow these steps:

Step 1 Log in to the console.

Step 2 Enter privileged mode.

```console> enable```

Step 3 Configure each IDSM2 for promiscuous or inline VLAN pair mode.

Step 4 Add the IDSM2 monitoring ports to an EtherChannel.

```console (enable)> set port channel slot_number/port_number channel_number```

Example

```
console (enable)> set port channel 1/7,7/7 1
```

Step 5 Set the distribution method.

```
console (enable)> set port channel all distribution ip both
Channel distribution is set to ip both.
console (enable)>
```

Step 6 Enable ECLB.

console (enable)> **set port channel** *slot_number/port_number* **mode on**

Example

```
console (enable)> set port channel 1/7,7/7 mode on
```

For More Information

For the procedure for configuring inline VLAN pairs, see Configuring Inline VLAN Pairs, page 5-21.

ECLB in Inline Mode

For inline mode, add the single port 7 from each IDSM2 to an EtherChannel and port 8 from each IDSM2 to a different EtherChannel.

To configure the monitoring ports on the IDSM2 for ECLB in inline mode, follow these steps:

Step 1 Log in to the console.

Step 2 Enter privileged mode.

console> **enable**

Step 3 Configure each IDSM2 for inline mode.

Step 4 Add the IDSM2 monitoring port 7s to an EtherChannel.

console (enable)> **set port channel** *slot_number/***7** *channel_A_number*

Example

```
console (enable)> set port channel 1/7,7/7 1
```

Step 5 Enable ECLB for that EtherChannel.

console (enable)> **set port channel** *slot_number/***7 mode on**

Example

```
console (enable)> set port channel 1/7,7/7 mode on
```

Step 6 Add the IDSM2 monitoring port 8s to another EtherChannel.

console (enable)> **set port channel** *slot_number/***8** *channel_B_number*

Example

```
console (enable)> set port channel 1/8,7/8 2
```

Step 7 Enable ECLB for that EtherChannel.

console (enable)> **set port channel** *slot_number/***8 mode on**

Example

```
console (enable)> set port channel 1/8,7/8 mode on
```

Step 8 Set the distribution method.

```
console (enable)> set port channel all distribution ip both
Channel distribution is set to ip both.
console (enable)>
```

For More Information

For the procedure for configuring inline VLAN pairs, see Configuring Inline VLAN Pairs, page 5-21.

Cisco IOS Software

Note IOS 12.2(18)SXF4 or later is required for inline mode.

This section describes how to enable ECLB for the three sensing modes in Cisco IOS software. It contains the following topics:

Restoring Defaults

Use the **intrusion-detection module** *module_number* **data-port {1 | 2} default** command to restore the defaults to the specified data port. This command restores the following defaults: allowed VLANs, autostate, portfast, cost, and priority settings. If the data port belongs to a port channel, this command has no effect. This command is useful for clearing the data port before you add it to a port channel group.

This command is equivalent to using all of the following commands:

- **no intrusion-detection module** *module_number* **data-port {1 | 2} trunk allowed-vlan**
- **intrusion-detection module** *module_number* **data-port {1 | 2} access vlan**
- **intrusion-detection module** *module_number* **data-port {1 | 2} autostate include**
- **intrusion-detection module** *module_number* **data-port {1 | 2} portfast**
- **intrusion-detection module** *module_number* **data-port {1 | 2} spanning-tree cost**
- **intrusion-detection module** *module_number* **data-port {1 | 2} spanning-tree priority**

ECLB in Promiscuous Mode

Note For Cisco IOS version and supervisor requirements for EtherChannel load balancing on the IDSM2, see Table 19-1.

Note Cisco IOS supports promiscuous IDSM2 EtherChannel using VACL capture (not SPAN or monitor).

An EtherChannel balances the traffic load across the links in an EtherChannel by reducing part of the binary pattern formed from the addresses in the frame to a numerical value that selects one of the links in the channel.

EtherChannel load balancing can use MAC addresses, IP addresses, or Layer 4 port numbers, which can be source or destination or both source and destination addresses or ports. The selected mode applies to all EtherChannels configured on the switch. ECLB can also use MPLS Layer 2 information.

Use the option that provides the balance criteria with the greatest variety in your configuration. For example, if the traffic on an EtherChannel is going only to a single MAC address and you use the destination MAC address as the basis of ECLB, the EtherChannel always chooses the same link in the EtherChannel; using source addresses or IP addresses might result in better load balancing.

To configure ECLB for promiscuous operation on the IDSM2, follow these steps:

Step 1 Configure each IDSM2 for promiscuous operation.

Note Make sure that all IDSM2 VACL capture or SPAN or monitor configuration lines have been removed before configuring ECLB for the IDSM2.

Step 2 Log in to the console.

Step 3 Enter global configuration mode.

```
Router# configure terminal
```

Step 4 Create the VACL.

```
Router(config)# ip access-list extended vacl_name
```

Example

```
Router(config)# ip access-list extended idstest
```

Step 5 Add any access control entries, for example, `permit any any`.

```
Router(config-ext-nacl)# permit ip any any
```

Step 6 Create at least one VLAN access map sequence.

```
Router(config-ext-nacl)# vlan access-map vlan_access_map_name sequence_number
Router(config-access-map)# match ip address vacl_name
Router(config-access-map)# action forward capture
```

Example

```
Router(config)# vlan access-map idstestmap 10
Router(config-access-map)# match ip address idstest
Router(config-access-map)# action forward capture
```

Step 7 Apply the VLAN access map to the VLAN(s).

```
Router(config-access-map)# vlan filter vlan_access_map_name vlan-list vlan-list
```

Example

```
Router(config)# vlan filter idstestmap vlan-list 50-60
```

Step 8 For each IDSM2, add the desired data ports into the desired EtherChannel.

```
Router(config)# intrusion-detection module module_number data-port data_port_number
channel-group channel_number
```

Example

```
Router(config)# intrusion-detection module 13 data-port 1 channel-group 3
Router(config)# intrusion-detection module 12 data-port 1 channel-group 3
```

Each EtherChannel has a numbered port channel interface. You can configure a maximum of 64 port channel interfaces, numbered from 1 to 256.

Step 9 Configure ECLB.

```
Router(config)# port-channel load-balance src-dst-ip
```

The default and only load balancing algorithm supported for the IDSM2 is **src-dst-ip**, which means EtherChannel uses the combination of source and destination IP addresses for its distribution method.

Step 10 Verify the load balancing.

```
Router# show etherchannel load-balance
EtherChannel Load-Balancing Configuration:
        src-dst-ip

EtherChannel Load-Balancing Addresses Used Per-Protocol:
Non-IP: Source XOR Destination MAC address
  IPv4: Source XOR Destination IP address
  IPv6: Source XOR Destination IP address
  MPLS: Label or IP
```

Step 11 Set the VLANs to be captured to the EtherChannel.

```
Router(config)# intrusion-detection port-channel channel_number capture allowed-vlan
vlan_list
```

Example

```
Router(config)# intrusion-detection port-channel 3 capture allowed-vlan 10
```

Step 12 Enable capture to the EtherChannel.

```
Router(config)# intrusion-detection port-channel channel_number capture
```

Example

```
Router(config)# intrusion-detection port-channel 3 capture
```

Step 13 Make sure autostate is included for the channel group.

```
Router(config)# intrusion-detection port-channel channel_number autostate include
```

Example

```
Router(config)# intrusion-detection port-channel 3 autostate include
```

This allows the switch virtual interface to stay up if the data port is the only port in the VLAN. The default is **no include**.

Step 14 (Optional) Enable PortFast for the channel group.

```
Router(config)# intrusion-detection port-channel channel_number portfast enable
```

Example

```
Router(config)# intrusion-detection port-channel 3 portfast enable
```

The default is disabled.

Step 15 Exit global configuration mode.

```
Router(config)# exit
```

Step 16 To save the changes.

```
Router# write memory
```

For More Information

- For more information on EtherChannel, refer to *Catalyst 6500 Release 12.2SXF and Rebuilds Software Configuration Guide*.
- For more information on autostate and PortFast, refer to the appropriate *Catalyst 6500 Series Cisco IOS Command Reference*.
- For the procedure for configuring the IDSM2 in promiscuous mode, see Configuring the Catalyst Series 6500 Switch for the IDSM2 in Promiscuous Mode, page 19-9.

ECLB in Inline Mode

Note Make sure that all IDSM2 VACL capture or SPAN or monitor configuration lines have been removed before configuring ECLB for the IDSM2. You receive an error if you try to change the channel group to inline mode if you have capture enabled on any of the ports.

To configure ECLB for inline mode on the IDSM2, follow these steps:

Step 1 Log in to the console.

Step 2 Enter global configuration mode.

```
router# configure terminal
```

Step 3 For each IDSM2, add all data port 1s into an EtherChannel.

```
router(config)# intrusion-detection module module_number data-port 1 port-channel
channel_number
```

Example

```
router(config)# intrusion-detection module 1 data-port 1 port-channel 5
```

Each EtherChannel has a numbered port channel interface. You can configure a maximum of 64 port channel interfaces, numbered from 1 to 256. If the channel group and port channel have not been created, this command creates it with an empty allowed VLAN list. If the port channel exists, its allowed VLAN list, port fast, autostate, spanning tree cost, and priority settings are assigned to the data port.

Note You receive an error if you try to add a data port to a channel group that contains other port types or if you try to add another port type to a port channel containing one or more data ports.

Step 4 For each IDSM2, add all data port 2s into a different EtherChannel.

```
router(config)# intrusion-detection module module_number data-port 2 port-channel
channel_number
```

Example

```
router(config)# intrusion-detection module 1 data-port 2 port-channel 6
```

Step 5 Set the sensing mode to access (inline) and set the access VLAN for the channel group that contains the data port 1s.

```
router(config)# intrusion-detection port-channel channel_number access-vlan vlan_id
```

Example

```
router(config)# intrusion-detection port-channel 5 access-vlan 1050
```

Note You receive an error message if the port channel does not exist or if the port channel is already configured for trunk or capture mode. You must create the port channel or remove the port channel from trunk or capture mode.

Step 6 Set the sensing mode to access (inline) and set the access VLAN for the channel group that contains the data port 2s.

```
router(config)# intrusion-detection port-channel channel_number access-vlan vlan_id
```

Example

```
router(config)# intrusion-detection port-channel 6 access-vlan 10
```

Step 7 Configure ECLB:.

```
router(config)# port-channel load-balance src-dst-ip
```

The default is **src-dst-ip**, which means EtherChannel uses the combination of source and destination IP addresses for its distribution method.

Example

```
router(config)# port-channel load-balance src-dst-ip
```

Step 8 Verify ECLB.

```
router# show etherchannel load-balance
EtherChannel Load-Balancing Configuration:
        src-dst-ip

EtherChannel Load-Balancing Addresses Used Per-Protocol:
Non-IP: Source XOR Destination MAC address
  IPv4: Source XOR Destination IP address
  IPv6: Source XOR Destination IP address
  MPLS: Label or IP
```

Step 9 For access (inline) mode, set autostate to **include** each channel group.

```
router(config)# intrusion-detection port-channel channel_number autostate include
```

Example

```
router(config)# intrusion-detection port-channel 5 autostate include
```

The default is **no include**. This prevents the switch virtual interface from going down if the data port is up and in the VLAN.

Step 10 (Optional) You can enable or disable PortFast for each channel group.

```
router(config)# intrusion-detection port-channel channel_number portfast enable
```

Example

```
router(config)# intrusion-detection port-channel 5 portfast enable
```

The default is disabled.

Step 11 (Optional) Set the spanning tree path cost for each of the two channel groups.

```
router(config)# intrusion-detection port-channel channel_number spanning-tree cost
port_cost
```

Example

```
router(config)# intrusion-detection port-channel 5 spanning-tree cost 4
```

Both channel groups must be set to the same port cost to make sure that data port 1 and data port 2 of each IDSM2 are in the same state (forwarding versus blocking).

Step 12 (Optional) Set the spanning tree port priority for each of the two channel groups.

```
router(config)# intrusion-detection port-channel channel_number spanning-tree priority
priority
```

Example

```
router(config)# intrusion-detection port-channel 5 spanning-tree priority 16
```

The possible port priority value is a multiple of 16 from 0 to 240. The default is 32.

Step 13 Exit global configuration mode.

```
router(config)# exit
```

Step 14 To save the changes.

```
router# write memory
```

For More Information

- For more information on autostate and PortFast, refer to the appropriate *Catalyst 6500 Series Cisco IOS Command Reference*.
- For the procedure for configuring the IDSM2 in inline mode, see Configuring the Catalyst Series 6500 Switch for the IDSM2 in Inline Mode, page 19-19.

ECLB in Inline VLAN Pair Mode

Note Make sure that all IDSM2 VACL capture or SPAN or monitor configuration lines have been removed before configuring ECLB for the IDSM2. You receive an error if you try to change the channel group to inline VLAN pair mode if you have **capture** enabled on any of the ports.

To configure ECLB for inline VLAN pair mode on the IDSM2, follow these steps:

Step 1 Log in to the console.

Step 2 Enter global configuration mode:.

```
router# configure terminal
```

Step 3 Add the data port (either data port 1 or data port 2) from each IDSM2 to the Etherchannel.

```
router(config)# intrusion-detection module module_number data-port [1:2] port-channel
channel_number
```

Example

```
router(config)# intrusion-detection module 1 data-port 1 port-channel 5
```

Each EtherChannel has a numbered port channel interface. You can configure a maximum of 64 port channel interfaces, numbered from 1 to 256. If the channel group and port channel have not been created, this command creates it with an empty allowed VLAN list. If the port channel exists, its allowed VLAN list, port fast, autostate, spanning tree cost, and priority settings are assigned to the data port.

Note You receive an error if you try to add a data port to a channel group that contains other port types or if you try to add another port type to a port channel containing one or more data ports.

Step 4 Set the sensing mode to trunk (inline VLAN pair) and set the allowed VLANs for the channel group that contains the data port 1s. Determine which VLANs are going to be paired (100 and 200, 101 and 201) and set the allowed VLAN list to include all VLANs in all the pairs.

```
router(config)# intrusion-detection port-channel channel_number trunk allowed-vlan
vlan_list
```

Example

```
router(config)# intrusion-detection port-channel 5 trunk allowed-vlans 100,101,200,201
```

Note The allowed VLAN list on the switch must include all VLANs that are paired as inline VLAN pairs on the IDSM2. Otherwise, traffic may be dropped.

Note You receive an error message if the port channel does not exist or if the port channel is already configured for trunk or capture mode. You must create the port channel or remove the port channel from trunk or capture mode.

Step 5 Configure ECLB.

```
router(config)# port-channel load-balance src-dst-ip
```

The default is **src-dst-ip**, which means EtherChannel uses the combination of source and destination IP addresses for its distribution method.

Example

```
router(config)# port-channel load-balance src-dst-ip
```

Step 6 Verify ECLB.

```
router# show etherchannel load-balance
```

```
EtherChannel Load-Balancing Configuration:
        src-dst-ip

EtherChannel Load-Balancing Addresses Used Per-Protocol:
Non-IP: Source XOR Destination MAC address
  IPv4: Source XOR Destination IP address
  IPv6: Source XOR Destination IP address
  MPLS: Label or IP
```

Step 7 For access (inline) mode, set autostate to **include** the channel group.

```
router(config)# intrusion-detection port-channel channel_number autostate include
```

Example

```
router(config)# intrusion-detection port-channel 5 autostate include
```

The default is **no include**. This prevents the switch virtual interface from going down if the data port is up and in the VLAN.

Step 8 (Optional) You can enable or disable PortFast for the channel group.

```
router(config)# intrusion-detection port-channel channel_number portfast enable
```

Example

```
router(config)# intrusion-detection port-channel 5 portfast enable
```

The default is disabled.

Step 9 (Optional) Set the spanning tree port cost for the channel group.

```
router(config)# intrusion-detection port-channel channel_number spanning-tree cost
port_cost
```

Example

```
router(config)# intrusion-detection port-channel 5 spanning-tree cost 4
```

Step 10 (Optional) Set the spanning tree port priority for the channel group.

```
router(config)# intrusion-detection port-channel channel_number spanning-tree priority
priority
```

Example

```
router(config)# intrusion-detection port-channel 5 spanning-tree priority 16
```

The possible port priority value is a multiple of 16 from 0 to 240. The default is 32.

Step 11 Exit global configuration mode.

```
router(config)# exit
```

Step 12 To save the changes.

```
router# write memory
```

Step 13 Pair the VLANs from Step 4 on the IDSM2.

For More Information

- For the procedure for pairing VLANs, see Configuring Inline VLAN Pairs, page 5-21.
- For more information on autostate, PortFast, spanning tree, and port priority, refer to the appropriate *Catalyst 6500 Series Cisco IOS Command Reference*.

Disabling ECLB

This section explains how to disable ECLB, and contains the following topics:

Catalyst Software

To disable ECLB, follow these steps:

Step 1 Log in to the console.

Step 2 Enter privileged mode.

```
console> enable
```

Step 3 Disable ECLB for promiscuous or inline VLAN pair mode.

```
console (enable)> set port channel slot_number/port_number mode off
```

Example

```
console (enable)> set port channel 1/7,7/7 mode off
```

Step 4 Disable ECLB for inline mode:

a. Disable ECLB for one EtherChannel.

```
console (enable)> set port channel slot_number/7 mode off
```

Example

```
console (enable)> set port channel 1/7,7/7 mode off
```

b. Disable ECLB for the other EtherChannel.

```
console (enable)> set port channel slot_number/8 mode off
```

Example

```
console (enable)> set port channel 1/8,7/8 mode off
```

Cisco IOS Software

To disable ECLB for the IDSM2, follow these steps:

Step 1 Log in to the console.

Step 2 Enter global configuration mode.

```
Router# configure terminal
```

Step 3 To remove a single IDSM2 from the EtherChannel:

```
Router(config)# no intrusion-detection module module_number data-port data_port_number
channel-group channel_number
```

Example

```
Router(config)# no intrusion-detection module 1 data-port 1 channel-group 5
```

Step 4 To remove the whole EtherChannel:

```
Router(config)# no intrusion-detection module port-channel channel_number
```

Example

```
Router(config)# no intrusion-detection module port-channel 5
```

Note The VACL capture commands for the IDSM2 are left.

Verifying ECLB

This section explains how to verify your ECLB configuration, and contains the following topics:

- Catalyst Software, page 19-37
- Cisco IOS Software, page 19-38

Catalyst Software

To verify the ECLB configuration, follow these steps:

Step 1 Log in to the console.

Step 2 Enter privileged mode.

```
console> enable
```

Step 3 To see all EtherChannels:

```
console (enable)> show channel slot_number/port_number mode off
```

Example

```
console> (enable) show channel
Channel Id   Ports
-----------  ------------------------------------------------
```

```
1669         1/7,7/7
1698         2/1-6
console> (enable)
```

Note In this output, an EtherChannel with ID 1669 is created to have two IDSM2 data ports. Port 1/7 is for port 7 on the IDSM2 in slot 1 while port 7/7 is for port 7 on the IDSM2 in slot 7. Both IDSM2s are configured for promiscuous operation. The switch load balances between each of the two IDSM2 ports (one port on each IDSM2).

Step 4 To see specific EtherChannel status:

console (enable)> **show channel hash** *channel_id source_ip_addr dest_ip_addr*

Example

```
console> (enable) show channel hash 1669 10.20.2.1 10.20.5.3
Selected channel port: 1/7
console> (enable)
```

Note This output shows that traffic from 10.20.2.1 to 10.20.5.3 will be sent to port 1/7 (port 7 for the IDSM2 in slot 1).

Cisco IOS Software

To verify the IDSM2 ECLB configuration, follow these steps:

Step 1 Log in to the console.

Step 2 To see all EtherChannels:

```
router# show etherchannel
                Channel-group listing:
                ----------------------

Group: 10
----------
Group state = L2
Ports: 0   Maxports = 8
Port-channels: 1 Max Port-channels = 1
Protocol:    -

router#
```

Step 3 To see specific EtherChannel status:

router# **show etherchannel 1** [**summary** | **detail** | **port** | **port-channel** | **protocol**]

Example

```
router# show etherchannel 1 summary
Flags:  D - down        P - in port-channel
        I - stand-alone s - suspended
        H - Hot-standby (LACP only)
        R - Layer3      S - Layer2
        U - in use      f - failed to allocate aggregator
```

```
        u - unsuitable for bundling
Number of channel-groups in use: 1
Number of aggregators:           1

Group  Port-channel  Protocol    Ports
------+-------------+-----------+----------------------------

router#
```

Step 4 To see the ECLB setting:

```
router# show etherchannel load-balance
EtherChannel Load-Balancing Configuration:
        src-dst-ip
        mpls label-ip

EtherChannel Load-Balancing Addresses Used Per-Protocol:
Non-IP: Source XOR Destination MAC address
  IPv4: Source XOR Destination IP address
  IPv6: Source XOR Destination IP address
  MPLS: Label or IP
router#
```

Step 5 To see the IDSM2 data port information:

```
router# show intrusion-detection module module_number data-port data_port_number state
```

Example

```
router# show intrusion-detection module 11 data-port 2 state
Intrusion-detection module 11 data-port 2:

Switchport: Enabled
Administrative Mode: trunk
Operational Mode: trunk
Administrative Trunking Encapsulation: dot1q
Operational Trunking Encapsulation: dot1q
Negotiation of Trunking: Off
Access Mode VLAN: 662 (ward-victim3)
Trunking Native Mode VLAN: 1 (default)
Trunking VLANs Enabled: NONE
Pruning VLANs Enabled: 2-1001
Vlans allowed on trunk:none
Vlans allowed and active in management domain: none
Vlans in spanning tree forwarding state and not pruned:
   none
Administrative Capture Mode: Disabled
Administrative Capture Allowed-vlans: empty
```

Administrative Tasks for the IDSM2

This section contains procedures that help you with administrative tasks for the IDSM2. It contains the following topics:

- Enabling Full Memory Tests, page 19-40
- Resetting the IDSM2, page 19-41

Enabling Full Memory Tests

When the IDSM2 initially boots, by default it runs a partial memory test. You can enable a full memory test in Catalyst software and Cisco IOS software. This section describes how to enable full memory tests, and contains the following topics:

- Catalyst Software, page 19-40
- Cisco IOS Software, page 19-41

Catalyst Software

Use the **set boot device** *boot_sequence module_number* **mem-test-full** command to enable a full memory test. The full memory test takes about 12 minutes.

To enable a full memory test, follow these steps:

Step 1 Log in to the console.

Step 2 Enter privileged mode.

```
console> enable
```

Step 3 Enable the full memory test.

```
console> (enable) set boot dev cf:1 3 mem-test-full
Device BOOT variable = cf:1
Memory-test set to FULL
Warning: Device list is not verified but still set in the boot string.
console> (enable) set boot dev hdd:1 3 mem-test-full
Device BOOT variable = hdd:1
Memory-test set to FULL
Warning: Device list is not verified but still set in the boot string.
console> (enable)
```

The **set boot device** command can either contain **cf:1** or **hdd:1**.

Step 4 Reset the IDSM2.

The full memory test runs.

Note A full memory test takes more time to complete than a partial memory test.

For More Information

For the procedure for resetting the IDSM2, see Resetting the IDSM2, page 19-41.

Cisco IOS Software

Use the **hw-module module** *module_number* **reset mem-test-full** command to enable a full memory test. The full memory test takes about 12 minutes. To enable a full memory test, follow these steps:

Step 1 Log in to the console.

Step 2 Enable the full memory test.

```
router# hw-module module 9 reset mem-test-full
    Device BOOT variable for reset = <empty>
    Warning: Device list is not verified.

    Proceed with reload of module?[confirm]
    % reset issued for module 9
router#
```

Step 3 Reset the IDSM2.

The full memory test runs.

Note A full memory test takes more time to complete than a partial memory test.

For More Information

For the procedure for resetting the IDSM2, see Resetting the IDSM2, page 19-41.

Resetting the IDSM2

If for some reason you cannot communicate with the IDSM2 through SSH, Telnet, or the switch **session** command, you must reset the IDSM2 from the switch console. The reset process requires several minutes. This section describes how to reset the IDSM2, and contains the following topics:

Catalyst Software

To reset the IDSM2 from the CLI, follow these steps:

Step 1 Log in to the console.

Step 2 Enter privileged mode.

```
console> enable
```

Step 3 Reset the IDSM2 to the application partition or the maintenance partition.

```
console> (enable) reset module_number {hdd:1 | cf:1}
```

Note If you do not specify either the application partition (hdd:1 the default) or the maintenance partition (cf:1), the IDSM2 uses the boot device variable.

Example

```
console> (enable) reset 3
2003 Feb 01 00:18:23 %SYS-5-MOD_RESET: Module 3 reset from console//
Resetting module 3... This may take several minutes.
2003 Feb 01 00:20:03 %SYS-5-MOD_OK: Module 3 is online.
console> (enable)
```

Caution If the IDSM2 is removed from the switch chassis without first being shut down, or the chassis loses power, you may need to reset the IDSM2 more than once. If the IDSM2 fails to respond after three reset attempts, boot the maintenance partition, and perform the instructions for restoring the application partition.

For More Information

For the procedure for reimaging the IDSM2, see Installing the IDSM2 System Image, page 22-26.

Cisco IOS Software

Use the **hw-module module** *slot_number* **reset** {**hdd:1** | **cf:1**} command in EXEC mode to reset the IDSM2. The reset process takes several minutes. The IDSM2 boots into the boot partition you specify. If you do not specify the boot string, the default boot string is used.

To reset the IDSM2 from the CLI, follow these steps:

Step 1 Log in to the console.

Step 2 Reset the IDSM2.

```
router# hw-module module module-number reset {hdd:1 | cf:1}
```

Note If you do not specify either the application partition (**hdd:1** the default) or the maintenance partition (**cf:1**), the IDSM2 uses the boot device variable.

Example:

```
router# hw-module module 8 reset
Device BOOT variable for reset =
Warning: Device list is not verified.
Proceed with reload of module? [confirm]
% reset issued for module 8
router#
```

Catalyst and Cisco IOS Software Commands

Note For more detailed information on Catalyst and Cisco IOS software commands, refer to the command references found on Cisco.com. For instructions on how to locate these document, refer to *Documentation Roadmap for Cisco Intrusion Prevention System 6.1*.

This section lists the Catalyst and Cisco IOS software commands that pertain to the IDSM2, and contains the following topics:

Catalyst Software

This section lists supported and unsupported Catalyst Software Commands. It contains the following topics:

Supported Supervisor Engine Commands

The IDSM2 also supports the following supervisor engine CLI commands, which are described in more detail in the Catalyst 6500 Series Command References.

- **clear config** *module_number*

  Clears the configuration on the supervisor engine that is associated with the specified IDSM2.
- **clear log** *module_number*

  Deletes all entries in the error log for the specified IDSM2.
- **session** *slot_number*

  Logs in to the console of the IDSM2 from the switch console.
- **set module** commands (all other **set module** commands return an error message):
  - **set module name** *module_number*

    Sets the name of the module.
  - **set module power** *module_number* [**up** | **down**]

    Enables or disables power to the specified IDSM2.
- **set port name** *module_number*

  Configures the name for the specified IDSM2 port.
- **set span**

  Configures port 1 as a SPAN destination port. You cannot use port 1 on the IDSM2 as a SPAN source port.
- **set trunk**

  Configures trunk ports.

- **set vlan**

  Configures VLAN capture ports.
- **show config**

  Displays the supervisor engine NVRAM configurations.
- **show log**

  Displays the error logs for the specified IDSM2.
- **show mac** *module_number*

  Displays the MAC counters for the specified IDSM2.
- **show module** *module_number*

  With an IDSM2 installed, displays `Intrusion Detection System Module` under `Module-Type`.
- **show port** *module_number*

  Displays the port status for the specified IDSM2.
- **show port capabilities** [*module* | *module_number*]

  Displays the capabilities of the module and ports.
- **show test**

  Displays the errors reported from the diagnostic tests for both the SPAN port (port 1) and the management port (port 2) and the BIOS and CMOS boot results.

Unsupported Supervisor Engine Commands

The following supervisor engine CLI commands are not supported by the IDSM2:

- **set module** [**enable** | **disable**] *module_number*
- **set port broadcast**
- **set port channel**
- **set port cops**
- **set port disable**
- **set port enable**
- **set port flowcontrol**
- **set port gmrp**
- **set port gvrp**
- **set port host**
- **set port inlinepower**
- **set port jumbo**
- **set port membership**
- **set port negotiation**
- **set port protocol**
- **set port qos**
- **set port rsvp**
- **set port security**

- **set port speed**
- **set port trap**
- **set protocolfilter**
- **set rgmp**
- **set snmp**
- **set spantree**
- **set udld**
- **set vtp**

Cisco IOS Software

This section lists the Cisco IOS software commands that the IDSM2 supports. These commands are grouped according to mode. This section contains the following topics:

EXEC Commands

The following commands are all performed in EXEC mode:

- **clock read-calendar**

  Updates the clock time to the calendar time.
- **clock set** *time date*

  Sets the current time and date.
- **clock update-calendar**

  Updates the calendar time to the clock time.
- **hw-module module** *module_number* **reset** {**cf:1** | **hdd:1**}

  Resets the IDSM2 into the partition specified by the boot device variable; if the boot device variable has not been set, the IDSM2 is reset to the application partition by default. Use the command **show boot device module** *module_number* to view the current setting of the boot device variable. **cf:1** is the maintenance partition. **hdd:1** is the application partition.
- **hw-module module** *module_number* **shutdown**

  Shuts down the IDSM2 so that it can be safely removed from the chassis.
- **reload**

  Reloads the entire switch.
- **session slot** *module_number* **processor** *processor_number*

  Logs in to the console of the IDSM2 from the switch console.
- **show boot device module** *module_number*

  Displays the current boot string for the specified module.

- **show diagnostic result module** *module_number*

  Displays the results of the online diagnostics that were performed when the IDSM2 was last booted up.
- **show interface port-channel** *channel_number*

  Displays the status of the port channel.
- **show intrusion-detection module** *module_number* **data-port {1 | 2} {state | traffic}**

  Displays the state or traffic statistics of the specified IDSM2 data port.
- **show intrusion-detection module** *module_number* **management-port {state | traffic}**

  Displays the state or traffic statistics of the IDSM2 management port.
- **show ip access-lists**

  Displays the current access lists.
- **show module [***module_number* **| all | version]**

  Displays the installed modules, versions, and states.
- **show monitor session** *session_number*

  Displays the SPAN source and destination for the specified session.
- **show running-config**

  Displays the configuration that is currently running.
- **show spanning-tree active**

  Displays spanning tree state information for active interfaces only.
- **show spanning-tree detail**

  Displays detailed spanning tree state information.
- **show spanning-tree summary [totals]**

  Displays the high level state of spanning tree. Does not show interface specific information.
- **show spanning-tree vlan** *vlan_number*

  Displays spanning tree state information for the specified VLAN. Includes list of ports on which those VLANs are forwarded or blocked.
- **show startup-config**

  Displays the saved configuration.
- **show vlan access-map**

  Displays all current VLAN access maps.

Configuration Commands

The following configuration commands are all performed in either global configuration mode, interface configuration mode, or VACL configuration submode:

- Global configuration mode
  - **boot device module** *number_number* {**cf:1** | **hdd:1**}

    Sets the default boot device for the specified module. **cf:1** boots to the MP and **hdd:1** boots to the AP. The **no** option clears the boot string, which sets the default boot device to the AP.
  - **clock calendar valid**

    Sets the current calendar time as the switch time on bootup.
  - **clock summer-time** *zone* **recurring**

    Sets the switch to use the summertime settings.
  - **clock timezone** *zone offset*

    Sets the timezone for the switch/IDSM2.
  - **fabric switching-mode force busmode**

    Lets service modules that do not support packet recirculation, be forced into communicating through the chassis shared bus instead of the switched fabric. This forces the supervisor to handle the packet recirculation centrally and lets the service module communicate properly on VLANs meeting the conditions stated above. Other fabric enabled modules that are not affected by this problem continue to communicate through the switch fabric even if this command is enabled.
  - **[no] intrusion-detection module** *module_number* **data-port {1 | 2} access vlan** *vlan_id*

    Sets the data port to access (inline) mode and sets the access VLAN for the data port for the specified module.
  - **[no] intrusion-detection module** *module_number* **data-port {1 | 2} autostate include**

    Includes (or excludes) the specified data port in the autostate calculation. When included, the switch virtual interface associated with an MSFC or WLAN port remains up while the module's data port is enabled. When excluded, the switch virtual interface associated with the MSFC or WAN port goes down if the specified module's data port is the only active port in the VLAN. The default is **no include**.
  - **[no] intrusion-detection module** *module_number* **data-port {1 | 2} capture**

    Configures the specified data port as a capture destination port. You must also set the allowed VLAN list through the **intrusion-detection module** *module_number* **data-port {1 | 2} capture** command before any packets are captured. The IDSM2 must be in promiscuous mode.
  - **[no] intrusion-detection module** *module_number* **data-port {1 | 2} capture allowed-vlan** *vlan_list*

    Sets the allowed VLANs on the specified data port for packet capture. You must also enable capture mode on the data port through the **intrusion-detection module** *module_number* **data-port {1 | 2} capture** command before traffic is captured on the data port.
  - **intrusion-detection module** *module_number* **data-port {1 | 2} default**

    Restores the allowed VLANs, autostate, PortFast, port cost, and priority settings for the specified data port to the default values. This command is useful to remove any configuration from a data port before you add it to a channel group.

– **[no] intrusion-detection module** *module_number* **data-port {1 | 2}port-channel** *channel_number*

  Adds the data port for the specified module to the channel group, which creates a port channel with the same numeric ID. If the channel group and port channel have not been created, this command creates it with an empty allowed VLAN list. The **no** option removes the data port from the channel group, restores the data port settings to their defaults, and deletes the port channel if it is empty.

– **[no] intrusion-detection module** *module_number* **data-port {1 | 2}portfast {enable | disable} [trunk]**

  Enables or disables PortFast on the data port. When PortFast is enabled, traffic is forwarded by the switch to the IDSM2 data port while the spanning tree is being built. When disabled, traffic is inhibited until after the tree is built and the backplane port is in the forwarding state. The default is disabled. The trunk option enables or disables PortFast when the data port is configured as a trunk (in promiscuous or inline VLAN pairs mode).

– **[no] intrusion-detection module** *module_number* **data-port {1 | 2} spanning-tree cost** *path_cost*

  Sets the spanning tree path cost for the data port on the specified module. The **no** option restores the spanning tree cost for the data port on the specified module to the default cost value.

– **[no] intrusion-detection module** *module_number* **data-port {1 | 2} trunk allowed-vlan** *vlan_list*

  Sets the data port to trunking mode and sets the list of allowed VLANs on the data port for the specified module. The **no** option removes the data port from trunking mode and clears the list of allowed VLANs on the data port for the specified module.

– **intrusion-detection module** *module_number* **management-port access-vlan** *vlan_number*

  Sets the access VLAN for the IDSM2 command and control port.

– **intrusion-detection module** *module_number* **data-port** *data_port_number* **capture allowed-vlan** *allowed_capture_vlan(s)*

  Configures the VLAN(s) for VACL capture.

– **intrusion-detection module** *module_number* **data-port** *data_port_number* **capture**

  Enables VACL capture for the specified IDSM2 data port.

– [**no**] **intrusion-detection port-channel** *channel_number* **access vlan** *vlan_id*

  Sets all data ports in the specified port channel to access mode and sets the access VLAN for the data ports. The **no** option clears the list of allowed VLANs on the data ports of all modules in the specified port channel.

– [**no**] **intrusion-detection port-channel** *channel_number* **autostate include**

  Includes or excludes all data ports in the specified port channel from the autostate calculation. When included, the virtual switch interface associated with an MSFC or WLAN port remains up while the data port is enabled. When excluded, the virtual switch interface associated with the MSFC or WAN port goes down if the data port is the only active port in the VLAN. The data ports are excluded from the autostate calculations by default.

– **[no] intrusion-detection port-channel** *channel_number* **capture**

  Configures all data ports in the channel group as capture ports. The **no** option disables the capture function on all data ports in the channel group.

– [**no**] **intrusion-detection port-channel** *channel_number* **capture allowed-vlan** *vlan_id*

  Sets the list of capture VLANs on the data ports of all modules in the specified port channel. This command does not set the channel group to capture mode. Use the **intrusion-detection port-channel** *channel_number* **capture** command to set the channel group to capture mode. The **no** option clears the list of capture VLANs on the data ports of all modules in the specified port channel.

– **[no] intrusion-detection port-channel** *channel_number* **portfast {enable | disable} [trunk]**

  Enables or disables PortFast on the data ports in the port channel. When PortFast is enabled, traffic is forwarded by the switch to the data port while the spanning tree is being built. When disabled, traffic is inhibited until after the tree is built and the backplane port is in the forwarding state. Use the **trunk** option to enable or disable PortFast when the data port is configured as a trunk (in promiscuous or inline VLAN pair mode). Do not use the **trunk** option when the data ports are configured as access ports (inline mode). PortFast and PortFast trunk are disabled by default.

– [**no**] **intrusion-detection port-channel** *channel_number* **spanning-tree cost** *port_cost*

  Sets the spanning tree port cost for the data port on the specified module. The **no** option restores the spanning tree port cost for the data port on the specified module to the default value.

– [**no**] **intrusion-detection port-channel** *channel_number* **spanning-tree priority** *priority*

  Sets the spanning tree port priority for the data port on the specified module. The **no** option restores the spanning tree port priority for the data port on the specified module to the default value.

– [**no**] **intrusion-detection port-channel** *channel_number* **trunk allowed-vlan** *vlan_id*

  Sets the list of allowed VLANs on the data ports of all modules in the specified port channel. The **no** option clears the list of allowed VLANs on the data ports of all modules in the specified port channel.

– **ip access-list extended** *word*

  Creates access lists for use in the VACL maps.

– **[no] monitor session** *session_number* **destination intrusion-detection module** *module_number* **data-port {1 | 2}**

  Configures a SPAN destination port, which can be either a standard line card port or an IDSM2 data port.

– **[no] monitor session** *session_number* {**source** {**interface** *interface_number*} | {**vlan** *vlan_id*}} [, | - | **rx** | **tx** | **both**]

  Sets the sources for a SPAN session.

– **[no] power enable module** *module_number*

  Powers the IDSM2 off or on.

– **[no] spanning tree mode [pvst | mst | rapid-pvst]**

  Selects the spanning tree protocol (PVST+, MST, or Rapid-PVST+) to be used globally on the switch. The default is PVST. MST is not supported for the IDSM2. The **no** option restores the spanning tree mode to the default.

  - **vlan access-map** *map_name_sequence*

    Creates the VACL maps.
  - **vlan filter** *map_name* **vlan-list** *vlans*

    Maps the VACL maps to VLANs.
- Interface configuration mode
  - **switchport**

    Sets the interface as a switch port.
  - **switchport access vlan** *vlan*

    Sets the access VLAN for the interface.
  - **switchport capture**

    Sets the interface as a capture port.
  - **switchport mode access**

    Sets the interface as an access port.
  - **switchport mode trunk**

    Sets the interface as a trunk port.
  - **switchport trunk allowed vlan** *vlans*

    Sets the allowed VLANs for trunk.
  - **switchport trunk encapsulation dot1q**

    Sets dot1q as the encapsulation type.
  - **switchport trunk native vlan** *vlan*

    Sets the native VLAN for the trunk port.
- VACL configuration submode
  - **action forward capture**

    Designates that matched packets should be captured.
  - **match ip address** [*1-199* | *1300-2699* | *acl_name*]

    Specifies filtering in the VACL.

CHAPTER 20

Configuring the NME IPS

Note All IPS platforms allow ten concurrent log in sessions.

This chapter describes how to configure the NME IPS and get it ready to receive IPS traffic. After that you are ready to configure intrusion prevention. This chapter contains the following sections:

NME IPS Configuration Sequence

Perform the following tasks to configure the NME IPS:

1. Set up the interfaces.
2. Log in to the NME IPS.
3. Initialize the NME IPS.

   Run the **setup** command to initialize the NME IPS.
4. Configure the NME IPS to capture traffic for intrusion prevention.
5. Create the service account.

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 new password if the administrator password is lost. Analyze your situation to decide if you want a service account existing on the system.

6. Perform the other initial tasks, such as adding users, trusted hosts, and so forth.
7. Configure intrusion prevention.
8. Perform administrative tasks to keep your NME IPS running smoothly.
9. Upgrade the IPS software with new signature updates and service packs.
10. Reimage the boot helper and bootloader when needed.

For More Information

- For the procedure for setting up interfaces, see Setting Up Interfaces on the NME IPS and the Router, page 20-4.
- For the procedure for logging in to the NME IPS, see Logging In to the NME IPS, page 2-7.
- For the procedure for running the **setup** command on the NME IPS, see Advanced Setup for the NME IPS, page 3-24.
- For the procedure for creating the service account, see Creating the Service Account, page 4-14.
- For the procedures for setting up the sensor, see Chapter 4, "Setting Up the Sensor."
- For the procedures for configuring intrusion prevention, see Chapter 9, "Configuring Anomaly Detection," Chapter 7, "Configuring Event Action Rules," Chapter 8, "Defining Signatures," and Chapter 13, "Configuring Attack Response Controller for Blocking and Rate Limiting."
- For the procedures to keep your sensor running smoothly, see Chapter 16, "Administrative Tasks for the Sensor."
- For more information on obtaining Cisco IPS software, see Obtaining Cisco IPS Software, page 21-1.
- For the procedure for reimaging the NME IPS, see Installing the NME IPS System Image, page 22-38.

Verifying Installation and Finding the Serial Number

Use the **show inventory** command in privileged EXEC mode to verify the installation of the NME IPS.

Note You can also use this command to find the serial number of your NME IPS for use in troubleshooting with TAC. The serial number appears in the PID line, for example, SN: FHH1117001R.

To verify the installation of the NME IPS, follow these steps:

Step 1 Log in to the router.

Step 2 Enter privileged EXEC mode on the router.

```
router> enable
```

Step 3 Verify that the NME IPS is part of the router inventory.

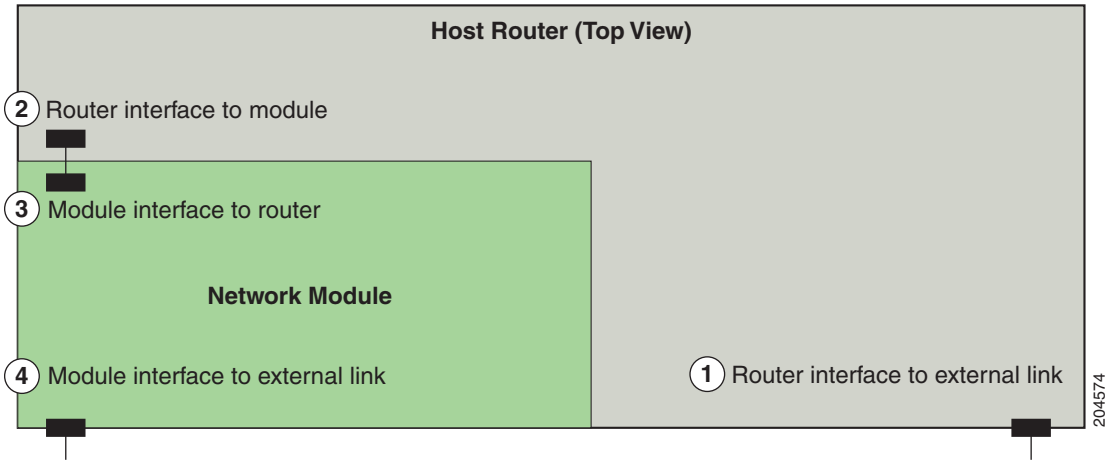
```
router# show inventory
NAME: "3845 chassis", DESCR: "3845 chassis"
PID: CISCO3845         , VID: V01 , SN: FTX1002C255

NAME: "c3845 Motherboard with Gigabit Ethernet on Slot 0", DESCR: "c3845 Motherb
oard with Gigabit Ethernet"
PID: CISCO3845-MB      , VID: V03 , SN: FOC09514J4Y
```

```
NAME: "4 Port FE Switch on Slot 0 SubSlot 0", DESCR: "4 Port FE Switch"
PID: HWIC-4ESW          , VID: V01 , SN: FOC1102394U

NAME: "High Speed WAN Interface Card - 1 Port Gigabit Ethernet on Slot 0 SubSlot
 3", DESCR: "High Speed WAN Interface Card - 1 Port Gigabit Ethernet"
PID: HWIC-1GE-SFP       , VID: V01 , SN: FOC10164DAR

NAME: "1000BASE-T SFP", DESCR: "1000BASE-T SFP"
PID: SP7041             , VID: C  , SN: 00000MTC101608RB

NAME: "Cisco Intrusion Prevention System NM on Slot 2", DESCR: "Cisco Intrusion
Prevention System NM"
PID: NME-IPS-K9         , VID: V01, SN: FHH1117001R


router#
```

Understanding the Hardware Interfaces

Figure 20-1 shows the router and the NME IPS interfaces used for internal and external communication. You can configure the router interfaces through the Cisco IOS CLI and the NME IPS interfaces through the IPS CLI, IDM, IME, or CSM.

Figure 20-1 NME IPS and Router Interfaces



1	Router interface to external link Configure the standard router settings using the Cisco IOS CLI.
2	Router interface to the NME IPS (ids-sensor x/0) Configure the IP address and default gateway router of the NME IPS using the Cisco IOS CLI.
3	The NME IPS interface to router (GigabitEthernet0/1) Configure the interface as inline or promiscuous using the Cisco IOS CLI.
4	The NME IPS interface to external link (Management0/1) Configure the command and control interface using the IPS CLI, IDM, IME, or CSM.

Setting Up Interfaces on the NME IPS and the Router

This section describes how to set up interfaces on the NME IPS and the router, contains the following topics:

Interface Configuration Sequence

Follow this sequence to set up interfaces on the NME IPS and the router:

1. Configure the IPS command and control interface on the router, and the NME IPS IP address, mask, and gateway.
2. Enable the monitoring interface and specify whether it is promiscuous or inline, assign the ACL to the interface, specify how you want the router to handle traffic if the module fails, and create a monitoring ACL (optional).
3. Save the configuration.

For More Information

For the procedure for enabling the monitoring interface, see Setting Up Interfaces on the NME IPS and the Router, page 20-4.

ARC and NAT

If you use NAT to establish management access to the NME IPS, ARC on the NME IPS does not know the external IP address of the NME IPS. To make sure that management access to the NME IPS is not interrupted by devices that the NME IPS is managing, you must state the NAT address of the NME IPS every time you add a blocking device.

For More Information

- For more information on ARC, see Chapter 13, “Configuring Attack Response Controller for Blocking and Rate Limiting.”
- For the procedures for configuring the NME IPS NAT address every time you add a blocking device, see the following procedures:
  - Configuring the Sensor to Manage Cisco Routers, page 13-22
  - Configuring the Sensor to Manage Catalyst 6500 Series Switches and Cisco 7600 Series Routers, page 13-25
  - Configuring the Sensor to Manage Cisco Firewalls, page 13-27

Configuring the IDS-Sensor Interfaces on the Router

To configure the NME IPS interfaces, follow these steps:

Step 1 Log in to the router.

Step 2 Enter privileged EXEC mode on the router.

```
router> enable
```

Step 3 Confirm the module slot number in your router.

```
router# show run | include ids-sensor
interface IDS-Sensor1/0
router#
```

Step 4 Enable the CEF switching path.

```
router> configuration terminal
router(config)# ip cef
router(config)#
```

Step 5 Create a loopback interface.

```
router(config)# interface loopback 0
router(config-if)#
```

Step 6 Assign an IP address and netmask to the loopback interface.

```
router(config-if)# ip address 10.99.99.99 255.255.255.255
router(config-if)# exit
router(config)#
```

Note You must assign an IP address to the internal interface of the NME IPS to session in to the NME IPS. Choose a network that does not overlap with any networks assigned to the other interfaces in the router.

Step 7 Assign an unnumbered loopback interface to the IDS-Sensor interface. Use slot 1 for this example.

```
router(config)# interface ids-sensor 1/0
router(config-if)# ip unnumbered Loopback 0
router(config-if)#
```

Step 8 Activate the port.

```
router(config-if)# no shutdown
router(config-if)#
```

Step 9 Exit configuration mode.

```
router(config-if)# end
```

Step 10 Write the configuration to NVRAM.

```
router# write memory
Building configuration
[OK]
```

You are now ready to initialize the NME IPS and configure intrusion prevention.

For More Information

- For more information on sessioning from the router to the NME IPS and exiting sessions, see Establishing Sessions, page 20-7.
- For the procedure for using the **setup** command to initialize the NME IPS, see Advanced Setup for the NME IPS, page 3-24.

Configuring Monitoring on the Router Interface

To configure the router interface to be monitored, follow these steps:

Step 1 Log in to the router.

Step 2 Enter privileged EXEC mode on the router.

```
router> enable
```

Step 3 (Optional) Configure a monitoring access list on the router.

```
router(config)# access-list 101 permit tcp any eq www any
```

You can set up a standard access list and apply it to filter what type of traffic you want to inspect. A matched ACL causes traffic not to be inspected for that ACL. This example bypasses inspection of HTTP traffic only. Refer to your Cisco IOS Command Reference for more information on the options for the **access-list** command.

Step 4 Enable monitoring on the interface in either inline or promiscuous mode and associate the access list.

```
router(config)# interface monitored_interface
router(config-if)# ids-service-module monitoring [inline | promiscuous] access-list 101
router(config-if)# exit
router(config)#
```

Note Associating the access list with the interface further controls what traffic is sent to the NME IPS.

Step 5 (For inline mode) Confirm the module slot number in your router.

```
router# show run | include ids-sensor
interface IDS-Sensor1/0
router#
```

Step 6 (For inline mode) Specify how the router handles traffic inspection during a module failure.

```
router(config)# interface ids-sensor 1/0
router(config-if)# service-module [fail-close | fail-open]
router(config-if)#
```

The default is fail-open.

Note The **fail-close** option means that if the NME IPS fails, then the router does not let traffic pass. The **fail-open** option means if the NME IPS fails, the router lets traffic pass, but it is not inspected by the IPS.

Step 7 Exit configuration mode.

```
router(config-if)# exit
router(config)# exit
router#
```

Step 8 Write the configuration to NVRAM.

```
router# write memory
Building configuration
[OK]
```

For More Information

- For more information on promiscuous mode, see Configuring Promiscuous Mode, page 5-15.
- For more information on inline mode, see Configuring Inline Interface Mode, page 5-16.

Establishing Sessions

Because the NME IPS does not have an external console port, console access to the NME IPS is enabled when you issue the **service-module ids-sensor** *slot/port* **session** command on the router, or when you initiate a Telnet connection into the router with the slot number corresponding to the NME IPS port number. The lack of an external console port means that the initial bootup configuration is possible only through the router.

When you issue the **service-module ids-sensor** *slot/port* **session** command, you create a console session with the NME IPS, in which you can issue any IPS configuration commands. After completing work in the session and exiting the IPS CLI, you are returned to the Cisco IOS CLI.

The **session** command starts a reverse Telnet connection using the IP address of the IDS-Sensor interface. The IDS-Sensor interface is an interface between the NME IPS and the router. You must assign an IP address to the IDS-Sensor interface before invoking the **session** command. Assigning a routable IP address can make the IDS-Sensor interface itself vulnerable to attacks, because the NME IPS is visible on the network through that routable IP address, meaning you can communicate with the NME IPS outside the router. To counter this vulnerability, assign an unnumbered IP address to the IDS-Sensor interface. Then the NME IPS IP address is only used locally between the router and the NME IPS, and is isolated for the purposes of sessioning in to the NME IPS.

Note Before you install your application software or reimage the module, opening a session brings up the bootloader. After you install the software, opening a session brings up the application.

Caution If you session to the module and perform large console transfers, character traffic may be lost unless the host console interface speed is set to 115200/bps or higher. Use the **show running config** command to check that the speed is set to 115200/bps.

For More Information

For the procedure for setting up monitoring, see Configuring the IDS-Sensor Interfaces on the Router, page 20-5.

Opening and Closing a Session

Note You must initialize the NME IPS (run the **setup** command) from the router. After networking is configured, SSH and Telnet are available.

Use the **service-module ids-sensor** *slot/port* **session** command to establish a session from the NME IPS to the module. Press **Ctrl-Shift-6**, then **x**, to return a session prompt to a router prompt, that is, to go from the NME IPS prompt back to the router prompt. Press **Enter** on a blank line to go back to the session prompt, which is also the router prompt. You should only suspend a session to the router if you will be returning to the session after executing router commands. If you do not plan on returning to the NME IPS session, you should close the session rather than suspend it.

When you close a session, you are logged completely out of the NME IPS CLI and a new session connection requires a username and password to log in. A suspended session leaves you logged in to the CLI. When you connect with the **session** command, you can go back to the same CLI without having to provide your username and password.

Note Telnet clients vary. In some cases, you may have to press **Ctrl-6 + x**. The control character is specified as ^^, Ctrl-^, or ASCII value 30 (hex 1E).

Caution If you use the **disconnect** command to leave the session, the session remains running. The open session can be exploited by someone wanting to take advantage of a connection that is still in place.

To open and close sessions to the NME IPS, follow these steps:

Step 1 Log in to the router.

Step 2 Check the status of the NME IPS to make sure it is running.

```
router# service-module ids-sensor 1/0 status
Service Module is Cisco IDS-Sensor1/0
Service Module supports session via TTY line 130
Service Module is in Steady state
Service Module heartbeat-reset is disabled
Getting status from the Service Module, please wait..

Cisco Systems Intrusion Prevention System Network Module
  Software version:  6.1(1)E2
  Model:             NME-IPS
  Memory:            443508 KB
  Mgmt IP addr:      10.89.148.195
  Mgmt web ports:    443
  Mgmt TLS enabled:  true


router#
```

Step 3 Open a session from the router to the NME IPS.

```
router# service-module ids-sensor 1/0 session
Trying 10.89.148.195, 2322 ... Open
```

Step 4 Exit, or suspend and close the module session:

- `sensor# exit`

Note If you are in submodes of the IPS CLI, you must exit all submodes. Enter **exit** until the sensor login prompt appears.

Caution Failing to close a session properly makes it possible for others to exploit a connection that is still in place. Remember to enter **exit** at the `router#` prompt to close the Cisco IOS session completely.

- To suspend and close the session to the NME IPS, press **Ctrl-Shift** and press **6**. Release all keys, and then press **x**.

Note When you are finished with a session, you need to return to the router to establish the association between a session (the IPS application) and the router interfaces you want to monitor.

Step 5 Disconnect from the router.

```
router# disconnect
```

Step 6 Press **Enter** to confirm the disconnection.

```
router# Closing connection to 10.89.148.196 [confirm] <Enter>
```

For More Information

For the procedure for initializing the NME IPS, see Advanced Setup for the NME IPS, page 3-24.

Displaying the Status of the NME IPS

Use the **service-module ids-sensor** *slot/port* **status** command in privileged EXEC mode to display the status and statistics of the NME IPS.

To display the status of the NME IPS, follow these steps:

Step 1 Log in to the router.

Step 2 Enter privileged EXEC mode on the router.

```
router> enable
```

Step 3 Display the status of the NME IPS.

```
router# service-module ids-sensor 1/0 status
Service Module is Cisco IDS-Sensor1/0
Service Module supports session via TTY line 130
Service Module is in Steady state
Service Module heartbeat-reset is disabled
Getting status from the Service Module, please wait..
```

```
Cisco Systems Intrusion Prevention System Network Module
  Software version:  6.1(1)E2
  Model:             NME-IPS
  Memory:            443508 KB
  Mgmt IP addr:      10.89.148.195
  Mgmt web ports:    443
  Mgmt TLS enabled:  true


router#
```

Enabling and Disabling Heartbeat Reset

Use the **service-module ids-sensor** *slot/port* **heartbeat reset [enable | disable]** command in privileged EXEC mode to reset the heartbeat of the NME IPS.

When the NME IPS is booted in failsafe mode or is undergoing an upgrade, you can use the **service-module ids heartbeat-reset** command to prevent a reboot during the process. If you leave the heartbeat reset enabled during an upgrade, you may lose the NME IPS heartbeat.

When the NME IPS heartbeat is lost, the router applies a fail-open or fail-close configuration option to the NME IPS and stops sending traffic to the NME IPS, and sets the NME IPS to error state. The router performs a hardware reset on the NME IPS and monitors the NME IPS until the heartbeat is reestablished.

Note Disabling the heartbeat reset prevents the router from resetting the module during system image installation if the process takes too long.

To reset the heartbeat of the NME IPS, follow these steps:

Step 1 Log in to the router.

Step 2 Enter privileged EXEC mode on the router.

```
router> enable
```

Step 3 Verify the status of heartbeat reset.

```
router# service-module ids-sensor 1/0 status
Service Module is Cisco IDS-Sensor 1/0
Service Module supports session via TTY line 194
Service Module heartbeat-reset is enabled
```

Step 4 To disable the heartbeat on the NME IPS:

```
router# service-module ids-sensor 1/0 heartbeat-reset disable
```

Step 5 To reenable the heartbeat on the NME IPS:

```
router# service-module ids-sensor 1/0 heartbeat-reset enable
```

Rebooting, Resetting, and Shutting Down the NME IPS

This section describes when and how the NME IPS shuts down. It contains the following topics:

NME IPS Status Monitoring

The NME IPS uses RBCP to monitor its status. RBCP is monitored by the main application on the NME IPS, not by SensorApp. If the main application on the NME IPS fails, the RBCP heartbeat responses do not return from the NME IPS. When the router determines that the NME IPS has failed, a **reload** command is issued through RBCP to reboot the Linux kernel on the NME IPS. In the period during the attempt to bring the NME IPS back up, the router works in the mode determined by the failover operation configured.

In some cases, SensorApp may stop processing, but the main application on the NME IPS continues to process RBCP packets. In this case, packets are processed according to the bypass settings set for the NME IPS by the IPS CLI, IDM, or IME.

There are two situations in which the NME IPS shuts down:

- A hardware or software error forces it to fail. The router can detect this through the loss of the RBCP heartbeat.
- **Reload** or **shutdown** command.

For More Information

- For more information on SensorApp, see SensorApp, page A-22.
- For more information on software bypass, see Configuring Inline Bypass Mode, page 5-33.

Rebooting, Resetting, and Shutting Down the NME IPS

Use the **service-module ids-sensor** *slot/port* [**reload | reset | shutdown**] command in privileged EXEC mode to reboot, reset, and shut down the NME IPS.

To reboot, reset, and shut down the NME IPS, follow these steps:

Step 1 Log in to the router.

Step 2 Enter privileged EXEC mode on the router.

```
router> enable
```

Step 3 To gracefully halt and reboot the operating system on the NME IPS.

```
router# service-module ids-sensor 1/0 reload
Do you want to proceed with the reload? [confirm]
```

Step 4 To reset the hardware on the NME IPS.

```
router# service-module ids-sensor 1/0 reset
Use reset only to recover from shutdown or failed state
Warning: May lose data on the hard disc!
Do you want to reset?[confirm]
```

Note The NME IPS has a compact flash device that functions as a permanent storage device rather than a hard-disk drive.

Caution Data loss occurs only if you issue the **reset** command without first shutting down the NME IPS. You can use the **reset** command safely in other situations.

Step 5 To shut down applications running on the NME IPS.

```
router# service-module ids-sensor 1/0 shutdown
Trying 10.10.10.1, 2129 . . . Open
%SERVICEMODULE-5-SHUTDOWN2:Service module IDS-Sensor1/0 shutdown complete
```

New and Modified Commands

This section describes the following new and modified Cisco IOS commands, and specific commands that are used to configure the NME IPS. All other Cisco IOS software commands are documented in the Cisco IOS Release 12.4(20)T command reference at Cisco.com, http://www.cisco.com/en/US/products/ps6441/index.html. This section contains the following topics:

interface ids-sensor

To configure the IPS sensor interface and enter config-if mode, use the **interface ids-sensor** command in config mode. To specify how the router handles traffic inspection during a module failure, use the **service-module** command in config-if mode. The default is fail open.

interface ids-sensor *slot/port*

ip {**address** | **unnumbered**}

service-module {***fail-close*** | ***fail-open***}

Syntax Description

slot	Number of the router chassis slot for the NME IPS.
/port	Port number of the NME IPS. **Note** The slash mark is required between the *slot* argument and the *unit* argument.
ids-sensor	The IPS interface for the sensor.
ip address	Sets the IP address of an interface.

ip unnumbered	Enables IP address processing without an explicit IP address.
service-module fail-close	The NME IPS drops all the traffic.
service-module fail-open	The NME IPS passes all the traffic through, but does not perform traffic inspection (default).

Caution Although there are 57 subcommands associated with the **ip** command, the only two supported for the modules are **ip address** and ip **unnumbered**. Enabling any of the other subcommands can result in unpredictable behavior.

Command Defaults

Command Modes

Config

Config-if

Command History

Release	Modification
12.4(20)T	This command was introduced.

Usage Guidelines

The **interface ids-sensor** *slot/port* command lets you enter config-if mode and configure the IPS sensor slot and port. On the NME IPS, the slot value is specified by identifying the physical location where the module is installed on the router and the port number is 0.

Examples

The following example demonstrates how to use the **interface IDS-Sensor** command to enter config-if mode on an NME IPS in slot 1, port 0:

```
router(config)# interface ids-sensor 1/0
router(config-if)#
```

The following example demonstrates how to use the **interface ids-sensor** command with the **ip unnumbered** subcommand to specify the router command and control interface:

```
router(config)# interface ids-sensor 1/0
router(config-if)# ip unnumbered router_command_and_control_interface
router(config-if)#
```

The following example demonstrates how to use the **service-module fail-open** command to configure the NME IPS to pass all traffic through the NME IPS when the hardware fails, but not to perform traffic inspection:

```
router(config)# interface ids-sensor 1/0
router(config-if)# service-module fail-open
router(config-if)#
```

Related Commands

Command	Description
interface *interface_name*	Lets you specify which interface should be monitored.

interface *interface_name*

To enter config-if mode, configure the interface for monitoring in promiscuous or inline mode, and apply a standard or extended ACL to inline monitoring, use the **interface** *interface_name* command in config mode.

interface *interface_name*

ids-service-module monitoring {**promiscuous** | **inline**} **access-list** *number*

Syntax Description

interface_name	The name of the router interface to be monitored.
ids-service-module	Configures IPS on the interface.
monitoring	Specifies how the NME IPS inspects traffic.
promiscuous	Specifies whether the NME IPS inspects traffic in promiscuous mode.
inline	Specifies whether the NME IPS inspects traffic in inline mode.
access-list	Specifies that you are applying a numbered or extended ACL to the inspected interface.
number	Number of the ACL,

Command Defaults

Command Modes

Config

Config-if

Command History

Release	Modification
12.4(20)T	This command was introduced.

Usage Guidelines

The **interface** *interface_name* command lets you enter config-if mode and configure the router to operate in inline or promiscuous mode for that interface.

Examples

The following example demonstrates how to use the **interface** command to enter config-if mode and configure monitoring for GigabitEthernet0/0 using ACL 101:

```
router(config)# interface GigabitEthernet0/0
router(config-if)# ids-service-module monitoring inline access-list 101
router(config-if)#
```

Related Commands

Command	Description
interface ids-sensor	Configures the IPS interface.

service-module ids-sensor

Caution When you reload the router, the NME IPS also reloads. To ensure that there is no loss of data on the NME IPS, make sure you shut down the module using the **shutdown** command before you use the **reload** command to reboot the router.

To prevent the Cisco IOS software from rebooting the NME IPS when the heartbeat is lost, to reboot, reset, enable console access to, shut down, see the statistics, and monitor the status of a module, use the **service-module ids-sensor** command in privileged EXEC mode.

service-module ids-sensor *slot/port* {**heartbeat-reset** {**enable** | **disable**} **reload** | **reset** | **session** | **shutdown** | **status**}

Syntax Description

slot	Number of the router chassis slot for the NME IPS.
/port	Port number of the NME IPS. **Note** The slash mark is required between the *slot* argument and the *unit* argument.
heartbeat-reset	Enables or disables the heartbeat reset. The default is enabled. **Note** Disabling the heartbeat reset prevents the router from resetting the NME IPS during system image installation if the process takes too long.
reload	Performs a graceful halt and reboot of the operating system on the NME IPS.
reset	Resets the hardware on the NME IPS. This command is usually used to recover from a shutdown.
session	Enables console access to the NME IPS from the router.
shutdown	Shuts down the IPS application running on the NME IPS.
statistics	Provides NME IPS statistics.
status	Provides information about the status of the IPS software.

Defaults

Command Modes Privileged EXEC

Command History

Release	Modification
12.4(20)T	This command was introduced.

Usage Guidelines

When the NME IPS is booted in failsafe mode or is undergoing an upgrade, you can use the **service-module ids heartbeat-reset** command to prevent a reboot during the process. If you leave the heartbeat reset enabled during an upgrade, you may lose the NME IPS heartbeat.

When the NME IPS heartbeat is lost, the router applies a fail-open or fail-close configuration option to the NME IPS and stops sending traffic to the NME IPS, and sets the NME IPS to error state. The router performs a hardware reset on the NME IPS and monitors the NME IPS until the heartbeat is reestablished.

If a confirmation prompt is displayed, press **Enter** to confirm the action or **n** to cancel.

Examples

The following example demonstrates how to disable or enable the reset action when the heartbeat is lost on an NME IPS in slot 1, port 0:

```
router# service-module ids-sensor 1/0 heartbeat-reset disable
```

The following example demonstrates how to enable the IDS heartbeat on the NME IPS:

```
router# service-module ids-sensor 1/0 heartbeat-reset enable
```

The status of the heartbeat-reset is displayed by using the **service-module ids** *slot/port* **status** command:

```
router# service-module ids-sensor 1/0 status
Service Module is Cisco IDS-Sensor 1/0
Service Module supports session via TTY line 194
Service Module heartbeat-reset is enabled
```

The following example demonstrates how to gracefully halt and reboot the operating system on the NME IPS:

```
router# service-module ids-sensor 1/0 reload

Do you want to proceed with reload?[confirm]
```

The following example demonstrates how to reset the hardware on an NME IPS. A warning is displayed.

```
router# service-module ids-sensor 1/0 reset

Use reset only to recover from shutdown or failed state
Warning: May lose data on the NVRAM, nonvolatile file system or unsaved configuration!

Do you want to reset?[confirm]
```

The following example demonstrates how to enable console access to the NME IPS operating system:

```
router# service-module ids-sensor 1/0 session
```

The following example demonstrates how to shut down IPS applications running on the NME IPS:

```
router# service-module ids-sensor 1/0 shutdown

Trying 10.10.10.1, 2129 ... Open
%SERVICEMODULE-5-SHUTDOWN2:Service module IDS-Sensor 1/0 shutdown complete
```

The following example demonstrates how todisplay IPS software statistics:

```
router# service-module ids-sensor 1/0 statistics
Module Reset Statistics:
  CLI reset count = 1
  CLI reload count = 0
  Registration request timeout reset count = 1
  Error recovery timeout reset count = 1
  Module registration count = 7
```

```
The last IOS initiated event was a cli reset at 20:18:36.038 UTC Tue Jan 16 2007
```

The following example demonstrates how to display the status of the IPS software on the NME IPS:

```
router# service-module ids-sensor 1/0 status

Service Module is Cisco IDS-Sensor1/0
Service Module supports session via TTY line 33
Service Module is in Steady state
Getting status from the Service Module, please wait...
Service Module Version information received, Major ver = 1, Minor ver= 1

Cisco Systems Intrusion Prevention System Network Module
  Software version:  6.1(1)E1
  Model:             NME-IPS
  Memory:            890996 KB
  Mgmt IP addr:      10.1.9.201
  Mgmt web ports:    443
  Mgmt TLS enabled:  true
```

Related Commands

Command	Description
ids-service-module monitoring	Enables IPS monitoring on a specified interface.

service-module ids-sensor bootmode

To enter failsafe or normal boot mode for the NME IPS, use the **service-module ids-sensor bootmode** command in privileged EXEC mode.

service-module ids-sensor *slot*/*port* **bootmode** {**failsafe** | **normal**}

Syntax Description

slot/	Number of the router chassis slot for the NME IPS. The slash mark (/) is required between the *slot* argument and the *port* argument.
port	Port number of the NME IPS.
failsafe	Enters failsafe boot mode on the NME IPS.
normal	Enters normal boot mode on the NME IPS.

Defaults

None

Command Modes

Privileged EXEC (#)

Command History

Release	Modification
12.4(20)T	This command was integrated into Cisco IOS Release 12.4(20)T.

Usage Guidelines

If a confirmation prompt is displayed, press **Enter** to confirm the action, or press **n** to cancel.

Examples

The following example demonstrates how to enter failsafe boot mode on an NME IPS in slot 1, port 0:

```
router# service-module ids-sensor 1/0 bootmode failsafe
```

The following example demonstrates how to enter normal boot mode on the NME IPS:

```
router# service-module ids-sensor 1/0 bootmode normal
```

Related Commands

Command	Description
ids-service-module monitoring	Enables IDS monitoring on a specified interface.

CHAPTER 21

Obtaining Software

This chapter provides information on obtaining Cisco IPS software for the sensor. It contains the following sections:

Caution The BIOS on Cisco IPS sensors is specific to Cisco IPS sensors and must only be upgraded under instructions from Cisco with BIOS files obtained from the Cisco website. Installing a non-Cisco or third-party BIOS on Cisco IPS sensors voids the warranty.

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 as needed. Major and minor updates are also posted periodically. Check Cisco.com regularly for the latest IPS software.

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 license to apply signature updates.

Downloading 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 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.

- Fill out the form and click **Submit**.

  The Cisco Systems Inc. Encryption Software Usage Handling and Distribution Policy appears.

- 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.

The File Download dialog box appears.

Step 11 Open the file or save it to your computer.

Step 12 Follow the instructions in the Readme to install the update.

Note Major and minor updates, service packs, recovery files, signature and signature engine updates are the same for all sensors. System image files are unique per platform.

For More Information

- For the procedure for obtaining and installing the license key, see Obtaining a License Key From Cisco.com, page 21-10.
- For an explanation of the IPS file versioning scheme, see IPS Software Versioning, page 21-3.

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 6.0 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 6.0(1) requires 5.*x*. With each major update there are corresponding system and recovery packages.

Note The 6.0(1) major update is only used to upgrade 5.*x* sensors to 6.0(1). If you are reinstalling 6.0(1) on a sensor that already has 6.0(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 6.0 is 6.1(1). 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 used for the release of defect fixes with no new enhancements. Service packs contain all service pack fixes since the last base version (minor or major) and the new 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 6.0(3) is released, and E3 is the latest engine level, the service pack is released as 6.0(3)E3.

Patch Release

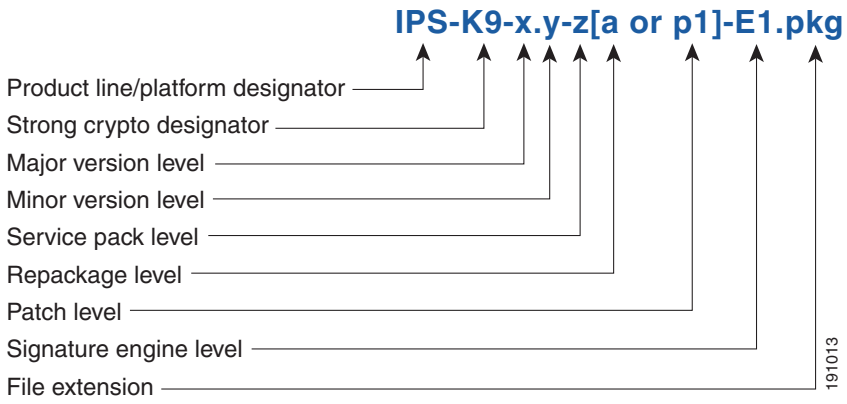
A patch release is used to address defects that are identified in the upgrade binaries after a software release. Rather than waiting until the next major or minor update, or service pack to address these defects, a patch can be posted. Patches include all prior patch releases within the associated service 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 5.0(1p1) requires 5.0(1).

Note Upgrading to a newer patch does not require you to uninstall the old patch. For example, you can upgrade from patch 5.0(1p1) to 5.0(1p2) without first uninstalling 5.0(1p1).

Figure 21-1 illustrates what each part of the IPS software file represents for major and minor updates, service packs, and patch releases.

Figure 21-1 IPS Software File Name for Major and Minor Updates, Service Packs, and Patch Releases

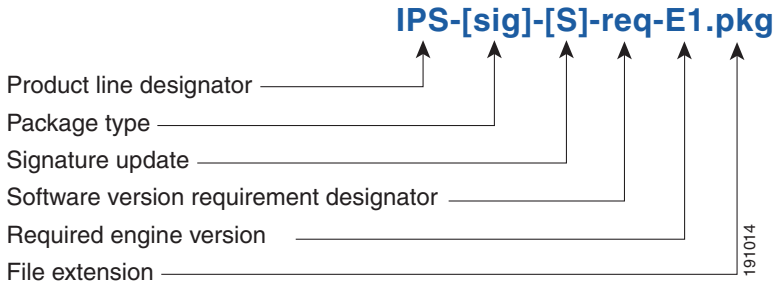


Signature Update

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.

Figure 21-2 illustrates what each part of the IPS software file represents for signature updates.

Figure 21-2 IPS Software File Name for Signature Updates

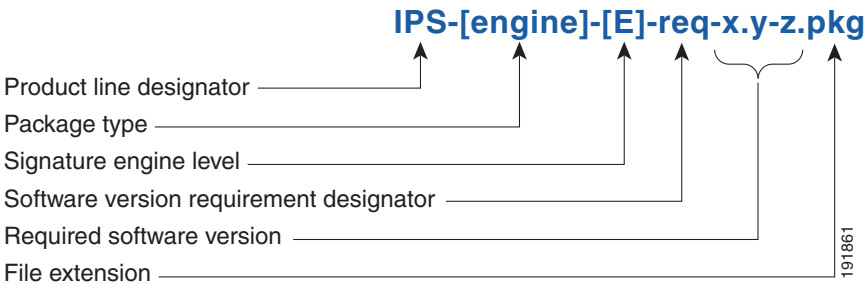


Signature Engine Update

A signature engine update is an executable file containing binary code to support new signature updates. Signature engine files require a specific service pack, which is also identified by the *req* designator.

Figure 21-3 illustrates what each part of the IPS software file represents for signature engine updates.

Figure 21-3 IPS Software File Name for Signature Engine Updates



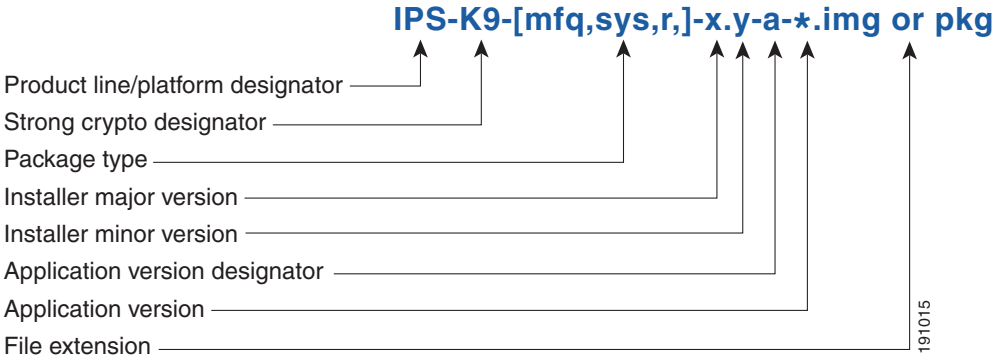
Recovery and System Image Files

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 21-4 illustrates what each part of the IPS software file represents for recovery and system image files.

Figure 21-4 IPS Software File Name for Recovery and System Image Files



Software Release Examples

Table 21-1 lists platform-independent Cisco IPS 6.*x* software release examples. Refer to the Readmes that accompany the software files for detailed instructions on how to install the files.

Table 21-1 Platform-Independent Release Examples

Release	Target Frequency	Identifier	Example Version	Example Filename
Signature update[1]	Weekly	sig	S353	IPS-sig-S353-req-E2.pkg
Signature engine update[2]	As needed	engine	E2	IPS-engine-E2-req-6.1-1.pkg
Service packs[3]	Semi-annually or as needed	—	6.1(3)	IPS-K9-6.1-3-E2.pkg
Minor version update[4]	Annually	—	6.1(1)	IPS-K9-6.1-1-E1.pkg **Note** The minor version update for the AIM IPS is IPS-AIM-K9-6.1-1-E1 .pkg.
Major version update[5]	Annually	—	6.0(1)	IPS-K9-6.0-1-E1.pkg
Patch release[6]	As needed	patch	6.0(1p1)	IPS-K9-patch-6.0-1pl-E1.pkg
Recovery package[7]	Annually or as needed	r	1.1-6.1(1)	IPS-K9-r-1.1-a-6.1-1-E2.pkg

1. Signature updates include the latest cumulative IPS signatures.
2. Signature engine updates add new engines or engine parameters that are used by new signatures in later signature updates.
3. Service packs include defect fixes.
4. Minor versions include new minor version features and/or minor version functionality.
5. Major versions include new major version functionality or new architecture.
6. Patch releases are for interim fixes.
7. 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 6.0(1), but the recovery partition image will be r 1.2.

Table 21-2 describes platform-dependent software release examples.

Table 21-2 Platform-Dependent Release Examples

Release	Target Frequency	Identifier	Supported Platform	Example Filename
System image[1]	Annually	sys	Separate file for each sensor platform	IPS-4240-K9-sys-1.1-a-6.1-1-E1.img
Maintenance partition image[2]	Annually	mp	IDSM2	c6svc-mp.2-1-2.bin.gz

Table 21-2 Platform-Dependent Release Examples (continued)

Release	Target Frequency	Identifier	Supported Platform	Example Filename
Bootloader	As needed	bl	AIM IPS NME IPS	pse_aim_x.y.z.bin pse_nm_x.y.z.bin (where x, y, z is the release number)
Mini-kernel	As needed	mini-kernel	AIM IPS NME IPS	pse_mini_kernel_1.1.10.64.bz2

1. The system image includes the combined recovery and application image used to reimage an entire sensor.
2. The maintenance partition image includes the full image for the IDSM2 maintenance partition. The file is installed from but does not affect the IDSM2 application partition.

Table 21-3 describes the platform identifiers used in platform-specific names.

Table 21-3 Platform Identifiers

Sensor Family	Identifier
IPS 4240 series	4240
IPS 4255 series	4255
IPS 4260 series	4260
IPS 4270-20 series	4270_20
IDS module for Catalyst 6K	IDSM2
IPS network module	AIM NME
AIP SSM	SSM_10 SSM_20 SSM_40

For More Information

For instructions on how to access these files on Cisco.com, see Obtaining Cisco IPS Software, page 21-1.

Upgrading Cisco IPS Software to 6.1

Observe the following when upgrading your sensor:

- The minimum required version for upgrading to 6.1(1) is 5.0(1) or later, which is available as a download from Cisco.com.
- Use the IPS-AIM-K9-6.1-1-E1.pkg upgrade file to upgrade the AIM IPS. For all other supported sensors, use the IPS-K9-6.1-1-E1.pkg upgrade file.
- After you upgrade any IPS software on your sensor, you must restart the IDM to see the latest software features.
- If you configured Auto Update for your sensor, copy the Cisco IPS 6.1(1)E1 update to the directory on the server that your sensor polls for updates. If you install an update on your sensor and the sensor is unusable after it reboots, you must reimage your sensor.

You can reimage your sensor in the following ways:

- For all sensors, use the **recover** command.
- For the IPS 4240, IPS 4255, IPS 4260, and IPS 4270-20, use the ROMMON to restore the system image.
- For the AIM IPS and the NME IPS, use the bootloader.
- For the IDSM2, reimage the application partition from the maintenance partition.

Note You cannot upgrade the IDSM (WS-X6381) to IPS 6.*x*. You must replace your IDSM (WS-X6381) with the IDSM2 (WS-SVC-IDSM2-K9), which supports version 6.1(1)E1.

- For the AIP SSM, reimage from the adaptive security appliance using the **hw-module module 1 recover configure/boot** command.

Caution When you install the system image for your sensor, all accounts are removed and the default account and password are reset to **cisco**.

For More Information

- For the procedure for accessing downloads on Cisco.com, see Obtaining Cisco IPS Software, page 21-1.
- For the procedure for using the **upgrade** command to upgrade the sensor, see Upgrading the Sensor, page 22-2.
- For the procedure for configuring automatic upgrades on the sensor, see Configuring Automatic Upgrades, page 22-6.
- For the procedure for using the **recover** command, see Recovering the Application Partition, page 22-11.
- For the procedures for using ROMMON to restore the system image, see Installing the IPS 4240 and the IPS 4255 System Images, page 22-14, Installing the IPS 4260 System Image, page 22-17, and Installing the IPS 4270-20 System Image, page 22-19.
- For the procedure for restoring the AIM IPS system image, see Installing the AIM IPS System Image, page 22-21.
- For the procedure for reimaging the IDSM2 application partition from the maintenance partition, see Installing the IDSM2 System Image, page 22-26.
- For the procedure for using the **hw-module module 1 recover configure/boot** command to reimage the AIP SSM, see Installing the AIP SSM System Image, page 22-24.
- For the procedure for restoring the NME IPS system image, see Installing the NME IPS System Image, page 22-38.

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.

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

Obtaining a License Key From Cisco.com

This section describes how to obtain a license key from Cisco.com and how to install it using the CLI or IDM. It contains the following topics:

Understanding Licensing

Although the sensor functions without the license key, you must have a license key to obtain signature updates. 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 IDM or IME, for IDM choose **Configuration > Sensor Management > Licensing**, and for 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:

- IDM Home window Licensing section on the Health tab
- IDM Licensing pane (**Configuration > Licensing**)
- IME Home page in the Device Details section on the Licensing tab
- License Notice at CLI login

Whenever you start IDM, 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 IDM, 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 IDM or 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 4240
- IPS 4255
- IPS 4260
- IPS 4270-20
- AIM IPS
- IDSM2
- NME IPS

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 the AIP SSM installed, or if you purchase the AIP SSM 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 purchased an ASA 5510 and then later wanted to add IPS and purchased an ASA-SSM-AIP-10-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 will change. You must then get a new license key for the new serial number.

Obtaining and Installing the License Key Using IDM or IME

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.

To obtain and install the license key, follow these steps:

Step 1 Log in to IDM or IME using an account with administrator privileges.

Step 2 For IDM choose **Configuration > Sensor Management > Licensing**. For IME choose **Configuration > *sensor_name* > Sensor Management > Licensing**.

The Licensing pane displays the status of the current license. If you have already installed your license, you can click **Download** to save it if needed.

Step 3 Obtain a license key by doing one of the following:

- Click the **Cisco.com** radio button to obtain the license from Cisco.com.

  IDM or IME contacts the license server on Cisco.com and sends the server the serial number to obtain the license key. This is the default method. Go to Step 4.

- Click the **License File** radio button to use a license file.

  To use this option, you must apply for a license key at this URL: www.cisco.com/go/license.

  The license key is sent to you in e-mail and you save it to a drive that IDM or IME can access. This option is useful if your computer cannot access Cisco.com. Go to Step 7.

Step 4 Click **Update License**, and in the Licensing dialog box, click **Yes** to continue.

The Status dialog box informs you that the sensor is trying to connect to Cisco.com. An Information dialog box confirms that the license key has been updated.

Step 5 Click **OK**.

Step 6 Go to www.cisco.com/go/license.

Step 7 Fill in the required fields.

Caution You must have the correct IPS device serial number because the license key only functions on the device with that number.

Your license key will be sent to the e-mail address you specified.

Step 8 Save the license key to a hard-disk drive or a network drive that the client running IDM or IME can access.

Step 9 Log in to IDM or IME.

Step 10 For IDM choose **Configuration > Sensor Management > Licensing**. For IME choose **Configuration > *sensor_name* > Sensor Management > Licensing**.

Step 11 Under Update License, click the **License File** radio button.

Step 12 In the Local File Path field, specify the path to the license file or click **Browse Local** to browse to the file.

Step 13 Browse to the license file and click **Open**.

Step 14 Click **Update License**.

For More Information

For more information about obtaining a Cisco Services for IPS service contract, see Service Programs for IPS Products, page 21-11.

Obtaining and Installing the License Key Using the CLI

Use the **copy** *source-url license_file_name* **license-key** command to copy the license key to your sensor.

The following options 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.

Note You cannot install an older license key over a newer license key.

The exact format of the source and destination URLs varies according to the file. Here are the valid types:

- 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.

Installing the License Key

To install the license key, follow these steps:

Step 1 Apply for the license key at this URL: 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 2 Fill in the required fields.

Note You must have the correct IPS device serial number because the license key only functions on the device with that number.

Your Cisco IPS Signature Subscription Service license key will be sent by e-mail to the e-mail address you specified.

Step 3 Save the license key to a system that has a web server, FTP server, or SCP server.

Step 4 Log in to the CLI using an account with administrator privileges.

Step 5 Copy the license key to the sensor.

```
sensor# copy scp://user@10.89.147.3://tftpboot/dev.lic license-key
Password: *******
```

Step 6 Verify the sensor is licensed.

```
sensor# show version
Application Partition:

Cisco Intrusion Prevention System, Version 6.1(1)E1

Host:
    Realm Keys          key1.0
Signature Definition:
    Signature Update    S391.0                  2008-04-16
    Virus Update        V1.2                    2005-11-24
OS Version:             2.4.30-IDS-smp-bigphys
Platform:               ASA-SSM-20
Serial Number:          P300000220
Sensor up-time is 3 days.
Using 1031888896 out of 2093682688 bytes of available memory (49% usage)
system is using 17.8M out of 29.0M bytes of available disk space (61% usage)
application-data is using 52.4M out of 166.6M bytes of available disk space (33% usage)
boot is using 37.8M out of 68.5M bytes of available disk space (58% usage)


MainApp          N-2007_JUN_19_16_45   (Release)   2007-06-19T17:10:20-0500   Running
AnalysisEngine   N-2007_JUN_19_16_45   (Release)   2007-06-19T17:10:20-0500   Running
CLI              N-2007_JUN_19_16_45   (Release)   2007-06-19T17:10:20-0500


Upgrade History:
```

```
  IPS-K9-6.1-1-E1   15:36:05 UTC Thu Apr 24 2008

Recovery Partition Version 1.1 - 6.1(1)E1

Host Certificate Valid from: 25-Apr-2008 to 26-Apr-2010

sensor#
```

Step 7 Copy your license key from a sensor to a server to keep a backup copy of the license.

```
sensor# copy license-key scp://user@10.89.147.3://tftpboot/dev.lic
Password: *******
sensor#
```

For More Information

- For the procedure for adding a remote host to the SSHM known hosts list, see Adding Hosts to the SSH Known Hosts List, page 4-42.
- For the procedure for adding a remote host to the trusted hosts list, see Adding TLS Trusted Hosts, page 4-46.
- For more information about obtaining a Cisco Services for IPS service contract, see Service Programs for IPS Products, page 21-11.

CHAPTER 22

Upgrading, Downgrading, and Installing System Images

This chapter describes how to upgrade, downgrade, and install system images. It contains the following sections:

Upgrades, Downgrades, and System Images

Note After you upgrade any IPS software on your sensor, you must restart the IDM to see the latest software features.

You can upgrade and downgrade the software on the sensor. Upgrading applies a service pack, signature update, signature engine update, minor version, major version, or recovery partition file. Downgrading removes the last applied service pack or signature update from the sensor.

Caution You cannot use the **downgrade** command to go from Cisco IPS 6.1 to 6.0. To revert to 6.0, you must reimage the sensor.

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.

To install a new system image on the sensor, use ROMMON, the bootloader file, or the maintenance partition depending on which platform you have.

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 file.

For More Information

- For the procedure for initializing the sensor, see Basic Sensor Setup, page 3-3.
- For the procedure for locating software on Cisco.com, see Obtaining Cisco IPS Software, page 21-1.

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:

- CMS - Apache Server (Tomcat)
- CMS - Apache Server (JRun)

For More Information

- For the procedure for downloading IPS software updates from Cisco.com, see Obtaining Cisco IPS Software, page 21-1.
- For the procedure for configuring automatic updates, see Configuring Automatic Upgrades, page 22-6.

Upgrading the Sensor

Note For the IDM procedure for upgrading the sensor, refer to Manually Updating the Sensor. For the IME procedure, refer to Manually Updating the Sensor.

This section explains how to use the **upgrade** command to upgrade the software on the sensor. It contains the following topics:

IPS 6.1 Upgrade Files

The following files are part of Cisco IPS 6.1(1)E1:

- Readme
  - IPS-6.1-1-E1.readme.txt
- Minor Version Upgrade File
  - IPS-K9-6.1-1-E1.pkg
  - IPS-AIM-K9-6.1-1-E1.pkg
- System Image Files
  - IPS-4240-K9-sys-1.1-a-6.1-1-E1.img
  - IPS-4255-K9-sys-1.1-a-6.1-1-E1.img
  - IPS-4260-K9-sys-1.1-a-6.1-1-E1.img
  - IPS-4270-K9-sys-1.1-a-6.1-1-E1.img
  - WS-SVC-IDSM2-K9-sys-1.1-a-6.1-1-E1.bin.gz
  - IPS-SSM_10-K9-sys-1.1-a-6.1-1-E1.img
  - IPS-SSM_20-K9-sys-1.1-a-6.1-1-E1.img
  - IPS-SSM_40-K9-sys-1.1-a-6.1-1-E1.img
  - IPS-AIM-K9-sys-1.1-a-6.1-1-E1.img
  - IPS-NME-K9-sys-1.1-a-6.1-1-E2.img
- Recovery Image Files
  - IPS-K9-r-1.1-a-6.1-1-E1.pkg
  - IPS-AIM-K9-r-1.1-a-6.1-1-E1.pkg
  - IPS-NME-K9-r-1.1-a-6.1-1-E2.pkg

For More Information

For the procedure for obtaining these files on Cisco.com, see Obtaining Cisco IPS Software, page 21-1.

upgrade Command and Options

Note For the IDM procedure for upgrading the sensor, refer to Manually Updating the Sensor. For the IME procedure, refer to Manually Updating the Sensor.

Use the **upgrade** *source-url* command to apply service pack, signature update, engine update, minor version, major version, or recovery partition file upgrades.

The following options apply:

- *source-url*—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.

Using the upgrade Command

Note For the IDM procedure for upgrading the sensor, refer to Manually Updating the Sensor. For the IME procedure, refer to Manually Updating the Sensor.

To upgrade the sensor, follow these steps:

Step 1 Download the appropriate file (for example, IPS-K9-6.1-1-E1.pkg) to an FTP, SCP, HTTP, or HTTPS server that is accessible from your sensor.

Note You must log in to Cisco.com using an account with cryptographic privileges to download the file. Do not change the filename. You must preserve the original filename for the sensor to accept the update.

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-K9-6.1-1-E1.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-K9-6.1-1-E1.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.

For More Information

- For a list of supported FTP and HTTP/HTTPS servers, see Supported FTP and HTTP/HTTPS Servers, page 22-2.
- For the procedure for locating software on Cisco.com and obtaining an account with cryptographic privileges, see Obtaining Cisco IPS Software, page 21-1.

Upgrading the Recovery Partition

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.

Note Recovery partition images are generated for major and minor updates and only in rare situations for service packs or signature updates.

Note The AIM IPS and the NME IPS have unique recovery images (IPS-AIM-K9-r-1.1-a-6.1-1-E1.pkg and IPS-NME-K9-r-1.1-a-6.1-1-E2.pkg) that you must use to upgrade the recovery partition.

To upgrade the recovery partition on your sensor, follow these steps:

Step 1 Download the recovery partition image file (IPS-K9-r-1.1-a-6.1-1.pkg) 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-K9-r-1.1-a-6.1-1-E1.pkg

sensor(config)#
upgrade ftp://user@server_ipaddress//upgrade_path/IPS-K9-r-1.1-a-6.1-1-E1.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 22-2.
- For the procedure for locating software on Cisco.com, see Obtaining Cisco IPS Software, page 21-1.
- For the procedure for using the **recover** command, see Using the recover Command, page 22-11.

Configuring Automatic Upgrades

Note For the IDM procedure for automatically upgrading the sensor, refer to Configuring Automatic Update. For the IME procedure, refer to Configuring Automatic Update.

This section describes how to configure the sensor to automatically look for upgrades in the upgrade directory. It contains the following topics:

Automatic Upgrades

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.

For More Information

For the procedure for locating software on Cisco.com, see Obtaining Cisco IPS Software, page 21-1.

auto-upgrade Command and Options

Note For the IDM procedure for automatically upgrading the sensor, refer to Configuring Automatic Update. For the IME procedure, refer to Configuring Automatic Update.

Use the **auto-upgrade-option enabled** command in the service host submode to configure automatic upgrades.

The following options apply:

- **cisco-server**—Enables automatic signature and engine updates from Cisco.com.
- **cisco-url**—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 where upgrade files are located on the file server.

  A leading '/' indicates an absolute path.
- **file-copy-protocol**— File copy protocol used to download files from the file server. The valid values are **ftp** or **scp**.

  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 of the file server.
- **password**— User password for Cisco server authentication.
- **schedule-option**—Schedules 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**—Configure the days of the week and times of day that automatic upgrades will be performed.

    days-of-week—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—Times of day at which auto-upgrades will begin. You can select multiple times. The valid value is *hh:mm[:ss]*.

- **periodic-schedule**—Configure the time that the first automatic upgrade should occur, and how long to wait between automatic upgrades.

  interval—The number of hours to wait between automatic upgrades. Valid values are 0 to 8760.

  start-time—The time of day to start the first automatic upgrade. The valid value is *hh:mm[:ss]*.

- **user-name**—Username for server authentication.
- **user-server**—Enables automatic upgrades from a user-defined server.

For More Information

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 4-42.

Using the auto-upgrade Command

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 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 198.133.219.25 port 443 for the initial automatic update connection to www.cisco.com, and you need 198.133.219.243 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:

```
sensor(config-hos-aut)# cisco-server enabled
```

Continue with Step 4.

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, which 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, which 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 22-2.
- 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 4-42.
- For the output of the **show statistics host** command, see Displaying Statistics, page 16-26.

Downgrading the Sensor

Caution You cannot use the **downgrade** command to go from Cisco IPS 6.1 to 6.0. To revert to 6.0, you must reimage the sensor. You can only use the **downgrade** command to downgrade from the latest service pack or signature update.

Use the **downgrade** command to remove the last applied service pack or signature upgrade from the sensor.

To remove the last applied service pack or signature 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 reboot the system and downgrade to
IPS-K9-sp.6.0-2-E1.pkg. Configuration changes made since the last upgrade will be lost and
the system may be rebooted.
Continue with downgrade?:
```

Step 4 Enter **yes** to continue with the downgrade.

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

This section explains how to recover the application partition, and contains the following topics:

Application Partition

You can recover the application partition image for the appliance 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 appliance.

Note 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**.

For More Information

For the procedure for upgrading the recovery partition to the most recent version, see Upgrading the Recovery Partition, page 22-5.

Using the recover Command

To recover the application partition image, follow these steps:

Step 1 Download the recovery partition image file (IPS-K9-r-1.1-a-6.1-1-E1.pkg) 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
```

Note To upgrade the recovery partition the sensor must already be running IPS 6.1(1) or later.

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 6.1(1)E1. 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 appliance 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 22-13.
- For the procedure for locating software on Cisco.com, see Obtaining Cisco IPS Software, page 21-1.
- For the procedure for using the **setup** command to initialize the sensor, see Basic Sensor Setup, page 3-3.

Installing System Images

This section contains the procedures for installing system images on the appliances and modules. It contains the following topics:

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 bootup.

Understanding 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 22-13.

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 180 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 IPS 4240 and the IPS 4255 System Images

You can install the IPS 4240 and the IPS 4255 system image by using the ROMMON on the appliance to TFTP the system image onto the compact flash device.

Note This procedure is for the IPS 4240, but is also applicable to the IPS 4255. The system image for the IPS 4255 has “4255” in the filename.

To install the IPS 4240 and the IPS 4255 system image, follow these steps:

Step 1 Download the IPS 4240 system image file (IPS-4240-K9-sys-1.1-a-6.1-1-E1.img) to the tftp root directory of a TFTP server that is accessible from your IPS 4240.

Note Make sure you can access the TFTP server location from the network connected to the Ethernet port of your IPS 4240.

Step 2 Boot the IPS 4240.

The console display resembles the following:

```
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
interface Device Table.
Bus Dev Func VendID DevID Class              Irq
00  00  00   8086   2578  Host Bridge
00  01  00   8086   2579  interface-to-interface Bridge
00  03  00   8086   257B  interface-to-interface Bridge
00  1C  00   8086   25AE  interface-to-interface 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  interface-to-interface 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-4240-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
```

The output on the configured system resembles the following:

```
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 4240
- Server—TFTP server IP address where the application image is stored
- Gateway—Gateway IP address used by the IPS 4240
- Port—Ethernet interface used for the IPS 4240 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 Management0/0, which corresponds to the MGMT interface of the IPS 4240.

```
rommon> PORT=interface_name
```

Step 6 If necessary, assign an IP address for the local port on the IPS 4240.

```
rommon> ADDRESS=ip_address
```

Note Use the same IP address that is assigned to the IPS 4240.

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-4240-K9-sys-1.1-a-6.1-1-E1.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-4240-K9-sys-1.1-a-6.1-1-E1.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 4240 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 4240. Be sure to use the IPS 4240 image.

For More Information

- For more information about TFTP servers, see TFTP Servers, page 22-13.
- For the procedure for locating software on Cisco.com, see Obtaining Cisco IPS Software, page 21-1.

Installing the IPS 4260 System Image

You can install the IPS 4260 system image by using the ROMMON on the appliance to TFTP the system image onto the flash device.

To install the IPS 4260 system image, follow these steps:

Step 1 Download the IPS 4260 system image file (IPS-4260-K9-sys-1.1-a-6.1-1-E1.img) to the tftp root directory of a TFTP server that is accessible from your IPS 4260.

Make sure you can access the TFTP server location from the network connected to your IPS 4260 Ethernet port.

Step 2 Boot the IPS 4260.

Step 3 Press **Ctrl-R** at the following prompt while the system is booting:

```
Evaluating Run Options...
```

Note You have five seconds to press **Ctrl-R**.

The console display resembles the following:

```
Assuming IPS-4260-K9 Platform
```

```
 2 Ethernet Interfaces detected

Cisco Systems ROMMON Version (1.0(11)1c) #26: Mon Mar 13 18:05:54 CST 2006

Platform IPS-4260-K9
Management0/0
Link is UP
MAC Address: 0004.23cc.6047


Use ? for help.
rommon #0>
```

Step 4 If necessary, change the port used for the TFTP download.

```
rommon #1> interface name
```

The port in use is listed just after the platform identification. In the example, port Management0/0 is being used.

Note The default port used for TFTP downloads is Management0/0, which corresponds with the command and control (MGMT) interface of the IPS 4260.

Note Ports Management0/0 (MGMT) and GigabitEthernet0/1 (GE 0/1) are labeled on the back of the chassis.

Step 5 Specify an IP address for the local port on the IPS 4260.

```
rommon> address ip_address
```

Note Use the same IP address that is assigned to the IPS 4260.

Step 6 Specify the TFTP server IP address.

```
rommon> server ip_address
```

Step 7 Specify the gateway IP address.

```
rommon> gateway ip_address
```

Step 8 Verify that you have access to the TFTP server by pinging it from the local Ethernet port:.

```
rommon> ping server_ip_address
rommon> ping server
```

Step 9 Specify the path and filename on the TFTP file server from which you are downloading the image.

```
rommon> file path/filename
```

UNIX example

```
rommon> file /system_images/IPS-4260-K9-sys-1.1-a-6.1-1-E1.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 file location.

Windows example

```
rommon> file <tftpboot_directory>IPS-4260-K9-sys-1.1-a-6.1-1-E1.img
```

Step 10 Download and install the system image.

```
rommon> tftp
```

Note The IPS 4260 reboots once during the reimaging process. Do not remove power from IPS 4260 during the update process or the upgrade can become corrupted.

For More Information

- For more information about TFTP servers, see TFTP Servers, page 22-13.
- For the procedure for locating software on Cisco.com, see Obtaining Cisco IPS Software, page 21-1.

Installing the IPS 4270-20 System Image

You can install the IPS 4270-20 system image by using the ROMMON on the appliance to TFTP the system image onto the compact flash device.

To install the IPS 4270-20 system image, follow these steps:

Step 1 Download the IPS 4270-20 system image file (IPS4270-20-K9-sys-1.1-a-6.1-1-E1.img) to the tftp root directory of a TFTP server that is accessible from your IPS 4270-20.

Note Make sure you can access the TFTP server location from the network connected to the Ethernet port of your IPS 4270-20.

Step 2 Boot IPS 4270-20.

The console display resembles the following:

```
Booting system, please wait...
Cisco Systems ROMMON Version (1.0(12)10) #7: Thu Jun 21 13:50:04 CDT 2007

ft_id_update: Invalid ID-PROM Controller Type (0x5df)

ft_id_update: Defaulting to Controller Type (0x5c2)
```

Note The controller type errors are a known issue and can be disregarded.

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
```

The output on the configured system resembles the following:

```
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—Local IP address of IPS 4270-20
- Server—TFTP server IP address where the application image is stored
- Gateway—Gateway IP address used by IPS 4270-20
- Port—Ethernet interface used for IPS 4270-20 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, assign an IP address for the local port on IPS 4270-20.

```
rommon> ADDRESS=ip_address
```

Note Use the same IP address that is assigned to IPS 4270-20.

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/IPS4270-20-K9-sys-1.1-a-6.1-1-E1.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\IPS4270-20-K9-sys-1.1-a-6.1-1-E1.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 IPS 4270-20 while the system image is being installed.

Note If the network settings are correct, the system downloads and boots the specified image on IPS 4270-20. Be sure to use the IPS 4270-20 image.

For More Information

- For more information about TFTP servers, see TFTP Servers, page 22-13.
- For the procedure for locating software on Cisco.com, see Obtaining Cisco IPS Software, page 21-1.

Installing the AIM IPS System Image

To install the AIM IPS system image, follow these steps:

Step 1 Download the AIM IPS system image file (IPS-AIM-K9-sys-1.1-6.1-1-E1.img), and place it on a TFTP server relative to the tftp root directory.

Note Make sure the network is configured so that the AIM IPS can access the TFTP server.

If no TFTP server is available, you can configure the router to operate as a TFTP server.

```
router# copy tftp: flash:
router# configure terminal
router(config)# tftp-server flash:IPS-AIM-K9-sys-1.1-6.0-3-E1.img
router(config)# exit
router#
```

Step 2 Disable the heartbeat reset.

```
router# service-module IDS-Sensor 0/slot_number heartbeat-reset disable
```

Note Disabling the heartbeat reset prevents the router from resetting the module during system image installation if the process takes too long.

Step 3 Session to the AIM IPS.

```
router# service-module IDS-Sensor 0/slot_number session
```

Note Use the **show configuration | include interface IDS-Sensor** command to determine the AIM IPS slot number.

Step 4 Suspend the session by pressing **Shift-Ctrl-6 X**.

You should see the `router#` prompt. If you do not see this prompt, try **Ctrl-6 X**.

Step 5 Reset the AIM IPS.

```
router# service-module IDS-Sensor 0/slot_number reset
```

You are prompted to confirm the **reset** command.

Step 6 Press **Enter** to confirm.

Step 7 Press **Enter** to resume the suspended session.

After displaying its version, the bootloader displays this prompt for 15 seconds:

```
Please enter '***' to change boot configuration:
```

Step 8 Enter ******* during the 15-second delay.

The bootloader prompt appears.

Step 9 Press **Enter** to session back to the AIM IPS.

Step 10 Configure the bootloader.

```
ServicesEngine bootloader> config

IP Address [10.89.148.188]>
Subnet mask [255.255.255.0]>
TFTP server [10.89.150.74]>
Gateway [10.89.148.254]>
Default boot [disk]>
Number cores [2]>
ServicesEngine boot-loader >
```

For each prompt, enter a value or accept the previously stored input that appears inside square brackets by pressing **Enter**.

Note The gateway IP address must match the IP address of the IDS-Sensor *slot/port* interface.

Note If you set up the module interfaces using the **unnumbered** command, the gateway IP address should be the IP address of the other router interface being used as part of the unnumbered command.

Caution The pathname for the AIM IPS image is full but relative to the tftp server root directory (typically /tftpboot).

Step 11 Start the bootloader.

```
ServicesEngine bootloader> upgrade
```

Step 12 Follow the bootloader instructions to install the software (choose option 1 and follow the wizard instructions).

Note In the following example, the AIM IPS IP address is 10.1.9.201. The imaging process accesses the AIM IPS image from the router TFTP server at IP address 10.1.9.1.

Example

```
Booting from flash...please wait.
Please enter '***' to change boot configuration:
11 ***
ServicesEngine boot-loader Version : 1.1.0
ServicesEngine boot-loader > config

IP Address [10.1.9.201]>
Subnet mask [255.255.255.0]>
TFTP server [10.1.9.1]>
Gateway [10.1.9.1]>
Default boot [disk]>
Number cores [2]>
ServicesEngine boot-loader > upgrade

Cisco Systems, Inc.
Services engine upgrade utility for AIM-IPS
-----
Main menu
1 - Download application image and write to USB Drive
2 - Download bootloader and write to flash
3 - Download minikernel and write to flash
r - Exit and reset card
x - Exit
Selection [123rx]
Download recovery image via tftp and install on USB Drive
TFTP server [10.1.9.1]>
full pathname of recovery image []:IPS-AIM-K9-sys-1.1-6.0-3-E1.img
Ready to begin
Are you sure [Y/N]
Returning TRUE
Press <CTRL-C> to abort.
octeth1:        Up     1Gbs Full duplex, (port  1)
octeth0:        Down  10Mbs Half duplex, (port  0)
```

```
Using octeth1 device
TFTP from server 10.1.9.1; our IP address is 10.1.9.201
Filename 'IPS-AIM-K9-sys-1.1-6.0-3-E1.img'.
Load address: 0x21000000
Loading: ##################################################################
         ##################################################################
         ##################################################################
         ##################################################################
         ##################################################################
         ##################################################################
         ##################################################################
         ##################################################################
         ##################################################################
         ##################################################################
         ######
         32 MB received
         ##################################################################
         #######################
done
```

Step 13 Suspend the session by pressing **Shift-Ctrl-6 X**.

You should see the `router#` prompt. If you do not see this prompt, try **Ctrl-6 X**.

Step 14 From the router CLI, clear the session.

```
router# service-module interface ids-sensor 0/slot_number session clear
```

Step 15 Enable the heartbeat reset.

```
router# service-module IDS-sensor 0/slot_number heartbeat-reset enable
```

For More Information

- For more information about TFTP servers, see TFTP Servers, page 22-13.
- For the procedure for setting up an unnumbered IP address, see Using an Unnumbered IP Address Interface, page 17-5.

Installing the AIP SSM System Image

This section describes how to install the AIP SSM system image, and contains the following topics:

Reimaging the AIP SSM

You can reimage the AIP SSM in one of the following ways:

- From ASA using the **hw-module module 1 recover configure/boot** command.
- Recovering the application image from the sensor CLI using the **recover application-partition** command.
- Upgrading the recovery image from the sensor CLI using the **upgrade** command.

For More Information

- For the procedure for using the **hw-module module 1 recover configure/boot** command, see Reimaging the AIP SSM Using the recover configure/boot Command, page 22-25.
- For the procedure for recovering the application partition, see Recovering the Application Partition, page 22-11.
- For the procedure for upgrading the recovery image, see Upgrading the Recovery Partition, page 22-5.

Reimaging the AIP SSM Using the recover configure/boot Command

To install the AIP SSM system image, follow these steps:

Step 1 Log in to the ASA.

Step 2 Enter enable mode.

```
asa# enable
```

Step 3 Configure the recovery settings for the AIP SSM.

```
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 4 Specify the TFTP URL for the system image.

```
Image URL [tftp://0.0.0.0/]:
```

Example

```
Image URL [tftp://0.0.0.0/]: tftp://10.89.146.1/IPS-SSM-K9-sys-1.1-a-6.1-1-E1.img
```

Step 5 Specify the command and control interface of the AIP SSM.

Note The port IP address is the management IP address of the AIP SSM.

```
Port IP Address [0.0.0.0]:
```

Example

```
Port IP Address [0.0.0.0]: 10.89.149.231
```

Step 6 Leave the VLAN ID at 0.

```
VLAN ID [0]:
```

Step 7 Specify the default gateway of the AIP SSM.

```
Gateway IP Address [0.0.0.0]:
```

Example:

```
Gateway IP Address [0.0.0.0]: 10.89.149.254
```

Step 8 Execute the recovery.

```
asa# hw-module module 1 recover boot
```

Step 9 Periodically check the recovery until it is complete.

Note The status reads `Recovery` during recovery and reads `Up` when reimaging is complete.

```
asa# show module 1

Mod Card Type                                    Model              Serial No.
--- -------------------------------------------- ------------------ -----------
  0 ASA 5540 Adaptive Security Appliance         ASA5540            P2B00000019
  1 ASA 5500 Series Security Services Module-20  ASA-SSM-20         P1D000004F4

Mod MAC Address Range                 Hw Version   Fw Version   Sw Version
--- --------------------------------- ------------ ------------ ---------------
  0 000b.fcf8.7b1c to 000b.fcf8.7b20  0.2          1.0(7)2      7.0(0)82
  1 000b.fcf8.011e to 000b.fcf8.011e  0.1          1.0(7)2      5.0(0.22)S129.0

Mod Status
--- ------------------
  0 Up Sys
  1 Up
asa#
```

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 10 Session to the AIP SSM and initialize the AIP SSM with the **setup** command.

For More Information

- For more information about TFTP servers, see TFTP Servers, page 22-13.
- For the procedure for initializing AIP SSM, see Advanced Setup for the AIP SSM, page 3-15.

Installing the IDSM2 System Image

This section describes how to the install the IDSM2 system image, and contains the following topics:

Understanding the IDSM2 System Image

If the IDSM2 application partition becomes unusable, you can reimage it from the maintenance partition. After you reimage the application partition of the IDSM2, you must initialize the IDSM2 using the **setup** command. When there is a new maintenance partition image file, you can reimage the maintenance partition from the application partition.

For More Information

For the procedure to use the **setup** command, see Advanced Setup for the IDSM2, page 3-20.

Installing the IDSM2 System Image for Catalyst Software

To install the system image, follow these steps:

Step 1 Download the IDSM2 system image file (WS-SVC-IDSM2-K9-sys-1.1-a-6.1-1.190-E0.1.bin.gz) to the FTP root directory of an FTP server that is accessible from your IDSM2.

Step 2 Log in to the switch CLI.

Step 3 Boot the IDSM2 to the maintenance partition.

```
console> (enable) reset module_number cf:1
```

Step 4 Log in to the maintenance partition CLI.

```
login: guest
Password: cisco
```

Note You must configure the maintenance partition on the IDSM2.

Step 5 Install the system image.

```
guest@hostname.localdomain# upgrade ftp://user@ftp server IP/directory
path/WS-SVC-IDSM2-K9-sys-1.1-a-6.1-1.190-E0.1.bin.gz
```

Step 6 Specify the FTP server password.

After the application partition file has been downloaded, you are asked if you want to proceed:

```
Upgrading will wipe out the contents on the hard disk. Do you want to proceed installing
it [y|n]:
```

Step 7 Enter **y** to continue.

When the application partition file has been installed, you are returned to the maintenance partition CLI.

Step 8 Exit the maintenance partition CLI and return to the switch CLI.

Step 9 Reboot the IDSM2 to the application partition.

```
console> (enable) reset module_number hdd:1
```

Step 10 When the IDSM2 has rebooted, check the software version.

Step 11 Log in to the application partition CLI and initialize the IDSM2.

For More Information

- For a list of supported FTP and HTTP/HTTPS servers, see Supported FTP and HTTP/HTTPS Servers, page 22-2.
- For the procedure for locating software on Cisco.com, see Obtaining Cisco IPS Software, page 21-1.
- For the procedure for configuration the maintenance partition on IDMS-2, see Configuring the IDSM2 Maintenance Partition for Catalyst Software, page 22-29 and Configuring the IDSM2 Maintenance Partition for Cisco IOS Software, page 22-33.
- For the procedure for initializing the IDSM2, see Advanced Setup for the IDSM2, page 3-20.

Installing the IDSM2 System Image for Cisco IOS Software

To install the system image, follow these steps:

Step 1 Download the IDSM2 system image file (WS-SVC-IDSM2-K9-sys-1.1-a-6.1-1.190-E0.1.bin.gz) to the FTP root directory of an FTP server that is accessible from your IDSM2.

Step 2 Log in to the switch CLI.

Step 3 Boot the IDSM2 to the maintenance partition.

```
router# hw-module module module_number reset cf:1
```

Step 4 Session to the maintenance partition CLI.

```
router# session slot slot_number processor 1
```

Step 5 Log in to the maintenance partition CLI.

```
login: guest
Password: cisco
```

Step 6 Configure the maintenance partition interface IP address.

```
guest@localhost.localdomain# ip address ip_address netmask
```

Note Choose an address that is appropriate for the VLAN on which the IDSM2 management interface is located based on the switch configuration.

Step 7 Configure the maintenance partition default gateway address.

```
guest@localhost.localdomain# ip gateway gateway_address
```

Step 8 Install the system image.

```
guest@hostname.localdomain# upgrade
ftp://user@ftp_server_ip_address/directory_path/WS-SVC-IDSM2-K9-sys-1.1-a-6.1-1.190-E0.
1.bin.gz-install
```

Step 9 Specify the FTP server password.

After the application partition file has been downloaded, you are asked if you want to proceed:

```
Upgrading will wipe out the contents on the hard disk.
Do you want to proceed installing it [y|n]:
```

Step 10 Enter **y** to continue.

When the application partition file has been installed, you are returned to the maintenance partition CLI.

Step 11 Exit the maintenance partition CLI and return to the switch CLI.

Step 12 Reboot the IDSM2 to the application partition.

```
router# hw-module module module_number reset hdd:1
```

Step 13 Verify that the IDSM2 is online and that the software version is correct and that the status is `ok`.

```
router# show module module_number
```

Step 14 Session to the IDSM2 application partition CLI.

```
router# session slot slot_number processor 1
```

Step 15 Initialize the IDSM2 using the **setup** command.

For More Information

- For a list of supported FTP and HTTP/HTTPS servers, see Supported FTP and HTTP/HTTPS Servers, page 22-2.
- For the procedure for locating software on Cisco.com, see Obtaining Cisco IPS Software, page 21-1.
- For the procedure for configuration the maintenance partition on IDMS-2, see Configuring the IDSM2 Maintenance Partition for Catalyst Software, page 22-29 and Configuring the IDSM2 Maintenance Partition for Cisco IOS Software, page 22-33.
- For the procedure for initializing the IDSM2, see Advanced Setup for the IDSM2, page 3-20.

Configuring the IDSM2 Maintenance Partition for Catalyst Software

To configure the IDSM2 maintenance partition, follow these steps:

Step 1 Log in to the switch CLI.

Step 2 Enter privileged mode.

```
console# enable
console(enable)#
```

Step 3 Reload the IDSM2.

```
console> (enable) reset module_number cf:1
```

Step 4 Session to the IDSM2.

```
console# session 9
Trying IDS-9...
Connected to IDS-9.
Escape character is '^]'.

Cisco Maintenance image
```

Note You cannot Telnet or SSH to the IDSM2 maintenance partition.You must session to it from the switch CLI.

Step 5 Log in as user **guest** and password **cisco**.

Note You can change the guest password, but we do not recommend it. If you forget the maintenance partition guest password, and you cannot log in to the IDSM2 application partition for some reason, the IDSM2 requires an RMA.

```
login: guest
Password: cisco

Maintenance image version: 2.1(2)

guest@idsm2.localdomain#
```

Step 6 View the IDSM2 maintenance partition host configuration.

```
guest@idsm2.localdomain# show ip

IP address      : 10.89.149.74
Subnet Mask     : 255.255.255.128
IP Broadcast    : 10.255.255.255
DNS Name        : idsm2.localdomain
Default Gateway : 10.89.149.126
Nameserver(s)   :

guest@idsm2.localdomain#
```

Step 7 Clear the IDSM2 maintenance partition host configuration (ip address, gateway, hostname).

```
guest@idsm2.localdomain# clear ip
guest@localhost.localdomain# show ip

IP address      : 0.0.0.0
Subnet Mask     : 0.0.0.0
IP Broadcast    : 0.0.0.0
DNS Name        : localhost.localdomain
Default Gateway : 0.0.0.0
Nameserver(s)   :

guest@localhost.localdomain#
```

Step 8 Configure the maintenance partition host configuration:

a. Specify the IP address.

```
guest@localhost.localdomain# ip address ip_address netmask
```

b. Specify the default gateway.

```
guest@localhost.localdomain# ip gateway gateway_ip_address
```

c. Specify the hostname.

```
guest@localhost.localdomain# ip host hostname
```

Step 9 View the maintenance partition host configuration.

```
guest@idsm2.localdomain# show ip

IP address      : 10.89.149.74
Subnet Mask     : 255.255.255.128
IP Broadcast    : 10.255.255.255
DNS Name        : idsm2.localdomain
Default Gateway : 10.89.149.126
Nameserver(s)   :
```

```
guest@idsm2.localdomain#
```

Step 10 Verify the image installed on the application partition.

```
guest@idsm2.localdomain# show images
Device name             Partition#              Image name
-----------             ----------               ----------
Hard disk(hdd)          1                       6.1(1)
guest@idsm2.localdomain#
```

Step 11 Verify the maintenance partition version (including the BIOS version).

```
guest@idsm2.localdomain# show version

Maintenance image version: 2.1(2)
mp.2-1-2.bin : Thu Nov 18 11:41:36 PST 2004 :
integ@kplus-build-lx.cisco.com

Line Card Number :WS-SVC-IDSM2-XL
Number of Pentium-class Processors : 2
BIOS Vendor: Phoenix Technologies Ltd.
BIOS Version: 4.0-Rel 6.0.9

Total available memory: 2012 MB
Size of compact flash: 61 MB
Size of hard disk: 19077 MB
Daughter Card Info: Falcon rev 3, FW ver 2.0.3.0 (IDS), SRAM  8 MB, SDRAM  256 MB

guest@idsm2.localdomain#
```

Step 12 Upgrade the application partition.

```
guest@idsm2.localdomain# upgrade
ftp://jsmith@10.89.146.11//RELEASES/Latest/6.1-1/WS-SVC-IDSM2-K9-sys-1.1-a-6.1-1.190-E0.
1.bin.gz
Downloading the image. This may take several minutes...
Password for jsmith@10.89.146.114:
500 'WS-SVC-IDSM2-K9-sys-1.1-a-6.0-0.190-E0.1.bin.gz': command not understood.

ftp://jsmith@10.89.146.11//RELEASES/Latest/6.1-1/WS-SVC-IDSM2-K9-sys-1.1-a-6.1-1.190-E0.1.
bin.gz (unknown size)
/tmp/upgrade.gz           [|]   28616K
29303086 bytes transferred in 5.34 sec (5359.02k/sec)

Upgrade file
ftp://jsmith@10.89.146.114//RELEASES/Latest/6.1-1/WS-SVC-IDSM2-K9-sys-1.1-a-6.1-1.190-E0.1
.bin.gz is downloaded.
Upgrading will wipe out the contents on the storage media.
Do you want to proceed installing it [y|N]:
```

Step 13 Enter **y** to proceed with the upgrade.

```
Proceeding with upgrade. Please do not interrupt.
If the upgrade is interrupted or fails, boot into maintenance image again and restart
upgrade.

Creating IDS application image file...

Initializing the hard disk...
Applying the image, this process may take several minutes...
Performing post install, please wait...
Application image upgrade complete. You can boot the image now.
guest@idsm3.localdomain#
```

Step 14 Display the upgrade log.

```
guest@idsm3.localdomain# show log upgrade

Upgrading the line card on Fri Mar 11 21:21:53 UTC 2005
Downloaded upgrade image
ftp://jsmith@10.89.146.114//RELEASES/Latest/6.1-1/WS-SVC-IDSM2-K9-sys-1.1-a-6.1-1.190-E0.1
.bin.gz
Extracted the downloaded file
Proceeding with image upgrade.
Fri Mar 11 21:22:06 2005 : argv1 = 0, argv2 = 0, argv3 = 3, argv4 = 1
Fri Mar 11 21:22:06 2005 : Creating IDS application image file...
Fri Mar 11 21:22:06 2005 : footer: XXXXXXXXXXXXXXXX
Fri Mar 11 21:22:06 2005 : exeoff: 0000000000031729
Fri Mar 11 21:22:06 2005 : image: 0000000029323770
Fri Mar 11 21:22:06 2005 : T: 29323818, E: 31729, I: 29323770
Fri Mar 11 21:22:07 2005 : partition: /dev/hdc1
Fri Mar 11 21:22:07 2005 : startIDSAppUpgrade:Image: /tmp/cdisk.gz
Fri Mar 11 21:22:07 2005 : startIDSAppUpgrade:Device: /dev/hdc1
Fri Mar 11 21:22:07 2005 : startIDSAppUpgrade:Install type: 1
Fri Mar 11 21:22:07 2005 : Initializing the hard disk...
Fri Mar 11 21:22:07 2005 : Required disk size: 524288 Kb (blocks)
Fri Mar 11 21:22:07 2005 : Available disk size: 19535040 Kb (blocks)
Fri Mar 11 21:22:13 2005 : Partitions created on '/dev/hdc'.
Fri Mar 11 21:22:13 2005 : Device '/dev/hdc' verified for OK.
Fri Mar 11 21:22:19 2005 : Created ext2 fileSystem on '/dev/hdc1'.
Fri Mar 11 21:22:19 2005 : Directory '/mnt/hd/' created.
Fri Mar 11 21:22:19 2005 : Partition '/dev/hdc1' mounted.
Fri Mar 11 21:22:19 2005 : Finished initializing the hard disk.
Fri Mar 11 21:22:19 2005 : Applying the image, this process may take several minutes...
Fri Mar 11 21:22:19 2005 : Directory changed to '/mnt/hd'.
Fri Mar 11 21:22:20 2005 : Performing post install, please wait...
Fri Mar 11 21:22:20 2005 : File /mnt/hd/post-install copied to /tmp/post-install.
Fri Mar 11 21:22:20 2005 : Directory changed to '/tmp'.
Fri Mar 11 21:22:28 2005 : Partition '/dev/hdc1' unmounted.
Fri Mar 11 21:22:28 2005 : Directory changed to '/tmp'.
Application image upgrade complete. You can boot the image now.
Partition upgraded successfully
guest@idsm2.localdomain#
```

Step 15 Clear the upgrade log.

```
guest@idsm2.localdomain# clear log upgrade
Cleared log file successfully
```

Step 16 Display the upgrade log.

```
guest@idsm2.localdomain# show log upgrade
guest@idsm2.localdomain#
```

Step 17 Ping another computer:.

```
guest@idsm2.localdomain# ping 10.89.146.114
PING 10.89.146.114 (10.89.146.114) from 10.89.149.74 : 56(84) bytes of data.
64 bytes from 10.89.146.114: icmp_seq=0 ttl=254 time=381 usec
64 bytes from 10.89.146.114: icmp_seq=1 ttl=254 time=133 usec
64 bytes from 10.89.146.114: icmp_seq=2 ttl=254 time=129 usec
64 bytes from 10.89.146.114: icmp_seq=3 ttl=254 time=141 usec
64 bytes from 10.89.146.114: icmp_seq=4 ttl=254 time=127 usec

--- 10.89.146.114 ping statistics ---
5 packets transmitted, 5 packets received, 0% packet loss
round-trip min/avg/max/mdev = 0.127/0.182/0.381/0.099 ms
guest@idsm2.localdomain#
```

Step 18 Reset the IDSM2.

Note You cannot specify a partition when issuing the **reset** command from the maintenance partition. The IDSM2 boots to whichever partition is specified in the boot device variable. If the boot device variable is blank, the IDSM2 boots to the application partition.

```
guest@idsm2.localdomain# reset
guest@idsm2.localdomain#
2005 Mar 11 21:55:46 CST -06:00 %SYS-4-MOD_SHUTDOWNSTART:Module 9 shutdown in progress. Do
not remove module until shutdown completes

Broadcast message from root Fri Mar 11 21:55:47 2005...

The system is going down for system halt NOW !!
console> (enable)#
```

For More Information

For a list of supported FTP and HTTP/HTTPS servers, see Supported FTP and HTTP/HTTPS Servers, page 22-2.

Configuring the IDSM2 Maintenance Partition for Cisco IOS Software

To configure the IDSM2 maintenance partition, follow these steps:

Step 1 Log in to the switch CLI.

Step 2 Session to the IDSM2.

```
router# session slot 11 processor 1
The default escape character is Ctrl-^, then x.
You can also type 'exit' at the remote prompt to end the session
Trying 127.0.0.111 ... Open

Cisco Maintenance image
```

Note You cannot Telnet or SSH to the IDSM2 maintenance partition.You must session to it from the switch CLI.

Step 3 Log in as user **guest** and password **cisco**.

Note You can change the guest password, but we do not recommend it. If you forget the maintenance partition guest password, and you cannot log in to the IDSM2 application partition for some reason, you will have to RMA the IDSM2.

```
login: guest
password: cisco

Maintenance image version: 2.1(2)

guest@idsm2.localdomain#
```

Step 4 View the maintenance partition host configuration.

```
guest@idsm2.localdomain# show ip

IP address      : 10.89.149.74
Subnet Mask     : 255.255.255.128
IP Broadcast    : 10.255.255.255
DNS Name        : idsm2.localdomain
Default Gateway : 10.89.149.126
Nameserver(s)   :

guest@idsm2.localdomain#
```

Step 5 Clear the maintenance partition host configuration (ip address, gateway, hostname).

```
guest@idsm2.localdomain# clear ip
guest@localhost.localdomain# show ip

IP address      : 0.0.0.0
Subnet Mask     : 0.0.0.0
IP Broadcast    : 0.0.0.0
DNS Name        : localhost.localdomain
Default Gateway : 0.0.0.0
Nameserver(s)   :

guest@localhost.localdomain#
```

Step 6 Configure the maintenance partition host configuration:

a. Specify the IP address.

```
guest@localhost.localdomain# ip address ip_address netmask
```

b. Specify the default gateway.

```
guest@localhost.localdomain# ip gateway gateway_ip_address
```

c. Specify the hostname.

```
guest@localhost.localdomain# ip host hostname
```

Step 7 View the maintenance partition host configuration.

```
guest@idsm2.localdomain# show ip

IP address      : 10.89.149.74
Subnet Mask     : 255.255.255.128
IP Broadcast    : 10.255.255.255
DNS Name        : idsm2.localdomain
Default Gateway : 10.89.149.126
Nameserver(s)   :

guest@idsm2.localdomain#
```

Step 8 Verify the image installed on the application partition.

```
guest@idsm2.localdomain# show images
Device name             Partition#              Image name
-----------             ----------               ----------
Hard disk(hdd)          1                       6.1(1)
guest@idsm2.localdomain#
```

Step 9 Verify the maintenance partition version (including the BIOS version).

```
guest@idsm2.localdomain# show version
```

```
Maintenance image version: 2.1(2)
mp.2-1-2.bin : Thu Nov 18 11:41:36 PST 2004 :
integ@kplus-build-lx.cisco.com

Line Card Number :WS-SVC-IDSM2-XL
Number of Pentium-class Processors : 2
BIOS Vendor: Phoenix Technologies Ltd.
BIOS Version: 4.0-Rel 6.0.9

Total available memory: 2012 MB
Size of compact flash: 61 MB
Size of hard disk: 19077 MB
Daughter Card Info: Falcon rev 3, FW ver 2.0.3.0 (IDS), SRAM  8 MB, SDRAM  256 MB

guest@idsm2.localdomain#
```

Step 10 Upgrade the application partition.

```
guest@idsm2.localdomain# upgrade
ftp://jsmith@10.89.146.11//RELEASES/Latest/6.1-1/WS-SVC-IDSM2-K9-sys-1.1-a-6.1-1-E1.img
Downloading the image. This may take several minutes...
Password for jsmith@10.89.146.114:
500 'SIZE WS-SVC-IDSM2-K9-sys-1.1-a-6.1-1.bin.gz': command not understood.

ftp://jsmith@10.89.146.11//RELEASES/Latest/6.1-1/WS-SVC-IDSM2-K9-sys-1.1-a-6.1-1-E1.img
(unknown size)
/tmp/upgrade.gz           [|]   28616K
29303086 bytes transferred in 5.34 sec (5359.02k/sec)

Upgrade file
ftp://jsmith@10.89.146.114//RELEASES/Latest/6.1-1/WS-SVC-IDSM2-K9-sys-1.1-a-6.1-1-E1.img
is downloaded.
Upgrading will wipe out the contents on the storage media.
Do you want to proceed installing it [y|N]:
```

Step 11 Enter **y** to proceed with the upgrade.

```
Proceeding with upgrade. Please do not interrupt.
If the upgrade is interrupted or fails, boot into maintenance image again and restart
upgrade.

Creating IDS application image file...

Initializing the hard disk...
Applying the image, this process may take several minutes...
Performing post install, please wait...
Application image upgrade complete. You can boot the image now.
guest@idsm3.localdomain#
```

Step 12 Display the upgrade log.

```
guest@idsm3.localdomain# show log upgrade

Upgrading the line card on Fri Mar 11 21:21:53 UTC 2005
Downloaded upgrade image
ftp://jsmith@10.89.146.114//RELEASES/Latest/6.1-1/WS-SVC-IDSM2-K9-sys-1.1-a-6.1-1-E1.img
Extracted the downloaded file
Proceeding with image upgrade.
Fri Mar 11 21:22:06 2005 : argv1 = 0, argv2 = 0, argv3 = 3, argv4 = 1
Fri Mar 11 21:22:06 2005 : Creating IDS application image file...
Fri Mar 11 21:22:06 2005 : footer: XXXXXXXXXXXXXXXX
Fri Mar 11 21:22:06 2005 : exeoff: 0000000000031729
Fri Mar 11 21:22:06 2005 : image: 0000000029323770
Fri Mar 11 21:22:06 2005 : T: 29323818, E: 31729, I: 29323770
Fri Mar 11 21:22:07 2005 : partition: /dev/hdc1
```

```
Fri Mar 11 21:22:07 2005 : startIDSAppUpgrade:Image: /tmp/cdisk.gz
Fri Mar 11 21:22:07 2005 : startIDSAppUpgrade:Device: /dev/hdc1
Fri Mar 11 21:22:07 2005 : startIDSAppUpgrade:Install type: 1
Fri Mar 11 21:22:07 2005 : Initializing the hard disk...
Fri Mar 11 21:22:07 2005 : Required disk size: 524288 Kb (blocks)
Fri Mar 11 21:22:07 2005 : Available disk size: 19535040 Kb (blocks)
Fri Mar 11 21:22:13 2005 : Partitions created on '/dev/hdc'.
Fri Mar 11 21:22:13 2005 : Device '/dev/hdc' verified for OK.
Fri Mar 11 21:22:19 2005 : Created ext2 fileSystem on '/dev/hdc1'.
Fri Mar 11 21:22:19 2005 : Directory '/mnt/hd/' created.
Fri Mar 11 21:22:19 2005 : Partition '/dev/hdc1' mounted.
Fri Mar 11 21:22:19 2005 : Finished initializing the hard disk.
Fri Mar 11 21:22:19 2005 : Applying the image, this process may take several minutes...
Fri Mar 11 21:22:19 2005 : Directory changed to '/mnt/hd'.
Fri Mar 11 21:22:20 2005 : Performing post install, please wait...
Fri Mar 11 21:22:20 2005 : File /mnt/hd/post-install copied to /tmp/post-install.
Fri Mar 11 21:22:20 2005 : Directory changed to '/tmp'.
Fri Mar 11 21:22:28 2005 : Partition '/dev/hdc1' unmounted.
Fri Mar 11 21:22:28 2005 : Directory changed to '/tmp'.
Application image upgrade complete. You can boot the image now.
Partition upgraded successfully
guest@idsm2.localdomain#
```

Step 13 Clear the upgrade log.

```
guest@idsm2.localdomain# clear log upgrade
Cleared log file successfully
```

Step 14 Display the upgrade log.

```
guest@idsm2.localdomain# show log upgrade
guest@idsm2.localdomain#
```

Step 15 Ping another computer.

```
guest@idsm2.localdomain# ping 10.89.146.114
PING 10.89.146.114 (10.89.146.114) from 10.89.149.74 : 56(84) bytes of data.
64 bytes from 10.89.146.114: icmp_seq=0 ttl=254 time=381 usec
64 bytes from 10.89.146.114: icmp_seq=1 ttl=254 time=133 usec
64 bytes from 10.89.146.114: icmp_seq=2 ttl=254 time=129 usec
64 bytes from 10.89.146.114: icmp_seq=3 ttl=254 time=141 usec
64 bytes from 10.89.146.114: icmp_seq=4 ttl=254 time=127 usec

--- 10.89.146.114 ping statistics ---
5 packets transmitted, 5 packets received, 0% packet loss
round-trip min/avg/max/mdev = 0.127/0.182/0.381/0.099 ms
guest@idsm2.localdomain#
```

Step 16 Reset the IDSM2.

Note You cannot specify a partition when issuing the **reset** command from the maintenance partition. The IDSM2 boots to whichever partition is specified in the boot device variable. If the boot device variable is blank, the IDSM2 boots to the application partition.

```
guest@idsm2.localdomain# reset
guest@idsm2.localdomain#
Broadcast message from root Fri Mar 11 22:04:53 2005...
```

```
The system is going down for system halt NOW !!

[Connection to 127.0.0.111 closed by foreign host]
router#
```

For More Information

For a list of supported FTP and HTTP/HTTPS servers, see Supported FTP and HTTP/HTTPS Servers, page 22-2.

Upgrading the IDSM2 Maintenance Partition for Catalyst Software

To upgrade the maintenance partition, follow these steps:

Step 1 Download the IDSM2 maintenance partition file (c6svc-mp.2-1-2.bin.gz) to the FTP root directory of an FTP server that is accessible from your IDSM2.

Step 2 Session to the IDSM2 from the switch.

console>(enable) **session** *slot_number*

Step 3 Log in to the IDSM2 CLI.

Step 4 Enter configuration mode.

idsm2# **configure terminal**

Step 5 Upgrade the maintenance partition.

idsm2(config)# **upgrade**
ftp://*user@ftp_server_IP_address/directory_path*/**c6svc-mp.2-1-2.bin.gz**

You are asked whether you want continue.

Step 6 Enter the FTP server password.

Step 7 Enter **y** to continue.

The maintenance partition file is upgraded.

For More Information

- For a list of supported FTP and HTTP/HTTPS servers, see Supported FTP and HTTP/HTTPS Servers, page 22-2.
- For the procedure for locating software on Cisco.com, see Obtaining Cisco IPS Software, page 21-1.

Upgrading the IDSM2 Maintenance Partition for Cisco IOS Software

To upgrade the maintenance partition, follow these steps:

Step 1 Download the IDSM2 maintenance partition file (c6svc-mp.2-1-2.bin.gz) to the FTP root directory of an FTP server that is accessible from your IDSM2.

Step 2 Log in to the switch CLI.

Step 3 Session in to the application partition CLI.

```
router# session slot slot_number processor 1
```

Step 4 Log in to the IDSM2.

Step 5 Enter configuration mode.

```
idsm2# configure terminal
```

Step 6 Upgrade the maintenance partition.

```
idsm2(config)# upgrade
ftp://user@ftp_server_IP_address/directory_path/c6svc-mp.2-1-2.bin.gz
```

Step 7 Specify the FTP server password.

```
Password: ********
```

You are prompted to continue.

```
Continue with upgrade?:
```

Step 8 Enter **yes** to continue.

For More Information

- For a list of supported FTP and HTTP/HTTPS servers, see Supported FTP and HTTP/HTTPS Servers, page 22-2.
- For the procedure for locating software on Cisco.com, see Obtaining Cisco IPS Software, page 21-1.

Installing the NME IPS System Image

Note Use the **show configuration | include interface ids-sensor** command to determine the NME IPS slot number.

To install the NME IPS system image, follow these steps:

Step 1 Download the NME IPS system image file (IPS-NME-K9-sys-1.1-6.1-1-E2.img), and place it on a TFTP server relative to the tftp root directory.

Note Make sure the network is configured so that the NME IPS can access the TFTP server.

If no TFTP server is available, you can configure the router to operate as a TFTP server.

```
router# copy tftp: flash:
router# configure terminal
router(config)# tftp-server flash:IPS-NME-K9-sys-1.1-6.1-1-E2.img
router(config)# exit
router#
```

Step 2 Disable the heartbeat reset.

```
router# service-module ids-sensor 1/0 heartbeat-reset disable
```

Note Disabling the heartbeat reset prevents the router from resetting the module during system image installation if the process takes too long.

Step 3 Session to the NME IPS.

```
router# service-module ids-sensor 1/0 session
```

Step 4 Suspend the session by pressing **Shift-Ctrl-6 X**.

You should see the `router#` prompt. If you do not see this prompt, try **Ctrl-6 X**.

Step 5 Reset the NME IPS.

```
router# service-module ids-sensor 1/0 reset
```

You are prompted to confirm the **reset** command.

Step 6 Press **Enter** to confirm.

Step 7 Press **Enter** to resume the suspended session. After displaying its version, the bootloader displays this prompt for 15 seconds:

```
Please enter '***' to change boot configuration:
```

Step 8 Enter ******* during the 15-second delay.

The bootloader prompt appears.

Step 9 Press **Enter** to session back to the NME IPS.

Step 10 Configure the bootloader.

```
ServicesEngine bootloader> config

IP Address [10.89.148.195]>
Subnet mask [255.255.255.0]>
TFTP server [10.89.150.74]>
Gateway [10.89.148.254]>
Default boot [disk]>
Number cores [2]>
ServicesEngine boot-loader >
```

For each prompt, enter a value or accept the previously stored input that appears inside square brackets by pressing **Enter**.

Caution The pathname for the NME IPS image is full but relative to the tftp server root directory (typically /tftpboot).

Step 11 Start the bootloader.

```
ServicesEngine bootloader> upgrade
```

Step 12 Follow the bootloader instructions to install the software (choose option 1 and follow the wizard instructions).

Example

```
Booting from flash...please wait.
Please enter '***' to change boot configuration:
12 ***
ServicesEngine boot-loader Version : 1.2.0
ServicesEngine boot-loader > config
```

```
IP Address [10.89.148.195]>
Subnet mask [255.255.255.0]>
TFTP server [10.89.150.74]>
Gateway [10.89.148.254]>
Default boot [disk]>
Number cores [2]>
ServicesEngine boot-loader > upgrade

Cisco Systems, Inc.
Services engine upgrade utility for NM-IPS
-----
Main menu
1 - Download application image and write to USB Drive
2 - Download bootloader and write to flash
3 - Download minikernel and write to flash
r - Exit and reset card
x - Exit
Selection [123rx]
Download recovery image via tftp and install on USB Drive
TFTP server [10.89.150.74]>
full pathname of recovery image []:test/sensor/6.1-1-E2/IPS-NME-K9-sys-1.1-a-6.1-1-E2.img
Ready to begin
Are you sure [Y/N]
Press <CTRL-C> to abort.
octeth0:        Up     1Gbs Full duplex, (port  0)
octeth1:        Up     1Gbs Full duplex, (port  1)
Using octeth0 device
TFTP from server 10.89.150.74; our IP address is 10.89.148.195; sending through gateway
10.89.148.254
Filename 'test/sensor/6.1-1-E2/IPS-NME-K9-sys-1.1-a-6.1-1-E2.img'.
Load address: 0x21000000
Loading: octeth0: Down   1Gbs Half duplex, (port  0)
octeth0: Down   1Gbs Full duplex, (port  0)
octeth0: Up     1Gbs Full duplex, (port  0)
T T T T T T T T T ##################################################################
         ##################################################################
         ##################################################################
         ##################################################################
         ##################################################################
         ##################################################################
         ##################################################################
         ##################################################################
         ##################################################################
         ##################################################################
         ######
         32 MB received
         ##################################################################
         ##################################################################
         ##################
done
```

Step 13 Suspend the session by pressing **Shift-Ctrl-6 X**.

You should see the `router#` prompt. If you do not see this prompt, try **Ctrl-6 X**.

Step 14 From the router CLI, clear the session.

```
router# service-module interface ids-sensor 1/0 session clear
```

Step 15 Enable the heartbeat reset:

```
router# service-module IDS-sensor 1/0 heartbeat-reset enable
```

For More Information

- For more information about TFTP servers, see TFTP Servers, page 22-13.
- For the procedure for locating software on Cisco.com, see Obtaining Cisco IPS Software, page 21-1.

APPENDIX A

System Architecture

This chapter describes the Cisco IPS 6.1 system architecture, and contains the following topics:

Purpose of the Cisco IPS

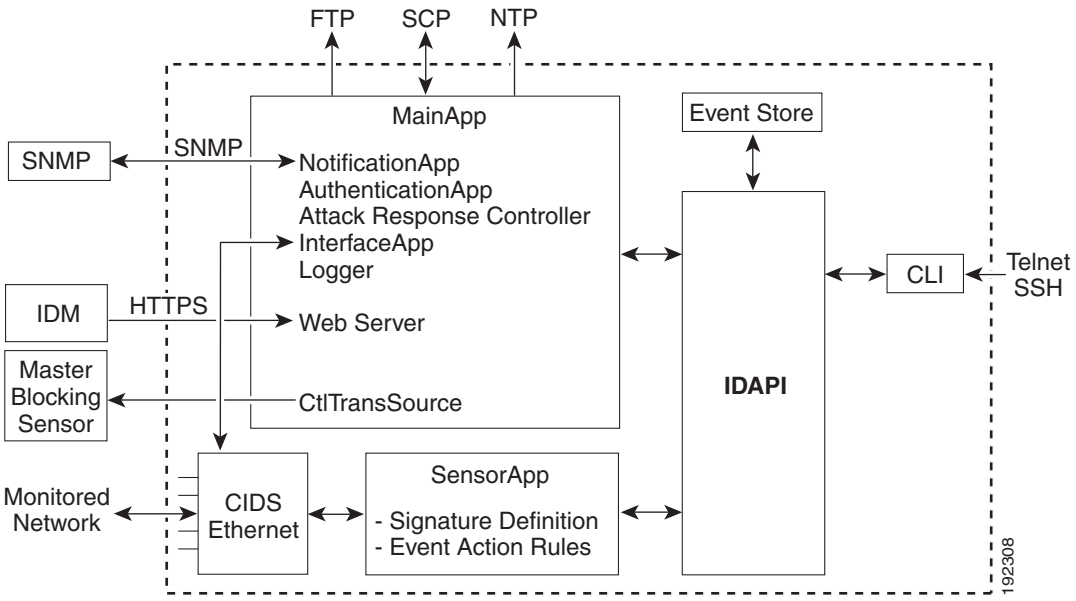
The purpose of Cisco IPS is to detect and prevent malicious network activity. You can install Cisco IPS software on two platforms: appliances and the modules. Cisco IPS contains a management application and a monitoring application. IDM is a network management JAVA application that you can use to manage and monitor the IPS. IME is an IPS network monitoring JAVA application that you can use to view IPS events. IME also contains the IDM configuration component. IDM and IME communicate with the IPS using HTTP or HTTPS and are hosted on your computer.

System Design

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.

Figure A-1 System Design



System Applications

Note Each application has its own configuration file in XML format.

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 except for AIP SSC-5, which has a fixed size of 10 MB.

  - InterfaceApp—Handles bypass and physical settings and defines paired interfaces. Physical settings are speed, duplex, and administrative state.
  - LogApp—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. 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. NotificationApp uses the public domain SNMP agent. SNMP GETs provide information about the general health of the sensor.
  - Web Server (HTTP RDEP2 SDEE server)—Provides a web interface and communication with other IPS devices through RDEP2 and SDEE protocols using several servlets to provide 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.
- 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 IDAPI. Remote applications (other sensors, management applications, and third-party software) communicate with sensors through RDEP2 and SDEE protocols.

The sensor has the following partitions:

- Application partition—A full IPS system image.
- Maintenance partition—A special purpose IPS image used to reimage the application partition of the IDSM2. When you reimage the maintenance partition, all configuration settings are lost.
- 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.

Cisco IPS 6.1 New Features

Cisco IPS 6.1 contains the following new features:

- Simplified **setup** command—Refinement of the existing **setup** command, presenting the most basic aspects of CLI setup first. You can save and exit or continue with more advanced device setup using the CLI or the Setup Wizard in IDM or IME.
- Setup Wizard—Streamlines the process of setting up Cisco IPS sensors. After you run the basic setup command, you can use the Setup Wizard for more advanced device setup.
- IME—Cisco IPS network monitoring application that performs IPS event viewing and archiving. It also contains the IDM configuration component.
- Performance improvements—Significant optimizations have been made to the startup process.
- Risk category—You can configure risk levels to add to event action overrides. The new category is automatically assigned event actions that span its range.
- Sensor health and network security status—Displays the overall health of the sensor and network security.

- Gadgets—IDM and IME contains gadgets that report various types of information such as sensor and network health, license status, interface status, signature and signature update status.
- Automatic update—Cisco IPS can now automatically download signature and signature engine updates from Cisco.com. When automatic update is enabled, Cisco IPS logs in to Cisco.com and checks for signature and signature engine updates. When an update is available, Cisco IPS downloads the update from Cisco.com and installs it.
- Persistent views—You can set column length, move and hide columns, and sort data in the Signature Definition component of IDM, and those views remain when you exit IDM, and then log in again. The information is stored on your computer.
- Event action grouping—Event actions are now grouped into three categories: Alert and Log, Deny, and Other.
- Real-time alert viewing—You can view alerts in real time in IME, and you can pause, resume, and clear events in the viewer.
- RSS feeds—You can subscribe to RSS channels through IME.

User Interaction

Note The Event Server is now disabled by default. You need to enable Event Server subscriptions only if you are using a third-party event client that is only able to parse IDS 4.x alerts.

You interact with Cisco IPS 6.1 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 6.1 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 6.1 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 6.1 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, CS MARS or another application using SDEE.

Security Features

Cisco IPS 6.1 has the following security features:

- Network access is restricted to hosts who are specifically allowed access.
- All remote hosts who attempt to connect through 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 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.

MainApp

This section describes MainApp, and contains the following topics:

Understanding MainApp

MainApp includes all IPS components except SensorApp and the CLI. It is loaded by the operating system at startup and loads SensorApp. 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

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 Cisco IPS 6.1 MainApp can automatically download signature and signature engine updates from Cisco.com.

- Shut down or reboot the operating system

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 (for example, IDS-4240-K9, WS-SVC-IDSM-2)
- Version of sensor build on the other partition

MainApp also gathers the host statistics and reports the health and security monitoring status.

Event Store

This section describes Event Store, and contains the following topics:

Understanding Event Store

Note The Event Store has a fixed size of 30 MB for all platforms except for AIP SSC-5, which has a fixed size of 10 MB.

Each IPS event is stored in 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. 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 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, IME, and ASDM.

SensorApp and ARC log response actions (TCP resets, IP logging start and stop, blocking start and stop, trigger packet) as status messages.

NotificationApp

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. NotificationApp supports sending sets and gets. The SNMP GETs provide information about basic sensor health.

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

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

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). NotificationApp sends the following information from the evError event:

- Originator information
- Event ID
- Event severity

- Time (UTC and local time)
- Error message

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.

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

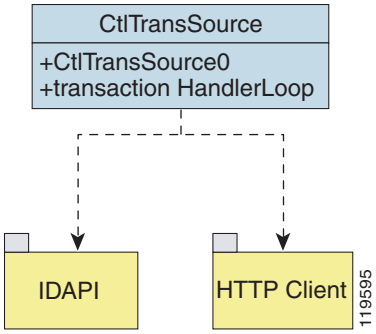
CtlTransSource is an application that forwards locally initiated remote control transactions to their remote destinations using the RDEP and HTTP protocols. CtlTransSource initiates either TLS or non-TLS connections and communicates remote control transactions to HTTP servers over these connections.

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 CtlTransSource. The transactionHandlerLoop method is a loop that waits on remote control transactions that are directed to CtlTransSource.

Figure A-2 shows the transactionHandlerLoop method in the CtlTransSource.

Figure A-2 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 an RDEP control transaction message. The transactionHandlerLoop uses the HttpClient classes to issue the RDEP control transaction request to the HTTP server on the remote node. The remote HTTP server handles the remote control transaction and returns the appropriate RDEP 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 RDEP 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.

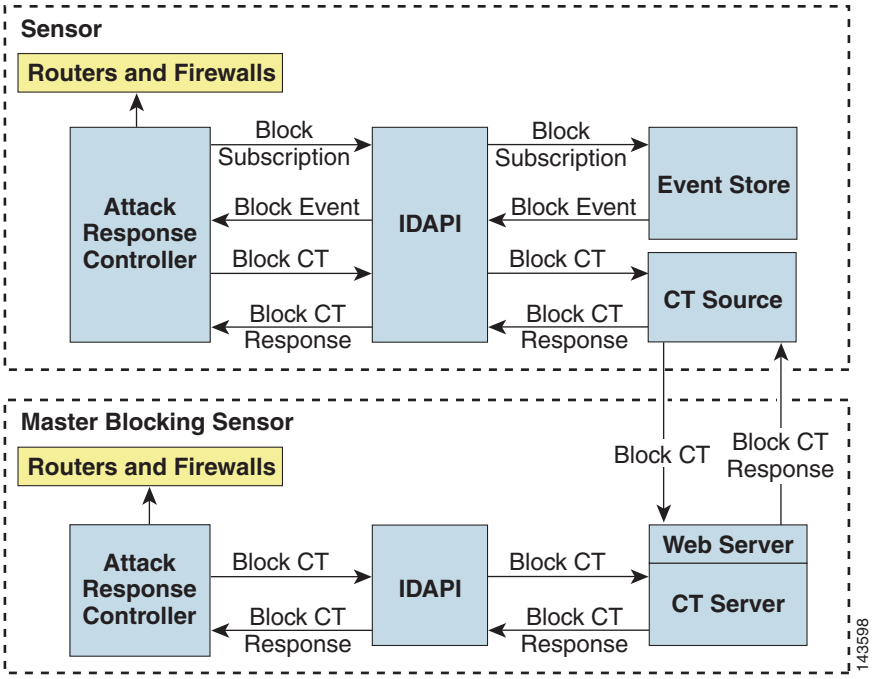
Attack Response Controller

This section describes ARC, and contains the following topics:

Understanding ARC

The main responsibility of 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 ARC. ARC on the master blocking sensor then interacts with the devices it is managing to enable the block. Figure A-3 illustrates ARC.

Figure A-3 ARC



Note An ARC instance can control 0, 1, or many network devices. 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.

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, IME, or ASDM
- A block configured permanently against a host or network address

When you configure ARC to block a device, it initiates either a Telnet or SSH connection with the device. ARC maintains the connection with each device. After the block is initiated, 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

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, 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 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 ARC controls. The firewall device types use a different API to perform blocks and 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

  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. 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

  ARC can block a host or network for a specified number of minutes or indefinitely. ARC determines when a block has expired and unblocks the host or network at that time.

- Logging important events

  ARC writes a confirmation event when block or unblock actions are completed successfully or if any errors occur. 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

  ARC reapplies blocks that have not expired when a shutdown or restart occurs. ARC removes blocks that have expired while it was shut down.

  Note 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

  ARC reapplies blocks and removes expired blocks as needed whenever a network device is shut down and restarted. ARC is not affected by simultaneous or overlapping shutdowns and restarts of ARC.

- Authentication and authorization

  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

  ARC supports host blocks and network blocks. Host blocks are connection based or unconditional. Network blocks are always unconditional.

- NAT addressing

  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

  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 ARC through the CLI or any Cisco IPS manager. When ARC is reenabled, it completely reinitializes itself, including rereading the current configuration for each controlled network device.

  Note We recommend that you disable ARC from blocking when you are configuring any network device, including firewalls.

- Maintains up to 250 active blocks at any given time

  ARC can maintain up to 250 active blocks at a time. Although 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

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 500 series models: ASA 5510, ASA 5520, and ASA 5540
- FWSM

  Note The FWSM cannot block in multi-mode admin context.

ACLs and VACLs

To filter packets on an interface or direction that ARC controls, you can configure 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. 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.

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, ARC writes all blocks and rate limits (with starting timestamps) to a local file (nac.shun.txt) that is maintained by ARC. When 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 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 ARC is not running.

Caution Do not make manual changes to the nac.shun.txt file.

The following scenarios demonstrate how ARC maintains state across restarts.

Scenario 1

There are two blocks in effect when ARC stops and one of them expires before ARC restarts. When 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 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

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, ARC checks for the connectionShun attribute on the host block. If connectionShun is set to true, 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, 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

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*]

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 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.

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, 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. 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 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, IDM, ASDM, and RDEP clients.

The IPS applications use Logger to log messages. Logger sends log messages at any of five levels of severity: debug, timing, warning, error, and fatal. 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), Logger converts the message to an evError event (with the corresponding error severity) and inserts it in Event Store.

Logger receives all syslog messages, except cron messages, that are at the level of informational and above (*.info;cron.none), and inserts them into Event Store as evErrors with the error severity set to Warning. Logger and application logging are controlled through the service logger commands.

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.

InterfaceApp

The InterfaceApp is a subsystem of the MainApp, which is used for configuring and managing the Ethernet interfaces on the IPS device. There are two types of interfaces—management interfaces and sensing interfaces. The management interface is used for managing the IPS device using management applications, such as the IDM, IME, CSM, or CLI.The sensing interfaces represent the packet interfaces, which are used for directing the traffic meant for inspection. In addition to configuration, the InterfaceApp also provides packet statistics for the interfaces.

The InterfaceApp interacts with other applications on the IPS device such as the SensorApp, through control transactions. It also communicates with NIC drivers on each platform to set the interface properties such as speed, duplex, and so forth. The current interface configuration is stored by the InterfaceApp and used when the IPS device is started.

NIC drivers on each platform send asynchronous events called notifications that are related to the state of the Ethernet interfaces, for example, link up and link down notification, to the InterfaceApp. The InterfaceApp collects these notifications and sends the appropriate events.

The InterfaceApp provides a unified view of Ethernet interfaces on different platforms with varied hardware configuration, so that the same set of commands can be used for configuring and managing them.

AuthenticationApp

This section describes AuthenticationApp, and contains the following topics:

Understanding AuthenticationApp

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 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 IDM or 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 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.

AuthenticationApp responds to the execAuthenticateUser control transaction request by attempting to authenticate the identity of the user. 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, 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.

AuthenticationApp uses the underlying operating system to confirm the identity of a user. All the IPS applications send control transactions to 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 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 AuthenticationApp, you can operate sensors at a high level of security.

Web Server

Web Server provides RDEP2 SDEE support, which enables the sensor to report security events, receive IDIOM transactions, and serve IP logs.

Web Server supports HTTP 1.0 and 1.1. Communications with 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 In Cisco IPS 6.1, the RDEP event server service is disabled by default in the Web Server. You receive a warning message that the RDEP event server service is deprecated and will be deleted in a future release. You need to migrate to the SDEE event server. You need to enable RDEP event server subscriptions only if you are using a third-party event client that is only able to parse IDS 4.*x* alerts.

SensorApp

This section describes SensorApp, and contains the following topics:

Understanding SensorApp

SensorApp performs packet capture and analysis. Policy violations are detected through signatures in 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.

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. 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

  Event actions can be associated with an event risk rating threshold that must be surpassed for the actions to take place.

- 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 SRP 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.

Some of the processors call inspectors to perform signature analysis. All inspectors can call the Alarm Channel to produce alerts as needed.

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

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, 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

  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

  Event risk rating helps reduce false positives from the system and gives you more control over what causes an alert.

SensorApp New Features

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 the 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. The 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:

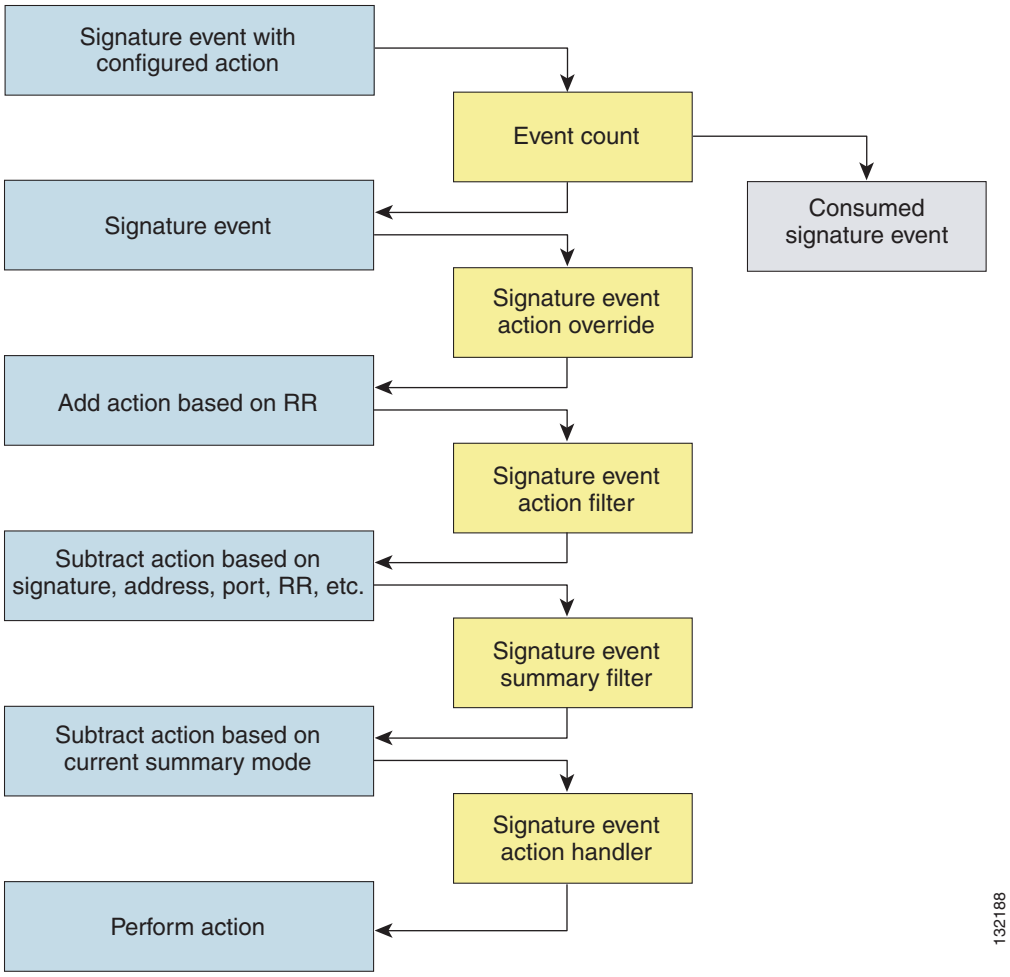
  - Signature ID
  - Subsignature ID
  - Attacker address
  - Attacker port
  - Victim address
  - Victim port
  - Risk rating threshold range
  - Actions to subtract
  - Sequence identifier (optional)
  - Stop-or-continue bit
  - Enable action filter line bit
  - Victim OS relevance or OS relevance

- 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-4 on page A-27 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-4 Signature Event Through Signature Event Action Processor



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 Cisco IPS CLI, and contains the following topics:

- User Roles, page A-28
- Service Account, page A-29

User Roles

There are four user roles:

- Viewers—Can view configuration and events, but cannot modify any configuration data except their user passwords.
- Operators—Can view everything and can modify the following options:
  - Signature tuning (priority, disable or enable)
  - Virtual sensor definition
  - Managed routers
  - Their user passwords
- Administrators—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 IDM. The service user logs in to a bash shell rather than the CLI.

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 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.

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.

Service Account

The service account is a support and troubleshooting tool that enables 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 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 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. 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 Cisco IPS 6.1 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.

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.

Communications

This section describes the communications protocols used by Cisco IPS 6.1, and contains the following topics:

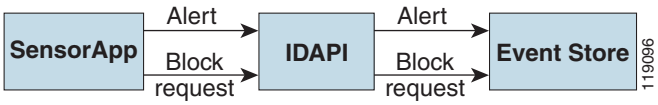
IDAPI

IPS applications use an interprocess communication API called IDAPI to handle internal communications. IDAPI reads and writes event data and provides a mechanism for control transactions. IDAPI is the interface through which all the applications communicate.

SensorApp captures and analyzes the network traffic on its interfaces. When a signature is matched, SensorApp generates an alert, which is stored in the Event Store. If the signature is configured to perform the blocking response action, SensorApp generates a block event, which is also stored in the Event Store.

Figure A-5 illustrates the IDAPI interface.

Figure A-5 IDAPI



Each application registers to the IDAPI to send and receive events and control transactions. 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.

IDAPI provides the necessary synchronization mechanisms to guarantee atomic data accesses.

RDEP2

External communications use RDEP2. RDEP2 is an application-level communications protocol used to exchange IPS event, IP log, configuration, and control messages between IPS clients and IPS servers. RDEP2 communications consist of request and response messages. RDEP2 clients initiate request messages to RDEP2 servers. RDEP2 servers respond to request messages with response messages.

RDEP2 defines three classes of request/response messages: event, IP log, and transaction messages. Event messages include IPS alert, status, and error messages. Clients use IP log requests to retrieve IP log data from servers. Transaction messages are used to configure and control IPS servers.

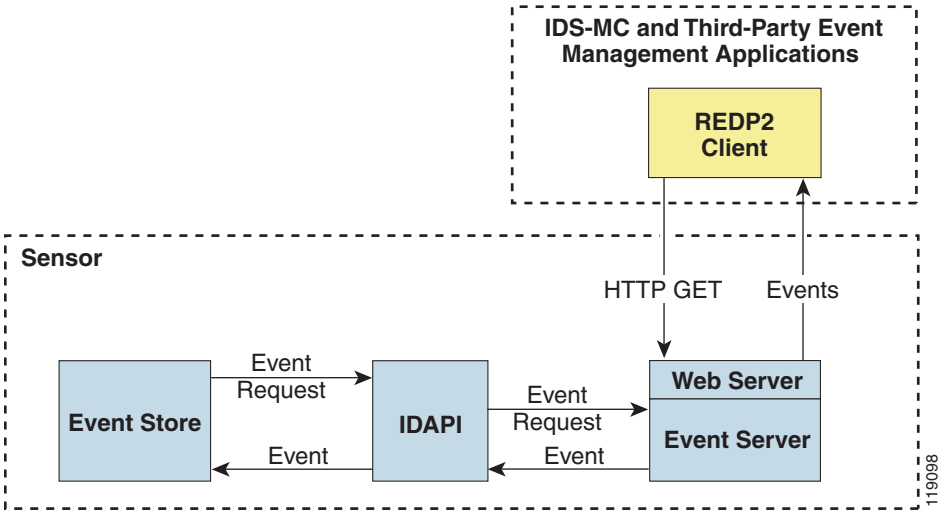
RDEP2 uses the industry standards HTTP, TLS and SSL and XML to provide a standardized interface between RDEP2 agents. The RDEP2 protocol is a subset of the HTTP 1.1 protocol. All RDEP2 messages are legal HTTP 1.1 messages. RDEP2 uses HTTP message formats and message exchange protocol to exchange messages between RDEP2 agents.

You use the IPS manager to specify which hosts are allowed to access the sensor through the network. Sensors accept connections from 1 to 10 RDEP2 clients simultaneously. Clients selectively retrieve data by time range, type of event (alert, error, or status message) and level (alert = high, medium, low, or informational; error = high, medium, low). Events are retrieved by a query (a single bulk get) or subscription (a real-time persistent connection) or both. Communications are secured by TLS or SSL.

Note For retrieving events, the sensor is backwards-compatible to RDEP even though the new standard for retrieval is RDEP2. We recommend you use RDEP2 to retrieve events and send configuration changes for Cisco IPS 6.1.

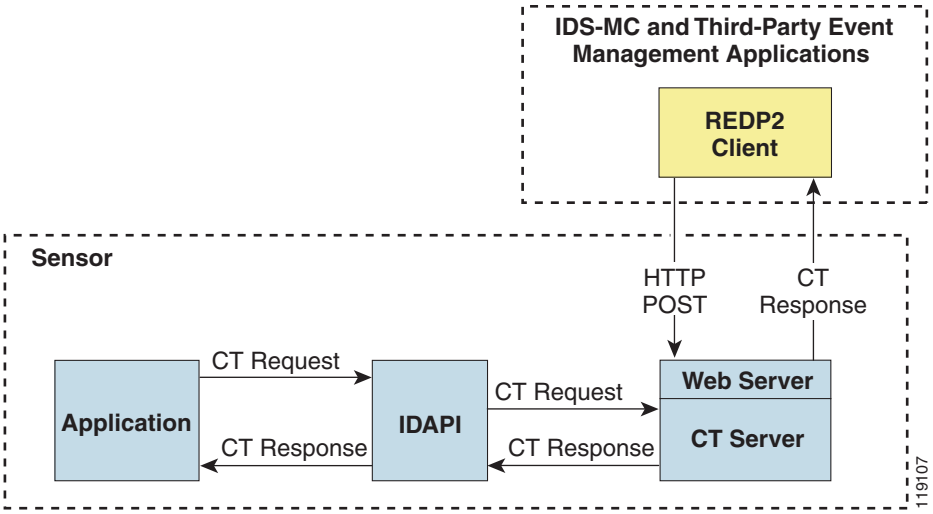
Remote applications retrieve events from the sensor through RDEP2. The remote client sends an RDEP2 event request to the Web Server of the sensor, which passes it to the Event Server. The Event Server queries the Event Store through IDAPI and then returns the result. Figure A-6 shows remote applications retrieving events from the sensor through RDEP2.

Figure A-6 Retrieving Events Through RDEP2



Remote applications send commands to the sensor through RDEP2. The remote client sends an RDEP2 control transaction to the Web Server of the sensor, which passes it to the Control Transaction Server. The Control Transaction Server passes the control transaction through IDAPI to the appropriate application, waits for the response of the application, and then returns the result. Figure A-7 on page A-32 shows remote applications sending commands to the sensor through RDEP2.

Figure A-7 Sending Commands Through RDEP2



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 using the RDEP2 protocol 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

Cisco IPS 6.1 manages its configuration using XML documents. IDCONF specifies the XML schema including Cisco IPS 6.0 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 exchanged over RDEP2 and 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

IPS produces various types of events including intrusion alerts and status events. IPS communicates events to clients such as management applications using the proprietary RDEP2. We have also developed an IPS-industry leading protocol, SDEE, which is a product-independent standard for communicating security device events. SDEE is an enhancement to the current version of RDEP2 that 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.

IPS includes Web Server, which processes HTTP or HTTPS requests. 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 is 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 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 CDIEE 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> …
```

Cisco IPS 6.1 File Structure

Cisco IPS 6.1 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/falcondump—Contains the application for getting packet dumps on the sensing ports of the IDSM2.
- /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 6.1 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.
CLI	Accepts command line input and modifies the local configuration using IDAPI.
SDEE Server[1]	Accepts RDEP2 request for events from remote clients.
MainApp	Reads the configuration and starts applications, handles starting and stopping of applications and node reboots, handles software upgrades.
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	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 a 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.
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.

Table A-2 Summary of Applications (continued)

Application	Description
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.
Control Transaction Server[2]	Accepts control transactions from a remote RDEP2 client, initiates a local control transaction, and returns the response to the remote client.
Control Transaction Source[3]	Waits for control transactions directed to remote applications, forwards the control transactions to the remote node using RDEP2, 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.
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 web server servlet.
3. This is a remote control transaction proxy.

APPENDIX **B**

Signature Engines

This appendix describes the IPS signature engines. It contains the following sections:

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 6.1 engines support a standardized Regex.

Cisco IPS 6.1 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 now combined into two 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.
  - 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—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. Service engine has the following protocol types:
  - DNS—Inspects DNS (TCP and UDP) traffic.
  - FTP—Inspects FTP traffic.
  - Generic—Decodes custom service and payload.
  - Generic Advanced—Analyzes traffic based on the mini-programs that are written to parse the packets.
  - 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.
  - 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 RPC over SMB packets.

Caution 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.

  - SNMP—Inspects SNMP traffic.
  - SSH—Inspects SSH traffic.
  - TNS—Inspects TNS traffic.

- State—Stateful searches of strings in protocols such as SMTP. The state engine now 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.
- 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.

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

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.	*number*
sub-signature-id	Specifies the sub ID of this signature	*number*
alert-severity	Specifies the severity of the alert: • 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.	*sig-name*
alert-notes	Provides additional information about this signature that will be included in the alert message.	*alert-notes*
user-comments	Provides comments about this signature.	*comments*
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.	*release*
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)

Table B-1 Master Engine Parameters (continued)

Parameter	Description	Value
event-count-key	Specifies the storage type on which to count events for this signature: • 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: • 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.	—
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.

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 5-15.
- For more information about passive OS fingerprinting, see Configuring OS Identifications, page 7-23.

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
alert-frequency	Summary options for grouping alerts.	—
summary-mode	Mode used for summarization.	—
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.	—
specify-summary-threshold	(Optional) Enables summary threshold.	yes | no
summary-threshold	Threshold number of alerts to send signature into summary mode.	0 to 65535
specify-global-summary-threshold	Enable global summary threshold.	yes | no

Table B-2 Master Engine Alert Frequency Parameters (continued)

Parameter	Description	Value
global-summary-threshold	Threshold number of events to take alerts into global summary.	1 to 65535
summary-interval	Time in seconds used in each summary alert.	1 to 1000
summary-key	The storage type on which to summarize this signature: • 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

Note Most of the following event actions belong to each signature engine unless they are not appropriate for that particular engine.

The following event action parameters belong to each signature engine (if it makes sense for that signature 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 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 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
  - request-block-connection—Requests ARC to block this connection.
  - request-block-host—Requests ARC to block this attacker host.
  - request-rate-limit—Requests ARC to perform rate limiting.
  - reset-tcp-connection—Sends TCP resets to hijack and terminate the TCP flow.

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:

- droppedPacket
- deniedFlow
- tcpOneWayResetSent

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.

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 *X* times.
{x,}	Minimum quantifier	Repeat at least *X* times.
.	Dot	Any one character except new line (0x0A).
[abc]	Character class	Any character listed.
[^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 *char* is a metacharacter or not, matches the literal *char*.
char	Character	When char is not a metacharacter, matches the literal char.
\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, bananananananas	ba(na)+s
foo and bar on the same line with anything except a new line between them	foo.*bar
Either foo or bar	foo|bar
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-10
- AIC Engine and Sensor Performance, page B-10
- 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 engine runs 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 engine defines signatures for deep inspection of web traffic. It also defines signatures that authorize and enforce FTP commands. There are two AIC engines: AIC HTTP and AIC FTP.

The AIC engine has 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.
content-types	AIC signature that deals with MIME types: • 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 content types recognized by the sensor.

Table B-5 AIC HTTP Engine Parameters (continued)

Parameter	Description
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	Maximum allowed HTTP requests per connection (1 to 16).
msg-body-pattern	Uses Regex to define signatures that look for specific patterns in the message body.
request-methods	AIC signature that allows actions to be associated with HTTP request methods: • define-request-method—get, put, and so forth. • recognized-request-methods—Lists methods recognized by the sensor.
transfer-encodings	AIC signature that deals with transfer encodings: • 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
signature-type	Specifies the type of AIC signature.
ftp-commands	Associates an action with an FTP command: • ftp-command—Lets you choose the FTP command you want to inspect.
unrecognized-ftp-command	Inspects unrecognized FTP commands.

For More Information

- For the procedures for configuring AIC engine signatures, see Configuring AIC Signatures, page 8-17.
- For an example of a custom AIC signature, see Creating an AIC Signature, page 8-26.
- For a list of the signature regular expression syntax, see Regular Expression Syntax, page B-9.

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

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
specify-mac-flip	Fires an alert when the MAC address changes more than this many times for this IP address.
specify-type-of-arp-sig	Specifies the type of ARP signatures you want to fire on: • Source Broadcast (default)—Fires an alert for this signature when it sees an ARP source address of 255.255.255.255. • 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 Multicast—Fires an alert for this signature when it sees an ARP source MAC address of 01:00:5e:(00-7f).
specify-request-inbalance	Fires an alert when there are this many more requests than replies on the IP address.
specify-arp-operation	The ARP operation code for this signature.

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.

Note 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-8 lists the parameters that are specific to the Atomic IP engine.

Table B-8 Atomic IP Engine Parameters

Parameter	Description
fragment-status	Specifies whether or not fragments are wanted.
specify-l4-protocol	Specifies Layer 4 protocol.
specify-ip-payload-length	Specifies IP datagram payload length.
specify-ip-header-length	Specifies IP datagram header length.
specify-ip-tos	Specifies type of service.
specify-ip-ttl	Specifies time to live.
specify-ip-version	Specifies IP protocol version.
specify-ip-id	Specifies IP identifier.
specify-ip-total-length	Specifies IP datagram total length.
specify-ip-option-inspection	Specifies IP options inspection.
specify-ip-addr-options	Specifies IP addresses.
swap-attacker-victim	True if attacker and victim addresses and ports (source and destination) are swapped in the alert message and actions. False for no swapping (default).

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.

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.

Table B-9 lists the Atomic IPv6 signatures.

Table B-9 Atomic IPv6 Signatures

Signature ID	Subsignature ID	Name	Description
1600	0	ICMPv6 zero length option	For any option type that has ZERO stated as its length.
1601	0	ICMPv6 option type 1 violation	Violation of the valid length of 8 or 16 bytes.
1602	0	ICMPv6 option type 2 violation	Violation of the valid length of 8 or 16 bytes.
1603	0	ICMPv6 option type 3 violation	Violation of the valid length of 32 bytes.
1604	0	ICMPv6 option type 4 violation	Violation of the valid length of 80 bytes.
1605	0	ICMPv6 option type 5 violation	Violation of the valid length of 8 bytes.
1606	0	ICMPv6 short option data	Not enough data signature (when the packet states there is more data for an option than is available in the real packet).
1607	0	IPv6 multiple-crafted fragment packets	Produces an alert when more than one first fragment is seen in a 30-second period.

Fixed Engine

This section describes the Fixed engine, and contains the following topics:

Understanding the Fixed Engine

The Fixed engine combines 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 Fixed TCP and Fixed UDP use the service-ports parameter as exclusion ports. Fixed ICMP uses the service-ports parameter as excluded ICMP types.

Fixed ICMP Engine

Table B-10 lists the parameters specific to the Fixed ICMP engine.

Table B-10 Fixed ICMP Engine Parameters

Parameter	Description	Value
direction	Direction of traffic: • 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.	string
specify-exact-match-offset	(Optional) Enables exact match offset: • exact-match-offset—The exact stream offset the regex-string must report for a match to be valid.	0 to 65535
specify-min-match-length	(Optional) Enables minimum match length: • min-match-length—Specifies the minimum number of bytes the regex-string must match.	0 to 65535
specify-icmp-type	(Optional) Enables inspection of the ICMP header type: • icmp-type—Specifies the ICMP header TYPE value.	0 to 65535
swap-attacker-victim	True if attacker and victim addresses and ports (source and destination) are swapped in the alert message and actions. False for no swapping (default).	true | false

For More Information

For a list of the signature regular expression syntax, see Regular Expression Syntax, page B-9.

Fixed TCP Engine

Table B-11 lists the parameters specific to the Fixed TCP engine.

Table B-11 Fixed TCP Engine Parameters

Parameter	Description	Value
direction	Direction of traffic: • 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.	string
specify-exact-match-offset	(Optional) Enables exact match offset: • exact-match-offset—The exact stream offset the regex-string must report for a match to be valid.	0 to 65535
specify-min-match-length	(Optional) Enables minimum match length: • min-match-length—Specifies the minimum number of bytes the regex-string must match.	0 to 65535
specify-service-ports	Enables service ports for use: • service-ports—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	True if attacker and victim addresses and ports (source and destination) are swapped in the alert message and actions. False for no swapping (default).	true | false

1. The second number in the range must be greater than or equal to the first number.

For More Information

For a list of the signature regular expression syntax, see Regular Expression Syntax, page B-9.

Fixed UDP Engine

Table B-12 lists the parameters specific to the Fixed UDP engine.

Table B-12 Fixed UDP Engine Parameters

Parameter	Description	Value
direction	Direction of traffic: • 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.	string
specify-exact-match-offset	(Optional) Enables exact match offset: • exact-match-offset—The exact stream offset the regex-string must report for a match to be valid.	0 to 65535
specify-min-match-length	(Optional) Enables minimum match length: • min-match-length—Specifies the minimum number of bytes the regex-string must match.	0 to 65535
specify-service-ports	Enables service ports for use: • service-ports—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	True if attacker and victim addresses and ports (source and destination) are swapped in the alert message and actions. False for no swapping (default).	true | false

1. The second number in the range must be greater than or equal to the first number.

For More Information

For a list of the signature regular expression syntax, see Regular Expression Syntax, page B-9.

Flood Engine

The Flood engine defines 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 types of 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	Which kind of traffic to inspect.	ICMP UDP
rate	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[3] a-b[,c-d]

1. An alert fires when the rate is greater than the packets per second.
2. The second number in the range must be greater than or equal to the first number.
3. 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	Gap of time allowed (in seconds) for a flood signature.	0 to 65535
peaks	Number of allowed peaks of flood traffic.	0 to 65535
protocol	Which kind of traffic to inspect.	ICMP TCP UDP
rate	Threshold number of packets per second.	0 to 65535[1]
sampling-interval	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.

Meta Engine

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.

Caution A large number of Meta signatures could adversely affect overall sensor performance.

Table B-15 lists the parameters specific to the Meta engine.

Table B-15 Meta Engine Parameters

Parameter	Description	Value
swap-attacker-victim	True if attacker and victim addresses and ports (source and destination) are swapped in the alert message and actions. False for no swapping (default).	true | false
meta-reset-interval	Time in seconds to reset the Meta signature.	0 to 3600
component-list	List of Meta components: • edit—Edits an existing entry • insert—Inserts a new entry into 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 • move—Moves an entry in the list	*name1*
meta-key	Storage type for the Meta signature: • Attacker address • Attacker and victim addresses • Attacker and victim addresses and ports • Victim address	AaBb AxBx Axxx xxBx
unique-victim-ports	Number of unique victims ports required per Meta signature.	1 to 256
component-list-in-order	Whether to fire the component list in order.	true | false

For More Information

For an example of a custom Meta engine signature, see Example Meta Signature, page 8-47.

Multi String Engine

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	Length of stream or packet that must contain all offending strings for the signature to fire.	0 to 4294967295
protocol	Layer 4 protocol selection.	icmp tcp udp
regex-component	List of regex components: • regex-string—The string to search for. • spacing-type—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	Type of TCP or UDP port to inspect: • 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]
exact-spacing	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	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	True if attacker and victim addresses and ports (source and destination) are swapped in the alert message and actions. False for no swapping (default).	true | false

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.

Caution The Multi String engine can have a significant impact on memory usage.

For More Information

For a list of the signature regular expression syntax, see Regular Expression Syntax, page B-9.

Normalizer Engine

The Normalizer engine deals with IP fragmentation and TCP normalization. This section describes the Normalizer engine, and contains the following topics:

Understanding the 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, 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 looks for any abnormal packets associated with evasion and attacks.

AIP SSM and the Normalizer Engine

The majority of the features in the Normalizer engine are not used on the AIP SSM, 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 the procedures for configuring signatures in the Normalizer engine, see Configuring IP Fragment Reassembly, page 8-28, and Configuring TCP Stream Reassembly, page 8-31.

Normalizer Engine Parameters

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.
specify-max-fragments-per-dgram	(Optional) Enables maximum fragments per datagram.
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.
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.

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.

This section describes the Service engines, and contains the following topics:

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	Protocol of interest for this inspector.	tcp udp
specify-query-chaos-string	(Optional) Enables the DNS Query Class Chaos String.	*query-chaos-string*
specify-query-class	(Optional) Enables the query class: • query-class—DNS Query Class 2 Byte Value	0 to 65535
specify-query-invalid-domain-name	(Optional) Enables query invalid domain name: • query-invalid-domain-name—DNS Query Length greater than 255	true | false
specify-query-jump-count-exceeded	(Optional) Enables query jump count exceeded: • query-jump-count-exceeded—DNS compression counter	true | false
specify-query-opcode	(Optional) Enables query opcode: • query-opcode—DNS Query Opcode 1 byte Value	0 to 65535

Table B-18 Service DNS Engine Parameters (continued)

Parameter	Description	Value
specify-query-record-data-invalid	(Optional) Enables query record data invalid: • query-record-data-invalid—DNS Record Data incomplete	true | false
specify-query-record-data-len	(Optional) Enables the query record data length: • query-record-data-len—DNS Response Record Data Length	0 to 65535
specify-query-src-port-53	(Optional) Enables the query source port 53: • query-src-port-53—DNS packet source port 53	true | false
specify-query-stream-len	(Optional) Enables the query stream length: • query-stream-len—DNS Packet Length	0 to 65535
specify-query-type	(Optional) Enables the query type: • query-type—DNS Query Type 2 Byte Value	0 to 65535
specify-query-value	(Optional) Enables the query value: • query-value—Query 0 Response 1	true | false

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	Direction of traffic: • Traffic from service port destined to client port • Traffic from client port destined to service port	from-service to-service
ftp-inspection-type	Type of inspection to perform: • 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 pasvI
service-ports	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	True if attacker and victim addresses and ports (source and destination) are swapped in the alert message and actions. False for no swapping (default).	true | false

1. The second number in the range must be greater than or equal to the first number.

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	(Optional) Enables the destination port: • dst-port—Destination port of interest for this signature	0 to 65535
specify-ip-protocol	(Optional) Enables IP protocol: • ip-protocol—The IP protocol this inspector should examine	0 to 255
specify-payload-source	(Optional) Enables payload source inspection: • payload-source—Payload source inspection for the following types: – 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-dataI
specify-src-port	(Optional) Enables the source port: • src-port—Source port of interest for this signature	0 to 65535
specify-regex-string	The regular expression to look for when the policy type is regex: • A regular expression to search for in a single TCP packet • (Optional) Enables min match length for use. The minimum length of the Regex match required to constitute a match.	regex-string specify-min-match-length
swap-attacker-victim	True if attacker and victim addresses and ports (source and destination) are swapped in the alert message and actions. False for no swapping (default).	true | false

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	Type of H225 message to which the signature applies: • SETUP • ASN.1-PER • Q.931 • TPKT	asn.1-per q.931 setup tpkt
policy-type	Type of H225 policy to which the signature applies: • 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. Regex and presence are not valid for TPKT signatures.	length presence regex validate value
specify-field-name	(Optional) Enables field name for use. Only valid for SETUP and Q.931 message types. Gives a dotted representation of the field name that this signature applies to. • field-name—Field name to inspect.	1 to 512
specify-invalid-packet-index	(Optional) Enables invalid packet index for use for specific errors in ASN, TPKT, and other errors that have fixed mapping. • invalid-packet-index—Inspection for invalid packet index.	0 to 255
specify-regex-string	The regular expression to look for when the policy type is regex. This is never set for TPKT signatures: • A regular expression to search for in a single TCP packet • (Optional) Enables min match length for use. The minimum length of the Regex match required to constitute a match. This is never set for TPKT signatures.	regex-string specify-min-match-length

Table B-21 Service H.225 Engine Parameters (continued)

Parameter	Description	Value
specify-value-range	Valid for the length or value policy types (0x00 to 6535). Not valid for other policy types. • value-range—Range of values.	0 to 65535[1] a-b
swap-attacker-victim	True if attacker and victim addresses and ports (source and destination) are swapped in the alert message and actions. False for no swapping (default).	true | false

1. The second number in the range must be greater than or equal to the first number.

For More Information

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 to web services 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	Maximum field sizes grouping.	—
specify-max-arg-field-length	(Optional) Enables maximum argument field length: • max-arg-field-length—Maximum length of the arguments field.	0 to 65535

Table B-22 Service HTTP Engine Parameters (continued)

Parameter	Description	Value
specify-max-header-field-length	(Optional) Enables maximum header field length: • max-header-field-length—Maximum length of the header field.	0 to 65535
specify-max-request-length	(Optional) Enables maximum request field length: • max-request-length—Maximum length of the request field.	0 to 65535
specify-max-uri-field-length	(Optional) Enables the maximum URI field length: • max-uri-field-length—Maximum length of the URI field.	0 to 65535
regex	Regular expression grouping.	—
specify-arg-name-regex	(Optional) Enables searching the Arguments field for a specific regular expression: • arg-name-regex—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	(Optional) Enables searching the Header field for a specific regular expression: • header-regex—Regular Expression to search in the HTTP Header field. The Header is defined after the first CRLF and continues until CRLFCRLF.	—
specify-request-regex	(Optional) Enables searching the Request field for a specific regular expression: • request-regex—Regular expression to search in both HTTP URI and HTTP Argument fields. • specify-min-request-match-length—Enables setting a minimum request match length.	0 to 65535
specify-uri-regex	(Optional) Regular expression to search in HTTP URI field. The URI field is defined to be after the HTTP method (GET, for example) and before the first CRLF. The regular expression is protected, which means you cannot change the value.	[/\\][a-zA-Z][a-zA-Z][a-zA-Z][a-zA-Z][a-zA-Z][a-zA-Z][a-zA-Z][a-zA-Z][.]jpeg
service-ports	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	True if attacker and victim addresses and ports (source and destination) are swapped in the alert message and actions. False for no swapping (default).	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 Service HTTP custom signature, see Example Service HTTP Signature, page 8-44.
- 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	Type of inspection to perform: • 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.	—
service-ports	A comma-separated list of ports or port ranges where the target service resides.	0 to 65535[1] a-b[,c-d]
direction	Direction of the traffic: • 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.

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 establish the channel and pass 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	Protocol of interest for this inspector: • type—UDP or TCP	tcp udp
specify-flags	Flags to set: • msrpc-flags • msrpc-tcp-flags-mask	concurrent-execution did-not-execute first-fragment last-fragment maybe-semantics object-uuid reserved
specify-operation	(Optional) Enables using MSRPC operation: • operation—MSRPC operation requested. Required for SMB_COM_TRANSACTION commands. Exact match.	0 to 65535
swap-attacker-victim	True if address (and ports) source and destination are swapped in the alert message. False for no swap (default).	true | false
specify-regex-string	(Optional) Enables using a regular expression string: • specify-exact-match-offset—Enables the exact match offset: – exact-match-offset—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: – min-match-length—Minimum number of bytes the regular expression string must match.	0 to 65535
specify-uuid	(Optional) Enables UUID: • uuid—MSRPC UUID field	000001a000000000c00 0000000000046

For More Information

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	Whether or not a password was used in an MS SQL login.	true | false
specify-sql-username	(Optional) Enables using an SQL username: • sql-username—Username (exact match) of user logging in to MS SQL service.	sa

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	Type of inspection to perform.	
inspect-ntp-packets	Inspects NTP packets: • control-opcode—Opcode number of an NTP control packet according to RFC1305, Appendix B. • max-control-data-size—Maximum allowed amount of data sent in a control packet. • mode—Mode of operation of the NTP packet per RFC 1305.	0 to 65535
is-invalid-data-packet	Looks for invalid NTP data packets. Checks the structure of the NTP data packet to make sure it is the correct size.	true | false
is-non-ntp-traffic	Checks for nonNTP packets on an NTP port.	true | false

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 a list of the Master engine parameters, see Master Engine, page B-3.

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	Direction of traffic: • Traffic from service port destined to client port. • Traffic from client port destined to service port.	from-service to-service
protocol	Protocol of interest.	tcp udp
service-ports	A comma-separated list of ports or port ranges where the target service resides.	0 to 65535[1] a-b[,c-d]
specify-regex-string	(Optional) Enables using a regular expression string: • specify-exact-match-offset—Enables the exact match offset: – exact-match-offset—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: – min-match-length—Minimum number of bytes the regular expression string must match.	0 to 65535
specify-is-spoof-src	(Optional) Enables the spoof source address: • is-spoof-src—Fires an alert when the source address is 127.0.0.1.	true | false
specify-port-map-program	(Optional) Enables the portmapper program: • port-map-program—The program number sent to the portmapper for this signature.	0 to 9999999999

Table B-27 Service RPC Engine Parameters (continued)

Parameter	Description	Value
specify-rpc-max-length	(Optional) Enables RPC maximum length: • rpc-max-length—Maximum allowed length of the entire RPC message. Lengths longer than what you specify fire an alert.	0 to 65535
specify-rpc-procedure	(Optional) Enables RPC procedure: • rpc-procedure—RPC procedure number for this signature.	0 to 1000000
specify-rpc-program	(Optional) Enables RPC program: • rpc-program—RPC program number for this signature.	0 to 1000000
swap-attacker-victim	True if attacker and victim addresses and ports (source and destination) are swapped in the alert message and actions. False for no swapping (default).	true | false

1. The second number in the range must be greater than or equal to the first number.

For More Information

For a list of the signature regular expression syntax, see Regular Expression Syntax, page B-9.

Service SMB Advanced Engine

Caution 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	A comma-separated list of ports or port ranges where the target service resides.	0 to 65535 a-b[,c-d] [1]
specify-command	(Optional) Enables SMB commands: • command—SMB command value; exact match required; defines the SMB packet type.[2]	0 to 255

Table B-28 Service SMB Advanced Engine Parameters (continued)

Parameter	Description	Value
specify-direction	(Optional) Enables traffic direction: • direction—Lets you specify the direction of traffic: – 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-operation	(Optional) Enables MSRPC over SMB: • msrpc-over-smb-operation—Required for SMB_COM_TRANSACTION commands, exact match required.	0 to 65535
specify-regex-string	(Optional) Enables searching for regex strings: • regex-string—A regular expression to search for in a single TCP packet.	
specify-exact-match-offset	(Optional) Enables exact match offset: • exact-match-offset—The exact stream offset the Regex string must report a match to be valid.	
specify-min-match-length	(Optional) Enables minimum match length: • min-match-length—Minimum number of bytes the Regex string must match.	
specify-payload-source	(Optional) Enables payload source: • payload-source—Payload source inspection.[3]	
specify-scan-interval	(Optional) Enables scan interval: • scan-interval—The interval in seconds used to calculate alert rates.	1 to 131071
specify-tcp-flags	(Optional) Enables TCP flags: • msrpc-tcp-flags • msrpc-tcp-flags-mask	• concurrent execution • did not execute • first fragment • last fragment • maybe • object UUID • pending cancel • reserved

Table B-28 Service SMB Advanced Engine Parameters (continued)

Parameter	Description	Value
specify-type	(Optional) Enables type of MSRPC over SMB packet: • type—Type field of MSRPC over SMB packet	• 0 = Request • 2 = Response • 11 = Bind • 12 = Bind Ack
specify-uuid	(Optional) Enables MSRPC over UUID: • uuid—MSRPC UUID field	32-character string composed of hexadecimal characters 0-9, a-f, A-F.
specify-hit-count	(Optional) Enables hit counting: • hit-count—The threshold number of occurrences in scan-interval to fire alerts.	1 to 65535
swap-attacker-victim	True if attacker and victim addresses and ports (source and destination) are swapped in the alert message and actions. False for no swapping (default).	true | false

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 \x26amp; 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 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	Type of inspection to perform.	—
brute-force-inspection	Inspects for brute force attempts: • brute-force-count—The number of unique SNMP community names that constitute a brute force attempt.	0 to 65535

Table B-29 Service SNMP Engine Parameters (continued)

Parameter	Description	Value
invalid-packet-inspection	Inspects for SNMP protocol violations.	—
non-snmp-traffic-inspection	Inspects for non-SNMP traffic destined for UDP port 161.	—
snmp-inspection	Inspects SNMP traffic: • specify-community-name [yes | no]: – community-name—Searches for the SNMP community name, that is, the SNMP password. • specify-object-id [yes | no]: – object-id—Searches for the SNMP object identifier.	*community-name* *object-id*

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 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
SSH Version		
length-type	Inspects for one of the following SSH length types: • key-length—Length of the SSH key to inspect for: – length—Keys larger than this fire the RSAREF overflow. • user-length—User length SSH inspection: – length—Keys larger than this fire the RSAREF overflow.	0 to 65535
service-ports	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	(Optional) Enables packet depth: • packet-depth—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.

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	Direction of traffic: • Traffic from service port destined to client port • Traffic from client port destined to service port	from-service to-service
type	Specifies the TNS frame value type: • 1—Connect • 2—Accept • 4—Refuse • 5—Redirect • 6—Data • 11—Resend • 12—Marker	1 2 4 5 6 11 12
specify-regex-string	(Optional) Enables using a regular expression string: • specify-exact-match-offset—Enables the exact match offset: – exact-match-offset—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: – min-match-length—Minimum number of bytes the regular expression string must match.	0 to 65535
specify-regex-payload-src	Specifies which protocol to inspect: payload-src: • 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 tns

For More Information

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 something 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 alert. 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	State machine grouping.	• smpt • lpr-format-string • cisco-login
cisco-login	Specifies the state machine for Cisco login: • state-name—Name of the state required before the signature fires an alert: – 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: • state-name—Name of the state required before the signature fires an alert: – Abort state to end LPR Format String inspection – Format character state – State state	• abort • format-char • start
state-name	Specifies the state machine for the SMTP protocol: • state-name—Name of the state required before the signature fires an alert: – 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

Table B-32 State Engine Parameters (continued)

Parameter	Description	Value
direction	Direction of the traffic: • Traffic from service port destined to client port. • Traffic from client port destined to service port.	from-service to-service
service-ports	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	(Optional) Enables exact match offset: • exact-match-offset—The exact stream offset the regular expression string must report for a match to be valid.	0 to 65535
specify-max-match-offset	(Optional) Enables maximum match offset: • max-match-offset—The maximum stream offset the regular expression string must report for a match to be valid.	0 to 65535
specify-min-match-offset	(Optional) Enables minimum match offset: • min-match-offset—The minimum stream offset the regular expression string must report for a match to be valid.	0 to 65535
specify-min-match-length	(Optional) Enables minimum match length: • min-match-length—Minimum number of bytes the regular expression string must match.	0 to 65535
swap-attacker-victim	True if attacker and victim addresses and ports (source and destination) are swapped in the alert message and actions. False for no swapping (default).	true| false

1. The second number in the range must be greater than or equal to the first number.

String Engines

This section describes the String engine, and contains the following topics:

Understanding 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.

String ICMP Engine Parameters

Table B-33 lists the parameters specific to the String ICMP engine.

Table B-33 String ICMP Engine Parameters

Parameter	Description	Value
direction	Direction of the traffic: • Traffic from service port destined to client port. • Traffic from client port destined to service port.	from-service to-service
icmp-type	ICMP header TYPE value.	0 to 18[1] a-b[,c-d]
specify-exact-match-offset	(Optional) Enables exact match offset: • exact-match-offset—The exact stream offset the regular expression string must report for a match to be valid.	0 to 65535
specify-min-match-length	(Optional) Enables minimum match length: • min-match-length—Minimum number of bytes the regular expression string must match.	0 to 65535
swap-attacker-victim	True if attacker and victim addresses and ports (source and destination) are swapped in the alert message and actions. False for no swapping (default).	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 Signature, page 8-41.

String TCP Engine Parameters

Table B-34 lists the parameters specific to the String TCP engine.

Table B-34 String TCP Engine

Parameter	Description	Value
direction	Direction of the traffic: • Traffic from service port destined to client port. • Traffic from client port destined to service port.	from-service to-service
service-ports	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	(Optional) Enables exact match offset: • exact-match-offset—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	(Optional) Enables minimum match length: • min-match-length—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	True if attacker and victim addresses and ports (source and destination) are swapped in the alert message and actions. False for no swapping (default).	true | false

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.

For More Information

For an example custom String engine signature, see Example String TCP Signature, page 8-41.

String UDP Engine Parameters

Table B-35 lists the parameters specific to the String UDP engine.

Table B-35 String UDP Engine

Parameter	Description	Value
direction	Direction of the traffic: • Traffic from service port destined to client port. • Traffic from client port destined to service port.	from-service to-service
service-ports	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	(Optional) Enables exact match offset: • exact-match-offset—The exact stream offset the regular expression string must report for a match to be valid.	0 to 65535
specify-min-match-length	(Optional) Enables minimum match length: • min-match-length—Minimum number of bytes the regular expression string must match.	0 to 65535
swap-attacker-victim	True if attacker and victim addresses and ports (source and destination) are swapped in the alert message and actions. False for no swapping (default).	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 Signature, page 8-41.

Sweep Engines

This section describes the Sweep engines, and contains the following topics:

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. The ICMP sweeps must have an ICMP type specified to discriminate among the various types of ICMP packets.

Data Node

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-36 lists the parameters specific to the Sweep engine.

Table B-36 Sweep Engine Parameters

Parameter	Description	Value
dst-addr-filter	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	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	Protocol of interest for this inspector.	• icmp • udp • tcp
specify-icmp-type	(Optional) Enables inspection of the ICMP header type: • icmp-type—Specifies the ICMP header TYPE value.	0 to 255
specify-port-range	(Optional) Enables using a port range for inspection: • port-range—UDP port range used in inspection.	0 to 65535 a-b[,c-d]
fragment-status	Specifies whether fragments are wanted or not: • 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
mask	Mask used in TCP flags comparison: • URG bit • ACK bit • PSH bit • RST bit • SYN bit • FIN bit	• urg • ack • psh • rst • syn • fin
storage-key	Type of address key used to store persistent data: • 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

Table B-36 Sweep Engine Parameters (continued)

Parameter	Description	Value
swap-attacker-victim	True if attacker and victim addresses and ports (source and destination) are swapped in the alert message and actions. False for no swapping (default).	true | false
tcp-flags	TCP flags to match when masked by mask: • URG bit • ACK bit • PSH bit • RST bit • SYN bit • FIN bit	• urg • ack • psh • rst • syn • fin
unique	Threshold number of unique port connections between the two hosts.	0 to 65535

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-37 lists the parameters specific to the Sweep Other TCP engine.

Table B-37 Sweep Other TCP Engine Parameters

Parameter	Description	Value
specify-port-range	(Optional) Enables using a port range for inspection: • port-range—UDP port range used in inspection.	0 to 65535 a-b[,c-d]
set-tcp-flags	Lets you set TCP flags to match. • tcp-flags—TCP flags used in this inspection: – URG bit – ACK bit – PSH bit – RST bit – SYN bit – FIN bit	• urg • ack • psh • rst • syn • fin

Traffic Anomaly Engine

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—(Inline only) Does not transmit this packet and future packets originating from the attacker address for a specified period of time.
- Log attacker pairs—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).

Note You can edit or tune anomaly detection signatures but you cannot create custom anomaly detection signatures.

Table 38 lists the anomaly detection worm signatures.

Table 38 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 38 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.
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.

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-39 lists the parameters specific to the Traffic ICMP engine.

Table B-39 Traffic ICMP Engine Parameters

Parameter	Description	Value
parameter-tunable-sig	Whether this signature has configurable parameters.	yes | no
inspection-type	Type of inspection to perform: • Inspects for original LOKI traffic. • Inspects for modified LOKI traffic.	is-loki is-mod-loki
reply-ratio	Inbalance 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

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.

APPENDIX C

Troubleshooting

This appendix contains troubleshooting tips and procedures for sensors and software. It contains the following sections:

Bug ToolKit

For the most complete and up-to-date list of caveats, use the Bug Toolkit to refer to the caveat release note. You can use the Bug Toolkit to search for known bugs based on software version, feature set, and keywords. The resulting matrix shows when each bug was integrated, or fixed if applicable. It also lets you save the results of a search in Bug Groups, and also create persistent Alert Agents that can feed those groups with new defect alerts.

Note You must be logged in to Cisco.com to access the Bug Toolkit.

If you are a registered Cisco.com user, you can view the Bug Toolkit at this URL:

http://tools.cisco.com/Support/BugToolKit/action.do?hdnAction=searchBugs

To become a registered cisco.com user, go to this URL:

http://tools.cisco.com/RPF/register/register.do

Preventive Maintenance

This section describes how to perform preventive maintenance for your sensor, and contains the following topics:

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-3.
- 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.

```
sensor# copy current-config backup-config
```

The current configuration is saved in a backup file.

Step 3 Display the backup configuration file.

```
sensor# more backup-config
```

The backup configuration file is displayed.

Step 4 You can either merge the backup configuration with the current configuration, or you can overwrite the current configuration:

- To merge the backup configuration into the current configuration:

  ```
  sensor# copy backup-config current-config
  ```

- To 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.

Options

The following options 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
- 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 also 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 HTTP and HTTPS prompt for a password if a username is required to access the website. If you use HTTPS protocol, 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 22-2.

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.

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.

Note 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 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.

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.

```
sensor(config)# user username privilege service
```

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.

Step 4 Specify a password when prompted.

The password must conform to the requirements set by the sensor administrator. 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 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 reimaged to guarantee proper operation.
*****************************************************************************************
```

Disaster Recovery

This section provides recommendations and steps to take if you need to recover your sensor after a disaster.

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-3.
- For the procedure for obtaining a list of the current users on the sensor, see Showing User Status, page 4-17.
- For the procedures for reimaging a sensor, see Chapter 22, "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 21-1.
- 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 4-42.
- For the procedure for adding users, see Adding and Removing Users, page 4-12.

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

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.

Note Administrators may need to disable the password recovery feature for security reasons.

Table C-1 lists the password recovery methods according to platform.

Table C-1 Password Recovery Methods According to Platform

Platform	Description	Recovery Method
4200 series sensors	Standalone IPS appliances	GRUB prompt or ROMMON
AIM IPS NME IPS	Router IPS modules	Bootloader command
AIP SSM	ASA 5500 series adaptive security appliance modules	ASA CLI command
IDSM2	Switch IPS module	Password recovery image file

Password Recovery for Appliances

This section describes the two ways to recover the password for appliances. It contains the following topics:

Using the GRUB Menu

For 4200 series 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.

Note You must have a terminal server or direct serial connection to the appliance to use the GRUB menu to recover the password.

To recover the password on appliances, follow these steps:

Step 1 Reboot the appliance.

The following menu appears:

```
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**. You can change the password the next time you log in to the CLI.

Using ROMMON

For the IPS 4240 and the IPS 4255 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.

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-4240-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
```

Password Recovery for the AIM IPS

To recover the password for the AIM IPS, use the **clear password** command. You must have console access to the AIM IPS and administrative access to the router.

To recover the password for the AIM IPS, follow these steps:

Step 1 Log in to the router.

Step 2 Enter privileged EXEC mode on the router.

```
router> enable
```

Step 3 Confirm the module slot number in your router.

```
router# show run | include ids-sensor
interface IDS-Sensor0/0
router#
```

Step 4 Session in to the AIM IPS.

```
router# service-module ids-sensor slot/port session
```

Example:

```
router# service-module ids-sensor 0/0 session
```

Step 5 Press **Control-shift-6** followed by **x** to navigate to the router CLI.

Step 6 Reset the AIM IPS from the router console.

```
router# service-module ids-sensor 0/0 reset
```

Step 7 Press **Enter** to return to the router console.

Step 8 When prompted for boot options, enter `***` quickly. You are now in the bootloader.

Step 9 Clear the password.

```
ServicesEngine boot-loader# clear password
```

The AIM IPS reboots. The password is reset to **cisco**. Log in to the CLI with username **cisco** and password **cisco**. You can then change the password.

Password Recovery for the AIP SSM

You can reset the password to the default (**cisco**) for the AIP SSM 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 7.2.2 or later.

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.
```

Resetting the Password Using the CLI

To reset the password on the AIP SSM, follow these steps:

Step 1 Log into the adaptive security appliance and enter the following command to verify the module slot number:

```
asa# show module
Mod Card Type                                    Model              Serial No.
--- -------------------------------------------- ------------------ -----------
  0 ASA 5510 Adaptive Security Appliance         ASA5510            JMX1135L097
  1 ASA 5500 Series Security Services Module-40  ASA-SSM-40         JAF1214AMRL

Mod MAC Address Range                 Hw Version   Fw Version   Sw Version
--- --------------------------------- ------------ ------------ ---------------
  0 001b.d5e8.e0c8 to 001b.d5e8.e0cc  2.0          1.0(11)2     8.4(3)
  1 001e.f737.205f to 001e.f737.205f  1.0          1.0(14)5     7.0(7)E4

Mod SSM Application Name           Status           SSM Application Version
--- ------------------------------ ---------------- -------------------------
  1 IPS                            Up               7.0(7)E4

Mod Status             Data Plane Status     Compatibility
--- ------------------ --------------------- -------------
  0 Up Sys             Not Applicable
  1 Up                 Up
```

Step 2 Reset the password for module 1.

```
asa# hw-module module 1 password-reset
Reset the password on module in slot 1? [confirm]
```

Step 3 Press **Enter** to confirm.

```
Password-Reset issued for slot 1.
```

Step 4 Verify the status of the module. Once the status reads Up, you can session to the AIP SSM.

```
asa# show module 1
Mod Card Type                                    Model              Serial No.
--- -------------------------------------------- ------------------ -----------
  1 ASA 5500 Series Security Services Module-40  ASA-SSM-40         JAF1214AMRL

Mod MAC Address Range                 Hw Version   Fw Version   Sw Version
--- --------------------------------- ------------ ------------ ---------------
  1 001e.f737.205f to 001e.f737.205f  1.0          1.0(14)5     7.0(7)E4

Mod SSM Application Name           Status           SSM Application Version
```

```
--- ------------------------------ ---------------- --------------------------
  1 IPS                              Up               7.0(7)E4

Mod Status             Data Plane Status     Compatibility
--- ------------------ --------------------- -------------
  1 Up                 Up
```

Step 5 Session to the AIP SSM.

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

Step 6 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 7 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.
aip_ssm#
```

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.

Password Recovery for the IDSM2

To recover the password for the IDSM2, you must install a special password recovery image file. This installation only resets the password, all other configuration remains intact. The password recovery image is version-dependent and can be found on the Cisco Download Software site. For IPS 6.*x*, download WS-SVC-IDSM2-K9-a-6.0-password-recovery.bin.gz. For IPS 7.*x*, download WS-SVC-IDSM2-K9-a-7.0-password-recovery.bin.gz.

FTP is the only supported protocol for image installations, so make sure you put the password recovery image file on an FTP server that is accessible to the switch. You must have administrative access to the Cisco 6500 series switch to recover the password on the IDSM2.

During the password recovery image installation, the following message appears:

```
Upgrading will wipe out the contents on the hard disk.
Do you want to proceed installing it [y|n]:
```

This message is in error. Installing the password recovery image does not remove any configuration, it only resets the login account.

Once you have downloaded the password recovery image file, follow the instructions to install the system image file but substitute the password recovery image file for the system image file. The IDSM2 should reboot into the primary partition after installing the recovery image file. If it does not, enter the following command from the switch:

```
hw-module module module_number reset hdd:1
```

Note The password is reset to **cisco**. Log in to the CLI with username **cisco** and password **cisco**. You can then change the password.

For More Information

- For the procedures for reimaging the IDSM2, see Installing the IDSM2 System Image, page 22-26.
- For more information on downloading Cisco IPS software, see Obtaining Cisco IPS Software, page 21-1.

Password Recovery for the NME IPS

To recover the password for the NME IPS, use the **clear password** command. You must have console access to the NME IPS and administrative access to the router.

To recover the password for the NME IPS, follow these steps:

Step 1 Log in to the router.

Step 2 Enter privileged EXEC mode on the router.

```
router> enable
```

Step 3 Confirm the module slot number in your router.

```
router# show run | include ids-sensor
interface IDS-Sensor1/0
router#
```

Step 4 Session in to the NME IPS.

```
router# service-module ids-sensor slot/port session
```

Example:

```
router# service-module ids-sensor 1/0 session
```

Step 5 Press **Control-shift-6** followed by **x** to navigate to the router CLI.

Step 6 Reset the NME IPS from the router console.

```
router# service-module ids-sensor 1/0 reset
```

Step 7 Press **Enter** to return to the router console.

Step 8 When prompted for boot options, enter ******* quickly. You are now in the bootloader.

Step 9 Clear the password.

```
ServicesEngine boot-loader# clear password
```

The NME IPS reboots. The password is reset to **cisco**. Log in to the CLI with username **cisco** and password **cisco**. You can then change the password.

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, 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 IDM

To disable password recovery in IDM or IME, follow these steps:

Step 1 Log in to IDM or IME 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, such as the AIM IPS and the NME IPS bootloader, ROMMON, and the maintenance partition for the IDSM2, 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.
- When performing password recovery on the IDSM2, you see the following message: `Upgrading will wipe out the contents on the storage media`. You can ignore this message. Only the password is reset when you use the specified password recovery image.

Time and the Sensor

This section describes how to maintain accurate time on the sensor, and contains the following topics:

Time Sources and the Sensor

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.

Note We recommend that you use an NTP server. 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.

Here is a summary of ways to set the time on sensors:

- For appliances
  - Use the **clock set** command to set the time. This is the default.
  - Use NTP—You can configure the appliance to get its time from an NTP time synchronization source.
- For the IDSM2
  - The IDSM2 can automatically synchronize its clock with the switch time. This is the default. The UTC time is synchronized between the switch and the IDSM2. The time zone and summertime settings are not synchronized between the switch and the IDSM2.

    Note Be sure to set the time zone and summertime settings on both the switch and the IDSM2 to ensure that the UTC time settings are correct. The local time of the IDSM2 could be incorrect if the time zone and/or summertime settings do not match between the IDSM2 and the switch.

  - Use NTP—You can configure the IDSM2 to get its time from an NTP time synchronization source.

- For the AIM IPS and the NME IPS
  - The AIM IPS and the NME IPS can automatically synchronize their clock with the clock in the router chassis in which they are installed (parent router). This is the default. The UTC time is synchronized between the parent router and the AIM IPS and the NME IPS. The time zone and summertime settings are not synchronized between the parent router and the AIM IPS and the NME IPS.

    Note Be sure to set the time zone and summertime settings on both the parent router and the AIM IPS and the NME IPS to ensure that the UTC time settings are correct. The local time of the AIM IPS and the NME IPS could be incorrect if the time zone and/or summertime settings do not match between the AIM IPS and the NME IPS and the router.

  - Use NTP—You can configure the AIM IPS and the NME IPS to get their time from an NTP time synchronization source, such as a Cisco router, other than the parent router.
- For the AIP SSM
  - The AIP SSM can automatically synchronize its clock with the clock in the adaptive security appliance in which it is installed. This is the default. The UTC time is synchronized between the adaptive security appliance and the AIP SSM. The time zone and summertime settings are not synchronized between the adaptive security appliance and the AIP SSM.

    Note Be sure to set the time zone and summertime settings on both the adaptive security appliance and the AIP SSM to ensure that the UTC time settings are correct. The local time of the AIP SSM could be incorrect if the time zone and/or summertime settings do not match between the AIP SSM and the adaptive security appliance.

  - Use NTP—You can configure the AIP SSM to get its 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 4-38.

Synchronizing IPS Module Clocks with Parent Device Clocks

All IPS modules (AIM IPS, AIP SSM, IDSM2, and NME IPS) synchronize their system clocks to the parent chassis clock (switch, router, or security appliance) each time the module boots up and any time the parent chassis clock is set. The module 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 module clock and the parent clock are synchronized to an external NTP server. If only the module 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 6.1, 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-93.

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:

- IPS 4240
- IPS 4255
- IPS 4260
- IPS 4270-20
- AIP SSM

Note The IDSM2 supports virtualization with the exception of VLAN groups on inline interface pairs. The AIM IPS and the NME IPS do not support virtualization.

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
- 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 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 Worms, see Understanding Worms, page 9-2.

Troubleshooting External Product Interfaces

This section lists issues that can occur with external product interfaces and provides troubleshooting tips. It contains the following topics:

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 7-25 and Displaying and Clearing OS Identifications, page 7-29.
- For the procedure for adding trusted hosts, see Adding TLS Trusted Hosts, page 4-46.
- For more information on external product interfaces, see Chapter 10, “Configuring External Product Interfaces.”

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 IDM and check the Interface state line in the response, or choose **Configuration >** ***sensor_name*** **> Sensor Monitoring > Support Information > Statistics** in 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 CSA MC using the browser.
- Check Event Store for CSA MC subscription errors.

For More Information

- For the procedure for adding trusted hosts, see Adding TLS Trusted Hosts, page 4-46.
- For the procedure for displaying events, see Displaying Events, page C-90.

Troubleshooting the 4200 Series Appliance

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.

This section contains information to troubleshoot the 4200 series appliance. It contains the following topics:

Troubleshooting Loose Connections

Perform the following actions to troubleshoot loose connections on a sensor:

- 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.

Analysis Engine is Busy

After you reimage a sensor, Analysis Engine is busy rebuilding Regex tables and does not respond to new configurations. You can check whether Analysis Engine is busy by using the **show statistics virtual-sensor** command. You receive the following error message if 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 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 Analysis Engine is busy, wait a while before trying to make configuration changes. Use the **show statistics virtual-sensor** command to find out when Analysis Engine is available again.

Connecting the IPS 4240 to a Cisco 7200 Series Router

When an IPS 4240 is connected directly to a 7200 series router and both the IPS 4240 and the router interfaces are hard-coded to speed 100 with duplex Full, the connection does not work. If you set the IPS 4240 to speed Auto and duplex Auto, it connects to the router but only at speed 100 and duplex Half.

To connect correctly at speed 100 and duplex Full, set the interfaces of both the IPS 4240 and the router to speed Auto and duplex Auto. Also, if either interface is hard-coded, you must make the connection using a crossover cable.

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

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.

```
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#
```

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.

Step 3 Make sure the sensor IP address is unique.

```
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 10.89.130.108/23,10.89.130.1
host-name sensor
telnet-option enabled
access-list 0.0.0.0/0
ftp-timeout 300
no login-banner-text
exit
--MORE--
```

If the management interface detects that another device on the network has the same IP address, it does not come up.

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.

```
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 10.89.130.108/23,10.89.130.1
host-name sensor
telnet-option enabled
access-list 0.0.0.0/0
ftp-timeout 300
no login-banner-text
exit
--MORE--
```

If the workstation network address is permitted in the sensor access list, go to Step 6.

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 4-4.
- 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 4-3.
- 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: 10.89.149.238/25,10.89.149.254 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.

```
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#
```

If the output says the command and control interface link status is down, there is a hardware issue or an IP address conflict.

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 *Installing Cisco Intrusion Prevention System Appliances and Module 6.1*.
- For the procedure for making sure the IP address is correct, see Changing the IP Address, Netmask, and Gateway, page 4-3.

SensorApp and Alerting

This section helps you troubleshoot issues with SensorApp and alerting. It contains the following topics:

SensorApp Not Running

The sensing process, SensorApp, should always be running. If it is not, you do not receive any alerts. SensorApp is part of Analysis Engine, so you must make sure the Analysis Engine is running.

To make sure Analysis Engine is running, follow these steps:

Step 1 Log in to the CLI.

Step 2 Determine the status of the Analysis Engine service.

```
sensor# show version
Application Partition:

Cisco Intrusion Prevention System, Version 6.1(1)E1

Host:
    Realm Keys          key1.0
Signature Definition:
    Signature Update    S329.0                   2008-04-16
    Virus Update        V1.2                     2005-11-24
OS Version:             2.4.30-IDS-smp-bigphys
Platform:               ASA-SSM-20
Serial Number:          JAB0948035P
License expired:        11-Apr-2008 UTC
Sensor up-time is 7 days.
Using 1018015744 out of 2093600768 bytes of available memory (48% usage)
system is using 17.7M out of 29.0M bytes of available disk space (61% usage)
application-data is using 39.7M out of 166.6M bytes of available disk space (25% usage)
boot is using 40.5M out of 68.5M bytes of available disk space (62% usage)


MainApp          M-2008_APR_24_19_16   (Release)   2008-04-24T19:49:05-0500   Running
AnalysisEngine   M-2008_APR_24_19_16   (Release)   2008-04-24T19:49:05-0500   NotRunning
CLI              M-2008_APR_24_19_16   (Release)   2008-04-24T19:49:05-0500

Upgrade History:

  IPS-K9-6.1-1-E1   01:16:00 UTC Fri Apr 25 2008

Recovery Partition Version 1.1 - 6.1(1)E1

Host Certificate Valid from: 29-Jun-2008 to 30-Jun-2010

sensor#
```

Step 3 If 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 Make sure you have the latest software updates.

```
sensor# show version
Application Partition:
```

```
Cisco Intrusion Prevention System, Version 6.1(1)E1

Host:
    Realm Keys          key1.0
Signature Definition:
    Signature Update    S329.0                  2008-04-16
    Virus Update        V1.2                    2005-11-24
OS Version:             2.4.30-IDS-smp-bigphys
Platform:               ASA-SSM-20
Serial Number:          JAB0948035P
License expired:        11-Apr-2008 UTC
Sensor up-time is 7 days.
Using 1018015744 out of 2093600768 bytes of available memory (48% usage)
system is using 17.7M out of 29.0M bytes of available disk space (61% usage)
application-data is using 39.7M out of 166.6M bytes of available disk space (25% usage)
boot is using 40.5M out of 68.5M bytes of available disk space (62% usage)


MainApp          M-2008_APR_24_19_16   (Release)   2008-04-24T19:49:05-0500   Running
AnalysisEngine   M-2008_APR_24_19_16   (Release)   2008-04-24T19:49:05-0500   NotRunning
CLI              M-2008_APR_24_19_16   (Release)   2008-04-24T19:49:05-0500

Upgrade History:

  IPS-K9-6.1-1-E1   01:16:00 UTC Fri Apr 25 2008

Recovery Partition Version 1.1 - 6.1(1)E1

Host Certificate Valid from: 29-Jun-2008 to 30-Jun-2010

sensor#
```

If you do not have the latest software updates, download them from Cisco.com.

Step 5 Read the Readme that accompanies the software upgrade for any known DDTS for SensorApp or Analysis Engine.

For More Information

- For more information on IPS system architecture, see Appendix A, "System Architecture."
- For the procedure for obtaining the latest Cisco IPS software, see Obtaining Cisco IPS Software, page 21-1.

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:

a. Make sure the sensing port is connected properly on the appliance.

b. Make sure the sensing port is connected to the correct SPAN or VACL capture port on the IDSM2.

Step 4 Verify the interface configuration:

a. Make sure you have the interfaces configured properly.

b. 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 *Installing Cisco Intrusion Prevention System Appliances and Modules 6.1*.
- For the procedure for connecting SPAN and VACL capture ports on the IDSM2, see Chapter 19, "Configuring the IDSM2."
- 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 be 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:

a. Check the cabling.

b. 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 information on installing the sensor properly, refer to your sensor chapter in *Installing Cisco Intrusion Prevention System Appliances and Modules 6.1*.

Cleaning Up a Corrupted SensorApp Configuration

If the SensorApp configuration has become corrupted and SensorApp cannot run, you must delete it entirely and restart 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

For more information on 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

After you have configured ARC, you can verify if it is running properly by using the **show version** command. To verify that ARC is connecting to the network devices, use the **show statistics network-access** command.

Note 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 ARC, follow these steps:

1. Verify that ARC is running.
2. Verify that 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 ARC is running, see Verifying ARC is Running, page C-37.
- For the procedure to verify that ARC is connecting, see Verifying ARC Connections are Active, page C-38.
- For the procedure to verify that the Event Action is set to Block Host, see Blocking Not Occurring for a Signature, page C-42.

- For the procedure to verify that the master blocking sensor is properly configured, see Verifying the Master Blocking Sensor Configuration, page C-43.
- For a discussion of ARC architecture, see Attack Response Controller, page A-12.

Verifying ARC is Running

To verify that ARC is running, use the **show version** command. If MainApp is not running, ARC cannot run. ARC is part of MainApp.

To verify that ARC is running, following these steps:

Step 1 Log in to the CLI.

Step 2 Verify that MainApp is running.

```
sensor# show version
Application Partition:

Cisco Intrusion Prevention System, Version 6.0(1)E1.1

Host:
    Realm Keys          key1.0
Signature Definition:
    Signature Update    S294.0                  2007-08-02
    Virus Update        V1.2                    2005-11-24
OS Version:             2.4.30-IDS-smp-bigphys
Platform:               ASA-SSM-20
Serial Number:          P300000220
No license present
Sensor up-time is 6 days.
Using 1026641920 out of 2093682688 bytes of available memory (49% usage)
system is using 17.8M out of 29.0M bytes of available disk space (61% usage)
application-data is using 38.4M out of 166.6M bytes of available disk space (24% usage)
boot is using 38.0M out of 68.5M bytes of available disk space (58% usage)


MainApp          N-2007_SEP_20_16_44   (Release)   2007-09-20T17:10:01-0500   Running
AnalysisEngine   N-2007_SEP_20_16_44   (Release)   2007-09-20T17:10:01-0500   Running
CLI              N-2007_SEP_20_16_44   (Release)   2007-09-20T17:10:01-0500


Upgrade History:

  IPS-K9-6.0-1-E1.1   16:44:00 UTC Thu Sep 20 2007

Recovery Partition Version 1.1 - 6.0(1)E1.1

sensor#
```

Step 3 If MainApp displays `Not Running`, ARC has failed. Contact the TAC. For more information on 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 ARC is connecting, and 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 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 2007 | include : nac
```

Step 4 Make sure you have the latest software updates.

```
sensor# show version
Application Partition:

Cisco Intrusion Prevention System, Version 6.0(1)E1.1

Host:
    Realm Keys          key1.0
Signature Definition:
    Signature Update    S294.0                  2007-08-02
    Virus Update        V1.2                    2005-11-24
OS Version:             2.4.30-IDS-smp-bigphys
Platform:               ASA-SSM-20
Serial Number:          P300000220
No license present
Sensor up-time is 6 days.
Using 1026641920 out of 2093682688 bytes of available memory (49% usage)
system is using 17.8M out of 29.0M bytes of available disk space (61% usage)
application-data is using 38.4M out of 166.6M bytes of available disk space (24% usage)
```

```
boot is using 38.0M out of 68.5M bytes of available disk space (58% usage)


MainApp          N-2007_SEP_20_16_44   (Release)   2007-09-20T17:10:01-0500   Running
AnalysisEngine   N-2007_SEP_20_16_44   (Release)   2007-09-20T17:10:01-0500   Running
CLI              N-2007_SEP_20_16_44   (Release)   2007-09-20T17:10:01-0500


Upgrade History:

  IPS-K9-6.0-1-E1.1   16:44:00 UTC Thu Sep 20 2007

Recovery Partition Version 1.1 - 6.0(1)E1.1

sensor#
```

Note If you do not have the latest software updates, download them from Cisco.com.

Step 5 Read the Readme that accompanies the software upgrade for any known DDTS for ARC.

Step 6 Make sure the configuration settings for each device are correct (the username, password, and IP address).

Step 7 Make sure the interface and directions for each network device are correct.

Step 8 If the network device is using SSH-DES or SSH-3DES, make sure that you have enabled SSH connections to the device.

Step 9 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 21-1.
- 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-41.
- For the procedure for enabling SSH, see Enabling SSH Connections to the Network Device, page C-42.

Device Access Issues

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.

Note SSH devices must support SSH 1.5. The sensor does not support SSH 2.0.

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-41.

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-DES or SSH-3DES as the communication protocol for the network device, you must make sure you have enabled it on the device.

To enable SSH 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.

```
sensor(config)# ssh 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 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 2 If the master blocking sensor does not show up in the statistics, you need to add it.

Step 3 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 4 Exit network access general submode.

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

Step 5 Press **Enter** to apply the changes or type **no** to discard them.

Step 6 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 7 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 8 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 13-28.

Logging

This section contains the following topics:

Understanding Debug 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.

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 To 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 To 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: intfc
      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: intfc
      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: intfc
      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-49.

Zone Names

Table C-2 lists the debug logger zone names:

Table C-2 Debug Logger Zone Names

Zone Name	Description
AuthenticationApp	Authentication zone
Cid	General logging zone
Cli	CLI zone
IdapiCtlTrans	All control transactions zone
IdsEventStore	Event Store zone
MpInstaller	IDSM2 master partition installer zone
cmgr	Card Manager service zone[1]
cplane	Control Plane zone[2]
csi	CIDS Servlet Interface[3]
ctlTransSource	Outbound control transactions zone
intfc	Interface zone
nac	ARC zone
sensorApp	AnalysisEngine zone
tls	SSL and TLS zone

1. The Card Manager service is used on the AIP SSM to exchange control and state information between modules in the chassis.
2. The Control Plane is the transport communications layer used by Card Manager on the AIP SSM.
3. The CIDS servlet interface is the interface layer between the CIDS web server and the servlets.

For More Information

For more information on 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

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|produc-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 submode.

```
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 alerts 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:

Upgrading to 6.0

If you try to upgrade an IPS 5.*x* sensor to 6.*x*, you may receive an error that Analysis Engine is not running:

```
sensor# upgrade scp://user@10.1.1.1/upgrades/IPS-K9-6.0-1-E1.pkg
Password: ********
Warning: Executing this command will apply a major version upgrade to the application
partition. The system may be rebooted to complete the upgrade.
Continue with upgrade?: yes
Error: Analysis Engine is not running. Please reset box and attempt upgrade again.
```

If you receive this error, you must get Analysis Engine running before trying to upgrade again. This error is often caused by a defect in the currently running version. Try rebooting the sensor, and after reboot, run the **setup** command and remove the interfaces from the virtual sensor vs0. When it is not monitoring traffic, Analysis Engine usually stays up and running. You can upgrade to 6.*x* at this time. After the upgrade to IPS 6.*x*, add the interfaces back to the virtual sensor vs0 using the **setup** command.

Or you can use the system image file to reimage directly to IPS 6.*x*. You can reimage a 5.*x* sensor to 6.*x* because the reimage process does not check to see if Analysis Engine is running.

Caution Reimaging using the system image file restores all configuration defaults.

For More Information

- For more information on running the **setup** command, see Chapter 3, "Initializing the Sensor."
- For more information on reimaging your sensor, see Chapter 22, "Upgrading, Downgrading, and Installing System Images."

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

For more information on how to interpret the IPS software filenames, see IPS Software Versioning, page 21-3.

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 4-14.
- For the procedure for reimaging your sensor, see Chapter 22, "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 4-42.
- For the procedure for determining the software version, see Displaying Version Information, page C-75.

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)# rsa1-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 21-1.

Troubleshooting IDM

Note These procedures also apply to the IPS section of 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 IDM, and contains the following topics:

Cannot Launch 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 (1.4.*x* and/or 1.3.*x*) 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 IDM-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 IDM.

Recommended Action Wait for a while and try again to connect.

IDM, Remote Manager, or Sensing Interfaces Cannot Access Sensor

If 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 10.89.130.108/23,10.89.130.1
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 4-4.
- For the procedure for configuring the web server, see Changing Web Server Settings, page 4-10.

Signatures Not Producing Alerts

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.

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.

To make sure you are getting alerts, check the statistics for the virtual sensor and Event Store.

For More Information

- For more information about event actions, see Event Actions, page 7-8.
- For the procedure for configuring event actions, see Assigning Actions to Signatures, page 8-15.
- For the procedure for obtaining statistics about virtual sensor and Event Store, see Displaying Statistics, page 16-26.

Troubleshooting IME

This section describes troubleshooting tools for IME, and contains the following sections:

- Time Synchronization on IME and the Sensor, page C-57
- Not Supported Error Message, page C-58

Time Synchronization on IME and the Sensor

Symptom IME displays `No Data Available` on the Events dashboard. A historical query does not return any events; however, events are coming in to 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 IME, it checks for the time synchronization and warns you to correct it if is in wrong. 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 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-16.
- For the procedure for changing the time on the sensor, see Correcting Time on the Sensor, page C-18.

Not Supported Error Message

Symptom 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. IME needs IPS 6.1 or later to obtain certain information. 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.

Troubleshooting the IDSM2

Note The IDSM2 has the same software architecture as the 4200 series sensors. You can use the same troubleshooting tools as outlined in Troubleshooting the 4200 Series Appliance, page C-22.

This section pertains specifically to troubleshooting the IDSM2, and contains the following topics:

Diagnosing IDSM2 Problems

Use the following list to diagnose IDSM2 problems:

- The ribbon cable between the IDSM2 and the motherboard is loose.

  During physical handling of the module, the connector can come loose from the base card, and cause the daughter card and the base card to lose contact with each other. A loose ribbon cable connector causes an on-line diagnostic error on ports 7 and 8. The module cannot operate when this condition exists. For more information, refer to Partner Field Notice 29877.

- Some IDSM2s were shipped with faulty DIMMs. For the procedure for checking the IDSM2 for faulty memory, refer to Partner Field Notice 29837.
- The hard-disk drive fails to read or write.

  When the hard-disk drive has been in constant use for extended periods of time (for more than 2 weeks), multiple symptoms, such as the following, can occur:

  - An inability to log in
  - I/O errors to the console when doing read/write operations (the **ls** command)
  - Commands do not execute properly (cannot find the path to the executable)

  The switch reports that the module is ok, but if you log in to the Service account and try to execute commands, you see that the problem exists. The 4.1(4) service pack alleviates this problem, but if you reimage the IDSM2 with the 4.1(4) application partition image, you must apply the 4.1(4b) patch. For more information, refer to CSCef12198.
- SensorApp either crashes or takes 99% of the CPU when IP logging is enabled for stream-based signatures (1300 series). For the workaround, refer to CSCed32093.
- The IDSM2 appears to lock up and remote access is prohibited (SSH, Telnet, IDM, Event Server, Control Transaction Server, and IP log Server).

  This defect is related to using SWAP. The IDSM2 responds to pings. Apply the 4.1(4) service pack to resolve this issue. For more information, refer to CSCed54146.
- Shortly after you upgrade the IDSM2 or you tune a signature with VMS, the IDSM2 becomes unresponsive and often produces a SensorApp core file. Apply the 4.1(4b) patch to fix this issue.
- Confirm that the IDSM2 has the supported configurations.

If you have confirmed that the IDSM2 does not suffer from any of the problems listed above and yet it appears unresponsive, for example, you cannot log in through SSH or Telnet, nor can you session to the switch, determine if the IDSM2 responds to pings and if you can log in through the service account. If you can log in, obtain a cidDump and any core files and contact TAC.

For More Information

- For information about the Bug Toolkit and how to access it, see Bug ToolKit, page C-1.
- For a table listing the supported IDSM2 configurations, see Minimum Supported IDSM2 Configurations, page C-59.

Minimum Supported IDSM2 Configurations

Note The following matrix is not intended to recommend any particular version, but rather lists the earliest supported versions.

Table C-3 lists the minimum supported configurations for the IDSM2.

Table C-3 Minimum Catalyst 6500 Software Version for IDSM2 Feature Support

Catalyst/IDSM2 Feature	Catalyst Software				Cisco IOS Software			
	Sup1	Sup2	Sup32	Sup720	Sup1	Sup2	Sup32	Sup720
SPAN	7.5(1)	7.5(1)	8.4(1)	8.1(1)	12.1(19)E1	12.1(19)E1 12.2(18)SXF1	12.2(18)SXF1	12.2(14)SX1

Table C-3 Minimum Catalyst 6500 Software Version for IDSM2 Feature Support (continued)

Catalyst/IDSM2 Feature	Catalyst Software				Cisco IOS Software			
VACL capture[1]	7.5(1)	7.5(1)	8.4(1)	8.1(1)	12.1(19)E1	12.1(19)E1 12.2(18)SXF1	12.2(18)SXF1	12.2(14)SX1
ECLB with VACL capture[2]	8.5(1)	8.5(1)	8.5(1)	8.5(1)	N/A	12.2(18)SXF4	12.2(18)SXF1	12.2(18)SXE1
Inline interface pairs	8.4(1)	8.4(1)	8.4(1)	8.4(1)	N/A	12.2(18)SXF4	12.2(18)SXF4	12.2(18)SXE1
ECLB with inline interface pairs	8.5(1)	8.5(1)	8.5(1)	8.5(1)	N/A	12.2(18)SXF4	12.2(18)SXF4	12.2(18)SXF4
Inline VLAN pairs	8.4(1)	8.4(1)	8.4(1)	8.4(1)	N/A	12.2(18)SXF4	12.2(18)SXF4	12.2(18)SXF4
ECLB with inline VLAN pairs	8.5(1)	8.5(1)	8.5(1)	8.5(1)	N/A	12.2(18)SXF4	12.2(18)SXF4	12.2(18)SXF4

1. Requires PFC2/3 or MSFC2/3.
2. Requires PFC2/3 or MSFC2/3.

Switch Commands for Troubleshooting

The following switch commands help you troubleshoot the IDSM2:

- **show module** (Catalyst software and Cisco IOS software)
- **show version** (Catalyst software and Cisco IOS software)
- **show port** (Catalyst software)
- **show trunk** (Catalyst software)
- **show span** (Catalyst software)
- **show security acl** (Catalyst software)
- **show intrusion-detection module** (Cisco IOS software)
- **show monitor** (Cisco IOS software)
- **show vlan access-map** (Cisco IOS software)
- **show vlan filter** (Cisco IOS software)

Status LED Off

If the status indicator is off on the IDSM2, you need to turn power on to the IDSM2.

To determine the status of the IDSM2, follow these steps:

Step 1 Log in to the console.

Step 2 Verify that the IDSM2 is online.

For Catalyst Software

```
console> enable

Enter password:
console> (enable) show module
```

```
Mod Slot Ports Module-Type               Model               Sub Status
--- ---- ----- ------------------------- ------------------- --- --------
1   1    2     1000BaseX Supervisor      WS-X6K-SUP1A-2GE    yes ok
15  1    1     Multilayer Switch Feature WS-F6K-MSFC         no  ok
2   2    48    10/100BaseTX Ethernet     WS-X6248-RJ-45      no  ok
3   3    48    10/100/1000BaseT Ethernet WS-X6548-GE-TX      no  ok
4   4    16    1000BaseX Ethernet        WS-X6516A-GBIC      no  ok
6   6    8     Intrusion Detection Mod   WS-SVC-IDSM2        yes ok

Mod Module-Name          Serial-Num
--- -------------------- -----------
1                        SAD041308AN
15                       SAD04120BRB
2                        SAD03475400
3                        SAD073906RC
4                        SAL0751QYN0
6                        SAD062004LV

Mod MAC-Address(es)                         Hw     Fw          Sw
--- -------------------------------------- ------ ---------- ----------------
1   00-d0-c0-cc-0e-d2 to 00-d0-c0-cc-0e-d3 3.1    5.3.1      8.4(1)
    00-d0-c0-cc-0e-d0 to 00-d0-c0-cc-0e-d1
    00-30-71-34-10-00 to 00-30-71-34-13-ff
15  00-30-7b-91-77-b0 to 00-30-7b-91-77-ef 1.4    12.1(23)E2 12.1(23)E2
2   00-30-96-2b-c7-2c to 00-30-96-2b-c7-5b 1.1    4.2(0.24)V 8.4(1)
3   00-0d-29-f6-01-98 to 00-0d-29-f6-01-c7 5.0    7.2(1)     8.4(1)
4   00-0e-83-af-15-48 to 00-0e-83-af-15-57 1.0    7.2(1)     8.4(1)
6   00-e0-b0-ff-3b-80 to 00-e0-b0-ff-3b-87 0.102  7.2(0.67)  5.0(0.30)

Mod Sub-Type                Sub-Model           Sub-Serial  Sub-Hw Sub-Sw
--- ----------------------- ------------------- ----------- ------ ------
1   L3 Switching Engine     WS-F6K-PFC          SAD041303G6 1.1
6   IDS 2 accelerator board WS-SVC-IDSUPG       .           2.0
console> (enable)
```

For Cisco IOS software

```
router# show module
Mod Ports Card Type                                Model              Serial No.
--- ----- ------------------------------------- ------------------ -----------
  1   48  48 port 10/100 mb RJ-45 ethernet       WS-X6248-RJ-45     SAD0401012S
  2   48  48 port 10/100 mb RJ45                 WS-X6348-RJ-45     SAL04483QBL
  3   48  SFM-capable 48 port 10/100/1000mb RJ45 WS-X6548-GE-TX     SAD073906GH
  5    8  Intrusion Detection System             WS-SVC-IDSM-2      SAD0751059U
  6   16  SFM-capable 16 port 1000mb GBIC        WS-X6516A-GBIC     SAL0740MMYJ
  7    2  Supervisor Engine 720 (Active)         WS-SUP720-3BXL     SAD08320L2T
  9    1  1 port 10-Gigabit Ethernet Module      WS-X6502-10GE      SAD071903BT
 11    8  Intrusion Detection System             WS-SVC-IDSM2       SAD05380608
 13    8  Intrusion Detection System             WS-SVC-IDSM-2      SAD072405D8

Mod MAC addresses                        Hw    Fw           Sw           Status
--- ---------------------------------- ------ ------------ ------------ -------
  1  00d0.d328.e2ac to 00d0.d328.e2db   1.1   4.2(0.24)VAI 8.5(0.46)ROC Ok
  2  0003.6c14.e1d0 to 0003.6c14.e1ff   1.4   5.4(2)       8.5(0.46)ROC Ok
  3  000d.29f6.7a80 to 000d.29f6.7aaf   5.0   7.2(1)       8.5(0.46)ROC Ok
  5  0003.fead.651a to 0003.fead.6521   4.0   7.2(1)       5.0(1.1)     Ok
  6  000d.ed23.1658 to 000d.ed23.1667   1.0   7.2(1)       8.5(0.46)ROC Ok
  7  0011.21a1.1398 to 0011.21a1.139b   4.0   8.1(3)       12.2(PIKESPE Ok
  9  000d.29c1.41bc to 000d.29c1.41bc   1.3   Unknown      Unknown      PwrDown
 11  00e0.b0ff.3340 to 00e0.b0ff.3347   0.102 7.2(0.67)    5.0(1.1)     Ok
 13  0003.feab.c850 to 0003.feab.c857   4.0   7.2(1)       5.0(1)       Ok

Mod Sub-Module                  Model               Serial        Hw     Status
--- --------------------------- ------------------ ------------ ------- -------
```

```
  5 IDS 2 accelerator board     WS-SVC-IDSUPG      07E91E508A    2.0    Ok
  7 Policy Feature Card 3       WS-F6K-PFC3BXL     SAD083305A1   1.3    Ok
  7 MSFC3 Daughterboard         WS-SUP720          SAD083206JX   2.1    Ok
 11 IDS 2 accelerator board     WS-SVC-IDSUPG      .             2.0    Ok
 13 IDS 2 accelerator board     WS-SVC-IDSUPG      0347331976    2.0    Ok

Mod Online Diag Status
--- -------------------
  1 Pass
  2 Pass
  3 Pass
  5 Pass
  6 Pass
  7 Pass
  9 Unknown
 11 Pass
 13 Pass
router#
```

Note It is normal for the status to read `other` when the IDSM2 is first installed. After the IDSM2 completes the diagnostics routines and comes online, the status reads `ok`. Allow up to 5 minutes for the IDSM2 to come online.

Step 3 If the status does not read `ok`, turn the module on.

```
router# set module power up module_number
```

Status LED On But the IDSM2 Does Not Come Online

If the status indicator is on, but the IDSM2 does not come online, try the following troubleshooting tips:

- Reset the IDSM2.
- Make sure the IDSM2 is installed properly in the switch.
- If the hard-disk drive status has failed, reimage the application partition.

To enable the IDSM2, follow these steps:

Step 1 Log in to the console.

Step 2 Make sure the IDSM2 is enabled.

```
router# show module
```

Step 3 If the status does not read `ok`, enable the IDSM2.

```
router# set module enable module_number
```

Step 4 If the IDSM2 still does not come online, reset it.

```
router# reset module_number
```

Wait for about 5 minutes for the IDSM2 to come online.

Step 5 If the IDSM2 still does not come online, make sure the hardware and operating system are ok.

```
router# show test module_number
```

Step 6 If the `port` status reads `fail`, make sure the IDSM2 is firmly connected in the switch.

Step 7 If the `hdd` status reads `fail`, you must reimage the application partition.

For More Information

For the procedure for reimaging the application partition, see Chapter 22, "Upgrading, Downgrading, and Installing System Images."

Cannot Communicate With the IDSM2 Command and Control Port

If you cannot communicate with the IDSM2 command and control port, the command and control port may not be in the correct VLAN.

To communicate with the command and control port of the IDSM2, follow these steps:

Step 1 Log in to the console.

Step 2 Make sure you can ping the command port from any other system.

Step 3 Make sure the IP address, mask, and gateway settings are correct.

```
router# show configuration
```

Step 4 Make sure the command and control port is in the correct VLAN.

For Catalyst software

```
console> (enable) show port 6/8
* = Configured MAC Address

# = 802.1X Authenticated Port Name.

Port  Name                 Status     Vlan       Duplex Speed       Type
----- -------------------- ---------- ---------- ------ ----------- ------------
 6/8                       connected  trunk        full        1000 IDS


Port  Status      ErrDisable Reason    Port ErrDisableTimeout  Action on Timeout
----  ----------  -------------------  ----------------------  -----------------
 6/8  connected                      -  Enable                  No Change

Port  Align-Err  FCS-Err    Xmit-Err   Rcv-Err    UnderSize
----- ---------- ---------- ---------- ---------- ---------
 6/8           0          0          0          0         0

Port  Single-Col Multi-Coll Late-Coll  Excess-Col Carri-Sen Runts     Giants
----- ---------- ---------- ---------- ---------- --------- --------- ---------
 6/8           0          0          0          0         0         0         -

Port  Last-Time-Cleared
----- --------------------------
 6/8  Wed Mar 2 2005, 15:29:49

Idle Detection
--------------
   --
console> (enable)
```

For Cisco IOS software

```
router# show intrusion-detection module 5 management-port state
Intrusion-detection module 5 management-port:

Switchport: Enabled
Administrative Mode: dynamic desirable
Operational Mode: static access
Administrative Trunking Encapsulation: negotiate
Operational Trunking Encapsulation: native
Negotiation of Trunking: On
Access Mode VLAN: 1 (default)
Trunking Native Mode VLAN: 1 (default)
Trunking VLANs Enabled: ALL
Pruning VLANs Enabled: 2-1001
Vlans allowed on trunk:1
Vlans allowed and active in management domain: 1
Vlans in spanning tree forwarding state and not pruned:
   1
Access Vlan = 1


router#
```

Step 5 If the command and control port is not in the correct VLAN, put it in the correct VLAN.

For More Information

For the procedure for configuring the switch for command and control access to the IDSM2, see Configuring the Catalyst 6500 Series Switch for Command and Control Access to the IDSM2, page 19-5.

Using the TCP Reset Interface

The IDSM2 has a TCP reset interface—port 1. The IDSM2 has a specific TCP reset interface because it cannot send TCP resets on its sensing ports.

If you have reset problems with the IDSM2, and the switch is running Catalyst software, try the following:

- If the sensing ports are access ports (a single VLAN), you need to configure the reset port to be in the same VLAN.
- If the sensing ports are dot1q trunk ports (multi-VLAN), the sensing ports and reset port all must have the same native VLAN, and the reset port must trunk all the VLANs being trunked by both the sensing ports.

Note In Cisco IOS when the IDSM2 is in promiscuous mode, the IDSM2 ports are always dot1q trunk ports (even when monitoring only 1 VLAN), and the TCP reset port is automatically set to a trunk port and is not configurable.

For More Information

For more information, see Chapter 19, “Configuring the IDSM2.”

Connecting a Serial Cable to the IDSM2

You can connect a serial cable directly to the serial console port on the IDSM2. This lets you bypass the switch and module network interfaces. To connect a serial cable to the IDSM2, follow these steps:

Step 1 Locate the two RJ-45 ports on the IDSM2.

You can find them approximately in the center of the mother board. If you are facing the module faceplate, the RJ-45 port on the right is the serial console port.

Step 2 Connect a straight-through cable to the right port on the IDSM2, and then connect the other end of the cable to a terminal server port.

Step 3 Configure the terminal server port to be 19200 baud, 8 bits, no parity.

You can now log directly in to the IDSM2.

Note Connecting a serial cable to the IDSM2 works only if there is no module located above the IDSM2 in the switch chassis, because the cable has to come out through the front of the chassis.

Troubleshooting the AIP SSM

Note The AIP SSM has the same software architecture as the 4200 series sensors. You can use the same troubleshooting tools as outlined in Troubleshooting the 4200 Series Appliance, page C-22.

The following section contains information for troubleshooting the AIP SSM, and contains the following topics:

Health and Status Information

To see the general health of the AIP SSM, use the **show module 1 details** command.

```
asa# show module 1 details
Getting details from the Service Module, please wait...
ASA 5500 Series Security Services Module-20
Model:              ASA-SSM-20
Hardware version:   0.2
Serial Number:      P2B000005D0
Firmware version:   1.0(10)0
Software version:   5.1(0.1)S153.0
Status:             Up
Mgmt IP addr:       10.89.149.219
```

```
Mgmt web ports:     443
Mgmt TLS enabled:   true
asa#
```

The output shows that the AIP SSM is up. If the status reads `Down`, you can reset the AIP SSM 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(config)# show module

Mod Card Type                                    Model              Serial No.
--- -------------------------------------------- ------------------ -----------
  0 ASA 5520 Adaptive Security Appliance         ASA5520            P2A00000014
  1 ASA 5500 Series Security Services Module-10  ASA-SSM-10         P2A0000067U

Mod MAC Address Range                 Hw Version   Fw Version   Sw Version
--- --------------------------------- ------------ ------------ ---------------
  0 000b.fcf8.7bdc to 000b.fcf8.7be0  0.2          1.0(10)0     7.0(1)
  1 000b.fcf8.0176 to 000b.fcf8.0176  0.2          1.0(10)0     5.1(0.1)S153.0

Mod Status
--- ------------------
  0 Up Sys
  1 Shutting Down
**************************************************
asa(config)# show module

Mod Card Type                                    Model              Serial No.
--- -------------------------------------------- ------------------ -----------
  0 ASA 5520 Adaptive Security Appliance         ASA5520            P2A00000014
  1 ASA 5500 Series Security Services Module-10  ASA-SSM-10         P2A0000067U

Mod MAC Address Range                 Hw Version   Fw Version   Sw Version
--- --------------------------------- ------------ ------------ ---------------
  0 000b.fcf8.7bdc to 000b.fcf8.7be0  0.2          1.0(10)0     7.0(1)
  1 000b.fcf8.0176 to 000b.fcf8.0176  0.2          1.0(10)0     5.1(0.1)S153.0

Mod Status
--- ------------------
  0 Up Sys
  1 Up
asa(config)#
```

If you have problems with recovering the AIP SSM, use the **debug module-boot** command to see the output as the AIP SSM 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 recover the AIP SSM.

```
asa(config)# hw-module module 1 recover configure
Image URL [tftp://0.0.0.0/]: tftp://10.89.146.1/IPS-SSM-K9-sys-1.1-a-5.1-0.1.i$
Port IP Address [0.0.0.0]: 10.89.150.227
VLAN ID [0]:
Gateway IP Address [0.0.0.0]: 10.89.149.254
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 2005
Slot-1 141> Platform ASA-SSM-10
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=10.89.150.227
Slot-1 147>   SERVER=10.89.146.1
Slot-1 148>   GATEWAY=10.89.149.254
Slot-1 149>   PORT=GigabitEthernet0/0
Slot-1 150>   VLAN=untagged
Slot-1 151>   IMAGE=IPS-SSM-K9-sys-1.1-a-5.1-0.1.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-SSM-K9-sys-1.1-a-5.1-0.1.img@10.89.146.1 via 10.89.149.254
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 2005
Slot-1 161> Platform ASA-SSM-10
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=10.89.150.227
Slot-1 167>   SERVER=10.89.146.1
Slot-1 168>   GATEWAY=10.89.149.254
Slot-1 169>   PORT=GigabitEthernet0/0
Slot-1 170>   VLAN=untagged
Slot-1 171>   IMAGE=IPS-SSM-K9-sys-1.1-a-5.1-0.1.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-SSM-K9-sys-1.1-a-5.1-0.1.img@10.89.146.1 via 10.89.149.254
```

Failover Scenarios

The following failover scenarios apply to the ASA in the event of configuration changes, signature/signature engine updates, service packs, and SensorApp crashes on the AIP SSM.

Single ASA in Fail-Open Mode

- If the ASA is configured in fail-open mode for the AIP SSM, and the AIP SSM 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 AIP SSM, and the AIP SSM 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 AIP SSM, and the AIP SSM 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 AIP SSM, and the AIP SSM 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 AIP SSM 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 AIP SSM on the active ASA experiences a SensorApp crash or a service pack upgrade, failover is triggered and traffic passes through the AIP SSM that was previously the standby module.

Two ASAs in Fail-Close Mode

- If the ASAs are configured in fail-close mode, and if the AIP SSM 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 AIP SSM on the active ASA experiences a SensorApp crash or a service pack upgrade, failover is triggered and traffic passes through the module that was previously the standby for the AIP SSM.

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 AIP SSM and the Data Plane

Symptom The AIP SSM data plane is kept in the Up state while applying signature updates. You can check the AIP SSM data plane status by using the **show module** command during signature updates.

Possible Cause Bypass mode is set to off. The issue is seen when updating signatures, and when you use either CSM or IDM to apply signature updates. This issue is not seen when upgrading IPS system software.

The AIP SSM and the Normalizer Engine

The majority of the features in the Normalizer engine are not used on the AIP SSM, 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-22.

TCP Reset Differences Between IPS Appliances and the AIP SSM

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 AIP SSM, 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.

For More Information

For detailed information about event actions, see Event Actions, page 7-8.

Troubleshooting the AIM IPS and the NME IPS

This section contains information for troubleshooting the IPS network modules, the AIM IPS and the NME IPS. It contains the following sections:

Interoperability With Other IPS Network Modules

The Cisco access routers only support one IDS/IPS module per router. If you have more than one IDS/IPS module installed, the most capable card is enabled. The most capable hierarchy is:

1. NME IPS
2. AIM IPS
3. NM-CIDS

This means, for example, that if all modules are installed, the NME IPS disables all other modules. The AIM IPS disables all NM-CIDS. If there are multiple modules with the same level of capability, the first one discovered is enabled and all others are disabled.

You cannot bring up, enable, or configure a disabled module. To bring up a less capable module, you must remove the more capable module from the router and reboot. Disabled modules are reported in the **show diag** command output. The state of the module is reported as present but disabled.

If the most capable module slot and port do not match the **interface ids slot/port** configuration command, the most capable module is disabled with the following warning:

```
The module in slot x will be disabled and configuration ignored.
```

The correct slot/port number are displayed so that you can change the configuration.

Caution You cannot upgrade an NM-CIDS to an NME IPS. For more information on NM-CIDS, refer to Introducing NM CIDS and Installing NM CIDS.

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 describes how to gather troubleshooting information about your sensor, and contains the following topics:

Health and Network Security Information

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 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.

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
```

```
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-72
- Displaying Tech Support Information, page C-72
- Tech Support Command Output, page C-73

Understanding the show tech-support Command

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.

To get the same information from IDM, choose **Monitoring > Sensor Monitoring > Support Information > System Information**.To get the same information from IME, choose **Configuration > *sensor_name* > Sensor Monitoring > Support Information > System Information**.

Note Always run the **show tech-support** command before contacting TAC.

For More Information

For the procedure for copying the output to a remote system, see Displaying Tech Support Information, page C-72.

Displaying Tech Support Information

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.

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.

```
sensor# show tech-support page
```

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.

Step 3 To send the output (in HTML format) to a file, follow these steps:

a. Enter the following command, followed by a valid destination.

```
sensor# show tech-support destination-url destination_url
```

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`.

For 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
```

The `password:` prompt appears.

b. Enter the password for this user account. The `Generating report:` message is displayed.

Tech Support Command Output

Note This output example shows the first part of the command and lists the information for the Interfaces, ARC, and cidDump services.

The following is an example of the **show tech-support** command output:

```
sensor# show tech-support page

System Status Report
This Report was generated on Mon Jun 23 19:49:30 2008.
Output from show version
Application Partition:

Cisco Intrusion Prevention System, Version 6.1(1)E2

Host:
    Realm Keys          key1.0
Signature Definition:
    Signature Update    S340.0                  2008-06-19
    Virus Update        V1.4                    2007-03-02
OS Version:             2.4.30-IDS-smp-bigphys
Platform:               ASA-SSM-20
```

```
Serial Number:          P300000220
Licensed, expires:      31-Dec-2009 UTC
Sensor up-time is 25 days.
Using 1052807168 out of 2093600768 bytes of available memory (50% usage)
system is using 17.7M out of 29.0M bytes of available disk space (61% usage)
application-data is using 41.1M out of 166.6M bytes of available disk space (26% usage)
boot is using 40.5M out of 68.5M bytes of available disk space (62% usage)


MainApp          M-2008_APR_24_19_16    (Release)   2008-04-24T19:49:05-0500   Running
AnalysisEngine   ME-2008_JUN_05_18_26   (Release)   2008-06-05T18:55:02-0500   Running
CLI              M-2008_APR_24_19_16    (Release)   2008-04-24T19:49:05-0500


Upgrade History:

* IPS-engine-E2-req-6.1-1   20:39:12 UTC Fri Jun 20 2008
  IPS-sig-S340-req-E2.pkg   20:42:45 UTC Fri Jun 20 2008

Recovery Partition Version 1.1 - 6.1(1)E1

Host Certificate Valid from: 28-May-2008 to 29-May-2010


Output from show interfaces
Interface Statistics
   Total Packets Received = 7561053
   Total Bytes Received = 620005608
   Missed Packet Percentage = 0
   Current Bypass Mode = Auto_off
MAC statistics from interface GigabitEthernet0/0
   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 = 7115688
   Total Bytes Received = 807518285
   Total Multicast Packets Received = 0
   Total Receive Errors = 0
   Total Receive FIFO Overruns = 0
   Total Packets Transmitted = 4988611
   Total Bytes Transmitted = 1004944745
   Total Transmit Errors = 0
   Total Transmit FIFO Overruns = 0
MAC statistics from interface GigabitEthernet0/1
   Interface function = Sensing interface
   Description =
   Media Type = backplane
   Default Vlan = 0
   Inline Mode = Unpaired
   Pair Status = N/A
   Hardware Bypass Capable = No
   Hardware Bypass Paired = N/A
   Link Status = Up
   Admin Enabled Status = Enabled
   Link Speed = Auto_1000
   Link Duplex = Auto_Full
   Missed Packet Percentage = 0
   Total Packets Received = 7561056
   Total Bytes Received = 620005854
   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 = 7561056
   Total Bytes Transmitted = 620006592
   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

Output from show statistics authentication
General
   totalAuthenticationAttempts = 1105
   failedAuthenticationAttempts = 5

Output from show statistics analysis-engine
Analysis Engine Statistics
   Number of seconds since service started = 256036
--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-75
- Displaying Version Information, page C-75

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.

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 6.1(1)E1

Host:
    Realm Keys          key1.0
Signature Definition:
    Signature Update    S323.0                  2008-03-24
    Virus Update        V1.2                    2005-11-24
OS Version:             2.4.30-IDS-smp-bigphys
Platform:               IPS-4240-K9
Serial Number:          P3000000652
No license present
Sensor up-time is 4 days.
Using 1421475840 out of 1984548864 bytes of available memory (71% usage)
system is using 17.7M out of 29.0M bytes of available disk space (61% usage)
application-data is using 41.0M out of 166.8M bytes of available disk space (26%
 usage)
boot is using 40.4M out of 68.6M bytes of available disk space (62% usage)


MainApp          M-2008_APR_16_21_44   (Release)   2008-04-16T22:25:36-0500   Running
AnalysisEngine   M-2008_APR_16_21_44   (Release)   2008-04-16T22:25:36-0500   Running
CLI              M-2008_APR_16_21_44   (Release)   2008-04-16T22:25:36-0500


Upgrade History:

  IPS-K9-6.1-1-E1   21:44:00 UTC Wed Apr 16 2008

Recovery Partition Version 1.1 - 6.1(1)E1

Host Certificate Valid from: 23-Apr-2008 to 24-Apr-2010

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 Thu Apr 24 16:21:25 2008
! ------------------------------
! Version 6.1(1)
! Host:
!     Realm Keys          key1.0
! Signature Definition:
!     Signature Update    S323.0   2008-03-24
!     Virus Update        V1.2     2005-11-24
```

```
! ------------------------------
service interface
exit
! ------------------------------
service authentication
exit
! ------------------------------
service event-action-rules rules0
exit
! ------------------------------
service host
network-settings
host-ip 10.89.147.45/25,10.89.147.126
telnet-option enabled
access-list 0.0.0.0/0
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 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

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 | authentication | event-server | event-store | external-product-interface |host | logger | network-access | notification | sdee-server | transaction-server | 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**] 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.

To display statistics for the sensor, follow these steps:

Step 1 Log in to the CLI.

Step 2 Display the statistics for Analysis Engine.

```
sensor# show statistics analysis-engine
Analysis Engine Statistics
   Number of seconds since service started = 1421127
   Measure of the level of current resource utilization = 0
   Measure of the level of maximum resource utilization = 0
   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 = 0
      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
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-4240#
```

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 Event Server.

```
sensor# show statistics event-server
General
   openSubscriptions = 0
   blockedSubscriptions = 0
Subscriptions
sensor#
```

Step 7 Display the statistics for Event Store.

```
sensor# show statistics event-store
Event 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 queries issued = 0
      The number of times the event store circular buffer has wrapped = 0
   Number of events of each type currently stored
      Debug events = 0
```

```
      Status events = 9904
      Log transaction events = 0
      Shun request events = 61
      Error events, warning = 67
      Error events, error = 83
      Error events, fatal = 0
      Alert events, informational = 60
      Alert events, low = 1
      Alert events, medium = 60
      Alert events, high = 0
sensor#
```

Step 8 Display the statistics for the host.

```
sensor# show statistics host
General Statistics
   Last Change To Host Config (UTC) = 16:11:05 Thu Feb 10 2008
   Command Control Port Device = FastEthernet0/0
Network Statistics
   fe0_0     Link encap:Ethernet HWaddr 00:0B:46:53:06:AA
             inet addr:10.89.149.185 Bcast:10.89.149.255 Mask:255.255.255.128
             UP BROADCAST RUNNING MULTICAST MTU:1500 Metric:1
             RX packets:1001522 errors:0 dropped:0 overruns:0 frame:0
             TX packets:469569 errors:0 dropped:0 overruns:0 carrier:0
             collisions:0 txqueuelen:1000
             RX bytes:57547021 (54.8 Mib) TX bytes:63832557 (60.8 MiB)
             Interrupt:9 Base address:0xf400 Memory:c0000000-c0000038
NTP Statistics
   status = Not applicable
Memory Usage
   usedBytes = 500592640
   freeBytes = 8855552
   totalBytes = 509448192
Swap Usage
   Used Bytes = 77824
   Free Bytes = 600649728

   Total Bytes = 600727552
CPU Statistics
   Usage over last 5 seconds = 0
   Usage over last minute = 1
   Usage over last 5 minutes = 1
Memory Statistics
   Memory usage (bytes) = 500498432
   Memory free (bytes) = 894976032
Auto Update Statistics
   lastDirectoryReadAttempt = N/A
   lastDownloadAttempt = N/A
   lastInstallAttempt = N/A
   nextAttempt = N/A
sensor#
```

Step 9 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 10 Display the statistics for 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 = 10.89.150.219
      NATAddr = 0.0.0.0
      Communications = ssh-des
   NetDevice
      Type = PIX
      IP = 10.89.150.250
      NATAddr = 0.0.0.0
      Communications = telnet
   NetDevice
      Type = Cisco
      IP = 10.89.150.158
      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 = 10.89.150.138
      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 = 10.89.150.171
      AclSupport = Does not use ACLs
      Version = 6.3
```

```
      State = Active
      Firewall-type = PIX
   NetDevice
      IP = 10.89.150.219
      AclSupport = Does not use ACLs
      Version = 7.0
      State = Active
      Firewall-type = ASA
   NetDevice
      IP = 10.89.150.250
      AclSupport = Does not use ACLs
      Version = 2.2
      State = Active
      Firewall-type = FWSM
   NetDevice
      IP = 10.89.150.158
      AclSupport = uses Named ACLs
      Version = 12.2
      State = Active
   NetDevice
      IP = 10.89.150.138
      AclSupport = Uses VACLs
      Version = 8.4
      State = Active
   BlockedAddr
      Host
         IP = 22.33.4.5
         Vlan =
         ActualIp =
         BlockMinutes =
      Host
         IP = 21.21.12.12
         Vlan =
         ActualIp =
         BlockMinutes =
      Host
         IP = 122.122.33.4
         Vlan =
         ActualIp =
         BlockMinutes = 60
         MinutesRemaining = 24
      Network
         IP = 111.22.0.0
         Mask = 255.255.0.0
         BlockMinutes =
sensor#
```

Step 11 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 12 Display the statistics for the SDEE server.

```
sensor# show statistics sdee-server
General
   Open Subscriptions = 0
   Blocked Subscriptions = 0
   Maximum Available Subscriptions = 5
```

```
   Maximum Events Per Retrieval = 500
Subscriptions
sensor#
```

Step 13 Display the statistics for the transaction server.

```
sensor# show statistics transaction-server
General
   totalControlTransactions = 35
   failedControlTransactions = 0
sensor#
```

Step 14 Display the statistics for a virtual sensor.

```
sensor# show statistics virtual-sensor vs0
   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 =
      General Statistics for this Virtual Sensor
         Number of seconds since a reset of the statistics = 1421711
         Measure of the level of resource utilization = 0
         Total packets processed since reset = 0
         Total IP 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 =
         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 packets with bad IP checksums processed since reset = 0
         Total packets with bad layer 4 checksums processed since reset = 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.


      The Signature Database Statistics.
         The Number of each type of node active in the system (can not be reset
            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 constraint
      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 dgrams limit since last reset = 0
   Fragments too small since last reset = 0
   Too many fragments per dgram 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
   Current Streams = 0
   Current Streams Closed = 0
   Current Streams Closing = 0
   Current Streams Embryonic = 0
   Current Streams Established = 0
   Current Streams Denied = 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
SigEvent Action Override Stage Statistics
   Number of Alerts received to Action Override Processor = 0
   Number of Alerts where an override was applied = 0
```

```
            Actions Added
               deny-attacker-inline = 0
               deny-attacker-victim-pair-inline = 0
               deny-attacker-service-pair-inline = 0
               deny-connection-inline = 0
               deny-packet-inline = 0
               modify-packet-inline = 0
               log-attacker-packets = 0
               log-pair-packets = 0
               log-victim-packets = 0
               produce-alert = 0
               produce-verbose-alert = 0
               request-block-connection = 0
               request-block-host = 0
               request-snmp-trap = 0
               reset-tcp-connection = 0
               request-rate-limit = 0
         SigEvent Action Filter Stage Statistics
            Number of Alerts received to Action Filter Processor = 0
            Number of Alerts where an action was filtered = 0
            Number of Filter Line matches = 0
            Number of Filter Line matches causing decreased DenyPercentage = 0
            Actions Filtered
               deny-attacker-inline = 0
               deny-attacker-victim-pair-inline = 0
               deny-attacker-service-pair-inline = 0
               deny-connection-inline = 0
               deny-packet-inline = 0
               modify-packet-inline = 0
               log-attacker-packets = 0
               log-pair-packets = 0
               log-victim-packets = 0
               produce-alert = 0
               produce-verbose-alert = 0
               request-block-connection = 0
               request-block-host = 0
               request-snmp-trap = 0
               reset-tcp-connection = 0
               request-rate-limit = 0
         SigEvent Action Handling Stage Statistics.
            Number of Alerts received to Action Handling Processor = 0
            Number of Alerts where produceAlert was forced = 0
            Number of Alerts where produceAlert was off = 0
            Actions Performed
               deny-attacker-inline = 0
               deny-attacker-victim-pair-inline = 0
               deny-attacker-service-pair-inline = 0
               deny-connection-inline = 0
               deny-packet-inline = 0
               modify-packet-inline = 0
               log-attacker-packets = 0
               log-pair-packets = 0
               log-victim-packets = 0
               produce-alert = 0
               produce-verbose-alert = 0
--MORE--
```

Step 15 Display the statistics for Web Server.

```
sensor# show statistics web-server
listener-443
   number of server session requests handled = 61
   number of server session requests rejected = 0
   total HTTP requests handled = 35
```

```
   maximum number of session objects allowed = 40
   number of idle allocated session objects = 10
   number of busy allocated session objects = 0
crypto library version = 6.0.3
sensor#
```

Step 16 To clear the statistics for an application, for example, the logging application.

```
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
```

The statistics were retrieved and cleared.

Step 17 Verify that the statistics have been cleared.

```
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#
```

The statistics all begin from 0.

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

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#
```

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-89
- Understanding the show events Command, page C-89
- Displaying Events, page C-90
- Clearing Events, page C-93

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

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 except for AIP SSC-5, which has a fixed size of 10 MB.

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] | **log** | **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.

Note Events are displayed as a live feed. To cancel the request, press **Ctrl-C.**

The following options 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 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**—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.
- **log**—Displays log events. Log events are generated when a transaction is received and responded to by an application. Contains information about the request, response, success or failure of the transaction.
- **NAC**—Displays ARC (block) requests.

  Note ARC is formerly known as NAC. This name change has not been completely implemented throughout IDM, IME, and the CLI for Cisco IPS 6.1.
- **status**—Displays status events.
- **past**—Displays events starting in the past for the specified hours, minutes, and seconds.
- *hh:mm:ss*—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**.

To display events from Event Store, follow these steps:

Step 1 Log in to the CLI.

Step 2 Display all events starting now.

```
sensor# show events
evError: eventId=1041472274774840147 severity=warning vendor=Cisco
  originator:
    hostId: sensor2
    appName: cidwebserver
    appInstanceId: 12075
  time: 2008/01/07 04:41:45 2008/01/07 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: 2008/01/07 04:41:45 2008/01/07 04:41:45 UTC
  errorMessage: name=errTransport WebSession::sessionTask(6) TLS connection exception:
handshake incomplete.
```

The feed continues showing all events until you press **Ctrl-C**.

Step 3 Display the block requests beginning at 10:00 a.m. on February 9, 2008.

```
sensor# show events NAC 10:00:00 Feb 9 2008
evShunRqst: eventId=1106837332219222281 vendor=Cisco
  originator:
    deviceName: Sensor1
    appName: NetworkAccessControllerApp
    appInstance: 654
  time: 2008/02/09 10:33:31 2008/08/09 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, 2008.

```
sensor# show events error warning 10:00:00 Feb 9 2008
evError: eventId=1041472274774840197 severity=warning vendor=Cisco
  originator:
    hostId: sensor
    appName: cidwebserver
    appInstanceId: 12160
  time: 2008/01/07 04:49:25 2008/01/07 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: 2008/03/02 14:15:59 2008/03/02 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: 2008/01/08 02:41:00 2008/01/08 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: 2008/01/08 02:41:00 2008/01/08 02:41:00 UTC
  syslogMessage:
    description: session opened for user cisco by cisco(uid=0)
```

Clearing Events

Use the **clear events** command to clear Event Store.

To clear events from Event Store, follow these steps:

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

Step 2 Clear 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 IDM, 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 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-94.

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.

APPENDIX D

Open Source License Files

The copyrights for certain works contained herein are owned by other third parties and are used and distributed under license. Some parts of this software are covered under the GNU public license. If you would like to get a copy of the source code for those portions of this software that are covered by the GNU public license, please send an e-mail to ips-opensource-request@cisco.com.

Some components of this product may be covered under one or more of the open source licenses printed below. However, the Cisco warranty for the product shall remain in effect to its full extent and shall apply to the entire product.

This section contains the following topics:

Artistic License

This document is freely plagiarized from the 'Artistic Licence', distributed as part of the Perl v4.0 kit by Larry Wall, which is available from most major archive sites.

This documents purpose is to state the conditions under which these Packages (See definition below) viz: "Crack", the Unix Password Cracker, and "CrackLib", the Unix Password Checking library, which are held in copyright by Alec David Edward Muffett, may be copied, such that the copyright holder maintains some semblance of artistic control over the development of the packages, while giving the users of the package the right to use and distribute the Package in a more-or-less customary fashion, plus the right to make reasonable modifications.

So there.

Definitions:

A "Package" refers to the collection of files distributed by the Copyright Holder, and derivatives of that collection of files created through textual modification, or segments thereof.

"Standard Version" refers to such a Package if it has not been modified, or has been modified in accordance with the wishes of the Copyright Holder.

"Copyright Holder" is whoever is named in the copyright or copyrights for the package.

"You" is you, if you're thinking about copying or distributing this Package.

"Reasonable copying fee" is whatever you can justify on the basis of media cost, duplication charges, time of people involved, and so on. (You will not be required to justify it to the Copyright Holder, but only to the computing community at large as a market that must bear the fee.)

"Freely Available" means that no fee is charged for the item itself, though there may be fees involved in handling the item. It also means that recipients of the item may redistribute it under the same conditions they received it.

1. You may make and give away verbatim copies of the source form of the Standard Version of this Package without restriction, provided that you duplicate all of the original copyright notices and associated disclaimers.
2. You may apply bug fixes, portability fixes and other modifications derived from the Public Domain or from the Copyright Holder. A Package modified in such a way shall still be considered the Standard Version.
3. You may otherwise modify your copy of this Package in any way, provided that you insert a prominent notice in each changed file stating how and when AND WHY you changed that file, and provided that you do at least ONE of the following:
   a. place your modifications in the Public Domain or otherwise make them Freely Available, such as by posting said modifications to Usenet or an equivalent medium, or placing the modifications on a major archive site such as uunet.uu.net, or by allowing the Copyright Holder to include your modifications in the Standard Version of the Package.
   b. use the modified Package only within your corporation or organization.
   c. rename any non-standard executables so the names do not conflict with standard executables, which must also be provided, and provide separate documentation for each non-standard executable that clearly documents how it differs from the Standard Version.
   d. make other distribution arrangements with the Copyright Holder.

4. You may distribute the programs of this Package in object code or executable form, provided that you do at least ONE of the following:
   a. distribute a Standard Version of the executables and library files, together with instructions (in the manual page or equivalent) on where to get the Standard Version.
   b. accompany the distribution with the machine-readable source of the Package with your modifications.
   c. accompany any non-standard executables with their corresponding Standard Version executables, giving the non-standard executables non-standard names, and clearly documenting the differences in manual pages (or equivalent), together with instructions on where to get the Standard Version.
   d. make other distribution arrangements with the Copyright Holder.
5. You may charge a reasonable copying fee for any distribution of this Package. You may charge any fee you choose for support of this Package. YOU MAY NOT CHARGE A FEE FOR THIS PACKAGE ITSELF. However, you may distribute this Package in aggregate with other (possibly commercial) programs as part of a larger (possibly commercial) software distribution provided that YOU DO NOT ADVERTISE this package as a product of your own.
6. The name of the Copyright Holder may not be used to endorse or promote products derived from this software without specific prior written permission.
7. THIS PACKAGE IS PROVIDED “AS IS” AND WITHOUT ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, WITHOUT LIMITATION, THE IMPLIED WARRANTIES OF MERCHANTIBILITY AND FITNESS FOR A PARTICULAR PURPOSE.

   The End

BSD 1.0 License

This file is part of the OpenSSH software.

The licences which components of this software fall under are as follows. First, we will summarize and say that all components are under a BSD licence, or a licence more free than that.

OpenSSH contains no GPL code.

1. Copyright (c) 1995 Tatu Ylonen <ylo@cs.hut.fi>, Espoo, Finland

   All rights reserved

   As far as I am concerned, the code I have written for this software can be used freely for any purpose. Any derived versions of this software must be clearly marked as such, and if the derived work is incompatible with the protocol description in the RFC file, it must be called by a name other than “ssh” or “Secure Shell”.

   [Tatu continues]

   However, I am not implying to give any licenses to any patents or copyrights held by third parties, and the software includes parts that are not under my direct control. As far as I know, all included source code is used in accordance with the relevant license agreements and can be used freely for any purpose (the GNU license being the most restrictive); see below for details.

   [However, none of that term is relevant at this point in time. All of these restrictively licensed software components which he talks about have been removed from OpenSSH, i.e.,

   - RSA is no longer included, found in the OpenSSL library
   - IDEA is no longer included, its use is deprecated

- DES is now external, in the OpenSSL library
- GMP is no longer used, and instead we call BN code from OpenSSL
- Zlib is now external, in a library
- The make-ssh-known-hosts script is no longer included
- TSS has been removed
- MD5 is now external, in the OpenSSL library
- RC4 support has been replaced with ARC4 support from OpenSSL
- Blowfish is now external, in the OpenSSL library

[The licence continues]

Note that any information and cryptographic algorithms used in this software are publicly available on the Internet and at any major bookstore, scientific library, and patent office worldwide. More information can be found e.g. at "http://www.cs.hut.fi/crypto".

The legal status of this program is some combination of all these permissions and restrictions. Use only at your own responsibility. You will be responsible for any legal consequences yourself; I am not making any claims whether possessing or using this is legal or not in your country, and I am not taking any responsibility on your behalf.

NO WARRANTY

BECAUSE THE PROGRAM IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY FOR THE PROGRAM, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE PROGRAM "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE PROGRAM IS WITH YOU. SHOULD THE PROGRAM PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY AND/OR REDISTRIBUTE THE PROGRAM AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE PROGRAM (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE PROGRAM TO OPERATE WITH ANY OTHER PROGRAMS), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

2. The 32-bit CRC implementation in crc32.c is due to Gary S. Brown. Comments in the file indicate it may be used for any purpose without restrictions:

   COPYRIGHT (C) 1986 Gary S. Brown. You may use this program, or code or tables extracted from it, as desired without restriction.

3. The 32-bit CRC compensation attack detector in deattack.c was contributed by CORE SDI S.A. under a BSD-style license.

   Cryptographic attack detector for ssh - source code

   Copyright (c) 1998 CORE SDI S.A., Buenos Aires, Argentina.

   All rights reserved. Redistribution and use in source and binary forms, with or without modification, are permitted provided that this copyright notice is retained.

THIS SOFTWARE IS PROVIDED “AS IS” AND ANY EXPRESS OR IMPLIED WARRANTIES ARE DISCLAIMED. IN NO EVENT SHALL CORE SDI S.A. BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY OR CONSEQUENTIAL DAMAGES RESULTING FROM THE USE OR MISUSE OF THIS SOFTWARE.

Ariel Futoransky <futo@core-sdi.com> <http://www.core-sdi.com>

4. ssh-keygen was contributed by David Mazieres under a BSD-style license.

   Copyright 1995, 1996 by David Mazieres <dm@lcs.mit.edu>.

   Modification and redistribution in source and binary forms is permitted provided that due credit is given to the author and the OpenBSD project by leaving this copyright notice intact.

5. The Rijndael implementation by Vincent Rijmen, Antoon Bosselaers and Paulo Barreto is in the public domain and distributed with the following license:

   @version 3.0 (December 2000)

   Optimized ANSI C code for the Rijndael cipher (now AES)

   @author Vincent Rijmen <vincent.rijmen@esat.kuleuven.ac.be>

   @author Antoon Bosselaers <antoon.bosselaers@esat.kuleuven.ac.be>

   @author Paulo Barreto <paulo.barreto@terra.com.br>

   This code is hereby placed in the public domain.

   THIS SOFTWARE IS PROVIDED BY THE AUTHORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE AUTHORS OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

6. One component of the ssh source code is under a 4-clause BSD license, held by the University of California, since we pulled these parts from original Berkeley code. The Regents of the University of California have declared that term 3 is no longer enforceable on their source code, but we retain that license as is.

7. Copyright (c) 1983, 1990, 1992, 1993, 1995

8. The Regents of the University of California. All rights reserved.

9. Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

   a. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

   b. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

   c. All advertising materials mentioning features or use of this software must display the following acknowledgement: This product includes software developed by the University of California, Berkeley and its contributors.

d. Neither the name of the University nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE REGENTS AND CONTRIBUTORS “AS IS” AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE REGENTS OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

10. Remaining components of the software are provided under a standard 2-term BSD licence with the following names as copyright holders:

Markus Friedl

Theo de Raadt

Niels Provos

Dug Song

Aaron Campbell

Damien Miller

Kevin Steves

Daniel Kouril

Per Allansson

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. 1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
2. 2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

THIS SOFTWARE IS PROVIDED BY THE AUTHOR ``AS IS'' AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

BusyBox License

--- A note on GPL versions

BusyBox is distributed under version 2 of the General Public License (included in its entirety, below). Version 2 is the only version of this license which this version of BusyBox (or modified versions derived from this one) may be distributed under.

GNU GENERAL PUBLIC LICENSE

Version 2, June 1991

Copyright (C) 1989, 1991 Free Software Foundation, Inc.

59 Temple Place, Suite 330, Boston, MA 02111-1307 USA

Everyone is permitted to copy and distribute verbatim copies of this license document, but changing it is not allowed.

Preamble

The licenses for most software are designed to take away your freedom to share and change it. By contrast, the GNU General Public License is intended to guarantee your freedom to share and change free software--to make sure the software is free for all its users. This General Public License applies to most of the Free Software Foundation's software and to any other program whose authors commit to using it. (Some other Free Software Foundation software is covered by the GNU Library General Public License instead.) You can apply it to your programs, too.

When we speak of free software, we are referring to freedom, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for this service if you wish), that you receive source code or can get it if you want it, that you can change the software or use pieces of it in new free programs; and that you know you can do these things.

To protect your rights, we need to make restrictions that forbid anyone to deny you these rights or to ask you to surrender the rights. These restrictions translate to certain responsibilities for you if you distribute copies of the software, or if you modify it.

For example, if you distribute copies of such a program, whether gratis or for a fee, you must give the recipients all the rights that you have. You must make sure that they, too, receive or can get the source code. And you must show them these terms so they know their rights.

We protect your rights with two steps: (1) copyright the software, and (2) offer you this license which gives you legal permission to copy, distribute and/or modify the software.

Also, for each author's protection and ours, we want to make certain that everyone understands that there is no warranty for this free software. If the software is modified by someone else and passed on, we want its recipients to know that what they have is not the original, so that any problems introduced by others will not reflect on the original authors' reputations.

Finally, any free program is threatened constantly by software patents. We wish to avoid the danger that redistributors of a free program will individually obtain patent licenses, in effect making the program proprietary. To prevent this, we have made it clear that any patent must be licensed for everyone's free use or not licensed at all.

The precise terms and conditions for copying, distribution and modification follow.

GNU GENERAL PUBLIC LICENSE

TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION

0. This License applies to any program or other work which contains a notice placed by the copyright holder saying it may be distributed under the terms of this General Public License. The "Program", below, refers to any such program or work, and a "work based on the Program" means either the Program or any derivative work under copyright law: that is to say, a work containing the Program or a portion of it, either verbatim or with modifications and/or translated into another language. (Hereinafter, translation is included without limitation in the term "modification".) Each licensee is addressed as "you".

Activities other than copying, distribution and modification are not covered by this License; they are outside its scope. The act of running the Program is not restricted, and the output from the Program is covered only if its contents constitute a work based on the Program (independent of having been made by running the Program). Whether that is true depends on what the Program does.

1. You may copy and distribute verbatim copies of the Program's source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice and disclaimer of warranty; keep intact all the notices that refer to this License and to the absence of any warranty; and give any other recipients of the Program a copy of this License along with the Program.

You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty protection in exchange for a fee.

2. You may modify your copy or copies of the Program or any portion of it, thus forming a work based on the Program, and copy and distribute such modifications or work under the terms of Section 1 above, provided that you also meet all of these conditions:

a) You must cause the modified files to carry prominent notices stating that you changed the files and the date of any change.

b) You must cause any work that you distribute or publish, that in whole or in part contains or is derived from the Program or any part thereof, to be licensed as a whole at no charge to all third parties under the terms of this License.

c) If the modified program normally reads commands interactively when run, you must cause it, when started running for such interactive use in the most ordinary way, to print or display an announcement including an appropriate copyright notice and a notice that there is no warranty (or else, saying that you provide a warranty) and that users may redistribute the program under these conditions, and telling the user how to view a copy of this License. (Exception: if the Program itself is interactive but does not normally print such an announcement, your work based on the Program is not required to print an announcement.)

These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived from the Program, and can be reasonably considered independent and separate works in themselves, then this License, and its terms, do not apply to those sections when you distribute them as separate works. But when you distribute the same sections as part of a whole which is a work based on the Program, the distribution of the whole must be on the terms of this License, whose permissions for other licensees extend to the entire whole, and thus to each and every part regardless of who wrote it.

Thus, it is not the intent of this section to claim rights or contest your rights to work written entirely by you; rather, the intent is to exercise the right to control the distribution of derivative or collective works based on the Program.

In addition, mere aggregation of another work not based on the Program with the Program (or with a work based on the Program) on a volume of a storage or distribution medium does not bring the other work under the scope of this License.

3. You may copy and distribute the Program (or a work based on it, under Section 2) in object code or executable form under the terms of Sections 1 and 2 above provided that you also do one of the following:

a) Accompany it with the complete corresponding machine-readable source code, which must be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,

b) Accompany it with a written offer, valid for at least three years, to give any third party, for a charge no more than your cost of physically performing source distribution, a complete machine-readable copy of the corresponding source code, to be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,

c) Accompany it with the information you received as to the offer to distribute corresponding source code. (This alternative is allowed only for noncommercial distribution and only if you received the program in object code or executable form with such an offer, in accord with Subsection b above.)

The source code for a work means the preferred form of the work for making modifications to it. For an executable work, complete source code means all the source code for all modules it contains, plus any associated interface definition files, plus the scripts used to control compilation and installation of the executable. However, as a special exception, the source code distributed need not include anything that is normally distributed (in either source or binary form) with the major components (compiler, kernel, and so on) of the operating system on which the executable runs, unless that component itself accompanies the executable.

If distribution of executable or object code is made by offering access to copy from a designated place, then offering equivalent access to copy the source code from the same place counts as distribution of the source code, even though third parties are not compelled to copy the source along with the object code.

4. You may not copy, modify, sublicense, or distribute the Program except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense or distribute the Program is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.

5. You are not required to accept this License, since you have not signed it. However, nothing else grants you permission to modify or distribute the Program or its derivative works. These actions are prohibited by law if you do not accept this License. Therefore, by modifying or distributing the Program (or any work based on the Program), you indicate your acceptance of this License to do so, and all its terms and conditions for copying, distributing or modifying the Program or works based on it.

6. Each time you redistribute the Program (or any work based on the Program), the recipient automatically receives a license from the original licensor to copy, distribute or modify the Program subject to these terms and conditions. You may not impose any further restrictions on the recipients' exercise of the rights granted herein. You are not responsible for enforcing compliance by third parties to this License.

7. If, as a consequence of a court judgment or allegation of patent infringement or for any other reason (not limited to patent issues), conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot distribute so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not distribute the Program at all. For example, if a patent license would not permit royalty-free redistribution of the Program by all those who receive copies directly or indirectly through you, then the only way you could satisfy both it and this License would be to refrain entirely from distribution of the Program.

If any portion of this section is held invalid or unenforceable under any particular circumstance, the balance of the section is intended to apply and the section as a whole is intended to apply in other circumstances.

It is not the purpose of this section to induce you to infringe any patents or other property right claims or to contest validity of any such claims; this section has the sole purpose of protecting the integrity of the free software distribution system, which is implemented by public license practices. Many people have made generous contributions to the wide range of software distributed through that system in reliance on consistent application of that system; it is up to the author/donor to decide if he or she is willing to distribute software through any other system and a licensee cannot impose that choice.

This section is intended to make thoroughly clear what is believed to be a consequence of the rest of this License.

8. If the distribution and/or use of the Program is restricted in certain countries either by patents or by copyrighted interfaces, the original copyright holder who places the Program under this License may add an explicit geographical distribution limitation excluding those countries, so that distribution is permitted only in or among countries not thus excluded. In such case, this License incorporates the limitation as if written in the body of this License.

9. The Free Software Foundation may publish revised and/or new versions of the General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Program specifies a version number of this License which applies to it and "any later version", you have the option of following the terms and conditions either of that version or of any later version published by the Free Software Foundation. If the Program does not specify a version number of this License, you may choose any version ever published by the Free Software Foundation.

10. If you wish to incorporate parts of the Program into other free programs whose distribution conditions are different, write to the author to ask for permission. For software which is copyrighted by the Free Software Foundation, write to the Free Software Foundation; we sometimes make exceptions for this. Our decision will be guided by the two goals of preserving the free status of all derivatives of our free software and of promoting the sharing and reuse of software generally.

NO WARRANTY

11. BECAUSE THE PROGRAM IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY FOR THE PROGRAM, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE PROGRAM "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK ASTO THE QUALITY AND PERFORMANCE OF THE PROGRAM IS WITH YOU. SHOULD THE PROGRAM PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

12. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY AND/ORREDISTRIBUTE THE PROGRAM AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE PROGRAM (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE PROGRAM TO OPERATE WITH ANY OTHER PROGRAMS), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

END OF TERMS AND CONDITIONS

How to Apply These Terms to Your New Programs

If you develop a new program, and you want it to be of the greatest possible use to the public, the best way to achieve this is to make it free software which everyone can redistribute and change under these terms.

To do so, attach the following notices to the program. It is safest to attach them to the start of each source file to most effectively convey the exclusion of warranty; and each file should have at least the "copyright" line and a pointer to where the full notice is found.

<one line to give the program's name and a brief idea of what it does.>

Copyright (C) <year> <name of author>

This program is free software; you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation; either version 2 of the License, or (at your option) any later version.

This program is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

You should have received a copy of the GNU General Public License along with this program; if not, write to the Free Software Foundation, Inc., 59 Temple Place, Suite 330, Boston, MA 02111-1307 USA

Also add information on how to contact you by electronic and paper mail.

If the program is interactive, make it output a short notice like this when it starts in an interactive mode:

Gnomovision version 69, Copyright (C) year name of author

Gnomovision comes with ABSOLUTELY NO WARRANTY; for details type `show w'.

This is free software, and you are welcome to redistribute it

under certain conditions; type `show c' for details.

The hypothetical commands `show w' and `show c' should show the appropriate parts of the General Public License. Of course, the commands you use may be called something other than `show w' and `show c'; they could even be mouse-clicks or menu items--whatever suits your program.

You should also get your employer (if you work as a programmer) or your school, if any, to sign a "copyright disclaimer" for the program, if necessary. Here is a sample; alter the names:

Yoyodyne, Inc., hereby disclaims all copyright interest in the program

`Gnomovision' (which makes passes at compilers) written by James Hacker.

<signature of Ty Coon>, 1 April 1989

Ty Coon, President of Vice

This General Public License does not permit incorporating your program into proprietary programs. If your program is a subroutine library, you may consider it more useful to permit linking proprietary applications with the library. If this is what you want to do, use the GNU Library General Public License instead of this License.

Curl License

COPYRIGHT AND PERMISSION NOTICE

Copyright (c) 1996 - 2003, Daniel Stenberg, <daniel@haxx.se>.

All rights reserved.

Permission to use, copy, modify, and distribute this software for any purpose with or without fee is hereby granted, provided that the above copyright notice and this permission notice appear in all copies.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT OF THIRD PARTY RIGHTS. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Except as contained in this notice, the name of a copyright holder shall not be used in advertising or otherwise to promote the sale, use or other dealings in this Software without prior written authorization of the copyright holder

expat License

Copyright (c) 1998, 1999, 2000 Thai Open Source Software Center Ltd and Clark Cooper Copyright (c) 2001, 2002 Expat maintainers.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

GNU Free Documentation License

Version 1.1, March 2000

Copyright (C) 2000 Free Software Foundation, Inc. 59 Temple Place, Suite 330, Boston, MA 02111-1307 USA

Everyone is permitted to copy and distribute verbatim copies of this license document, but changing it is not allowed.

0. PREAMBLE

The purpose of this License is to make a manual, textbook, or other written document "free" in the sense of freedom: to assure everyone the effective freedom to copy and redistribute it, with or without modifying it, either commercially or noncommercially. Secondarily, this License preserves for the author and publisher a way to get credit for their work, while not being considered responsible for modifications made by others.

This License is a kind of “copyleft”, which means that derivative works of the document must themselves be free in the same sense. It complements the GNU General Public License, which is a copyleft license designed for free software.

We have designed this License in order to use it for manuals for free software, because free software needs free documentation: a free program should come with manuals providing the same freedoms that the software does. But this License is not limited to software manuals; it can be used for any textual work, regardless of subject matter or whether it is published as a printed book. We recommend this License principally for works whose purpose is instruction or reference.

1. APPLICABILITY AND DEFINITIONS

This License applies to any manual or other work that contains a notice placed by the copyright holder saying it can be distributed under the terms of this License. The “Document”, below, refers to any such manual or work. Any member of the public is a licensee, and is addressed as “you”.

A “Modified Version” of the Document means any work containing the Document or a portion of it, either copied verbatim, or with modifications and/or translated into another language.

A “Secondary Section” is a named appendix or a front-matter section of the Document that deals exclusively with the relationship of the publishers or authors of the Document to the Document's overall subject (or to related matters) and contains nothing that could fall directly within that overall subject. (For example, if the Document is in part a textbook of mathematics, a Secondary Section may not explain any mathematics.) The relationship could be a matter of historical connection with the subject or with related matters, or of legal, commercial, philosophical, ethical or political position regarding them.

The “Invariant Sections” are certain Secondary Sections whose titles are designated, as being those of Invariant Sections, in the notice that says that the Document is released under this License.

The “Cover Texts” are certain short passages of text that are listed, as Front-Cover Texts or Back-Cover Texts, in the notice that says that the Document is released under this License.

A “Transparent” copy of the Document means a machine-readable copy, represented in a format whose specification is available to the general public, whose contents can be viewed and edited directly and straightforwardly with generic text editors or (for images composed of pixels) generic paint programs or (for drawings) some widely available drawing editor, and that is suitable for input to text formatters or

for automatic translation to a variety of formats suitable for input to text formatters. A copy made in an otherwise Transparent file format whose markup has been designed to thwart or discourage subsequent modification by readers is not Transparent. A copy that is not “Transparent” is called “Opaque”.

Examples of suitable formats for Transparent copies include plain ASCII without markup, Texinfo input format, LaTeX input format, SGML or XML using a publicly available DTD, and standard-conforming simple HTML designed for human modification. Opaque formats include PostScript, PDF, proprietary formats that can be read and edited only by proprietary word processors, SGML or XML for which the DTD and/or processing tools are not generally available, and the machine-generated HTML produced by some word processors for output purposes only.

The “Title Page” means, for a printed book, the title page itself, plus such following pages as are needed to hold, legibly, the material this License requires to appear in the title page. For works in formats which do not have any title page as such, “Title Page” means the text near the most prominent appearance of the work's title, preceding the beginning of the body of the text.

2. VERBATIM COPYING

You may copy and distribute the Document in any medium, either commercially or noncommercially, provided that this License, the copyright notices, and the license notice saying this License applies to the Document are reproduced in all copies, and that you add no other conditions whatsoever to those of this License. You may not use technical measures to obstruct or control the reading or further copying of the copies you make or distribute. However, you may accept compensation in exchange for copies. If you distribute a large enough number of copies you must also follow the conditions in section 3.

You may also lend copies, under the same conditions stated above, and you may publicly display copies.

3. COPYING IN QUANTITY

If you publish printed copies of the Document numbering more than 100, and the Document's license notice requires Cover Texts, you must enclose the copies in covers that carry, clearly and legibly, all these Cover Texts: Front-Cover Texts on the front cover, and Back-Cover Texts on the back cover. Both covers must also clearly and legibly identify you as the publisher of these copies. The front cover must present the full title with all words of the title equally prominent and visible. You may add other material on the covers in addition. Copying with changes limited to the covers, as long as they preserve the title of the Document and satisfy these conditions, can be treated as verbatim copying in other respects.

If the required texts for either cover are too voluminous to fit legibly, you should put the first ones listed (as many as fit reasonably) on the actual cover, and continue the rest onto adjacent pages.

If you publish or distribute Opaque copies of the Document numbering more than 100, you must either include a machine-readable Transparent copy along with each Opaque copy, or state in or with each Opaque copy a publicly-accessible computer-network location containing a complete Transparent copy of the Document, free of added material, which the general network-using public has access to download anonymously at no charge using public-standard network protocols. If you use the latter option, you must take reasonably prudent steps, when you begin distribution of Opaque copies in quantity, to ensure that this Transparent copy will remain thus accessible at the stated location until at least one year after the last time you distribute an Opaque copy (directly or through your agents or retailers) of that edition to the public.

It is requested, but not required, that you contact the authors of the Document well before redistributing any large number of copies, to give them a chance to provide you with an updated version of the Document.

4. MODIFICATIONS

You may copy and distribute a Modified Version of the Document under the conditions of sections 2 and 3 above, provided that you release the Modified Version under precisely this License, with the Modified Version filling the role of the Document, thus licensing distribution and modification of the Modified Version to whoever possesses a copy of it. In addition, you must do these things in the Modified Version:

a. Use in the Title Page (and on the covers, if any) a title distinct from that of the Document, and from those of previous versions (which should, if there were any, be listed in the History section of the Document). You may use the same title as a previous version if the original publisher of that version gives permission.

b. List on the Title Page, as authors, one or more persons or entities responsible for authorship of the modifications in the Modified Version, together with at least five of the principal authors of the Document (all of its principal authors, if it has less than five).

c. State on the Title page the name of the publisher of the Modified Version, as the publisher.

d. Preserve all the copyright notices of the Document.

e. Add an appropriate copyright notice for your modifications adjacent to the other copyright notices.

f. Include, immediately after the copyright notices, a license notice giving the public permission to use the Modified Version under the terms of this License, in the form shown in the Addendum below.

g. Preserve in that license notice the full lists of Invariant Sections and required Cover Texts given in the Document's license notice.

h. Include an unaltered copy of this License.

i. Preserve the section entitled “History”, and its title, and add to it an item stating at least the title, year, new authors, and publisher of the Modified Version as given on the Title Page. If there is no section entitled “History” in the Document, create one stating the title, year, authors, and publisher of the Document as given on its Title Page, then add an item describing the Modified Version as stated in the previous sentence.

j. Preserve the network location, if any, given in the Document for public access to a Transparent copy of the Document, and likewise the network locations given in the Document for previous versions it was based on. These may be placed in the “History” section. You may omit a network location for a work that was published at least four years before the Document itself, or if the original publisher of the version it refers to gives permission.

k. In any section entitled “Acknowledgements” or “Dedications”, preserve the section’s title, and preserve in the section all the substance and tone of each of the contributor acknowledgements and/or dedications given therein.

l. Preserve all the Invariant Sections of the Document, unaltered in their text and in their titles. Section numbers or the equivalent are not considered part of the section titles.

m. Delete any section entitled “Endorsements”. Such a section may not be included in the Modified Version.

n. Do not retitle any existing section as “Endorsements” or to conflict in title with any Invariant Section.

If the Modified Version includes new front-matter sections or appendices that qualify as Secondary Sections and contain no material copied from the Document, you may at your option designate some or all of these sections as invariant. To do this, add their titles to the list of Invariant Sections in the Modified Version's license notice. These titles must be distinct from any other section titles.

You may add a section entitled “Endorsements”, provided it contains nothing but endorsements of your Modified Version by various parties--for example, statements of peer review or that the text has been approved by an organization as the authoritative definition of a standard.

You may add a passage of up to five words as a Front-Cover Text, and a passage of up to 25 words as a Back-Cover Text, to the end of the list of Cover Texts in the Modified Version. Only one passage of Front-Cover Text and one of Back-Cover Text may be added by (or through arrangements made by) any one entity. If the Document already includes a cover text for the same cover, previously added by you or by arrangement made by the same entity you are acting on behalf of, you may not add another; but you may replace the old one, on explicit permission from the previous publisher that added the old one.

The author(s) and publisher(s) of the Document do not by this License give permission to use their names for publicity for or to assert or imply endorsement of any Modified Version.

5. COMBINING DOCUMENTS

You may combine the Document with other documents released under this License, under the terms defined in section 4 above for modified versions, provided that you include in the combination all of the Invariant Sections of all of the original documents, unmodified, and list them all as Invariant Sections of your combined work in its license notice.

The combined work need only contain one copy of this License, and multiple identical Invariant Sections may be replaced with a single copy. If there are multiple Invariant Sections with the same name but different contents, make the title of each such section unique by adding at the end of it, in parentheses, the name of the original author or publisher of that section if known, or else a unique number. Make the same adjustment to the section titles in the list of Invariant Sections in the license notice of the combined work.

In the combination, you must combine any sections entitled “History” in the various original documents, forming one section entitled “History”; likewise combine any sections entitled “Acknowledgements”, and any sections entitled “Dedications”. You must delete all sections entitled “Endorsements.”

6. COLLECTIONS OF DOCUMENTS

You may make a collection consisting of the Document and other documents released under this License, and replace the individual copies of this License in the various documents with a single copy that is included in the collection, provided that you follow the rules of this License for verbatim copying of each of the documents in all other respects.

You may extract a single document from such a collection, and distribute it individually under this License, provided you insert a copy of this License into the extracted document, and follow this License in all other respects regarding verbatim copying of that document.

7. AGGREGATION WITH INDEPENDENT WORKS

A compilation of the Document or its derivatives with other separate and independent documents or works, in or on a volume of a storage or distribution medium, does not as a whole count as a Modified Version of the Document, provided no compilation copyright is claimed for the compilation. Such a compilation is called an “aggregate”, and this License does not apply to the other self-contained works thus compiled with the Document, on account of their being thus compiled, if they are not themselves derivative works of the Document.

If the Cover Text requirement of section 3 is applicable to these copies of the Document, then if the Document is less than one quarter of the entire aggregate, the Document's Cover Texts may be placed on covers that surround only the Document within the aggregate. Otherwise they must appear on covers around the whole aggregate.

8. TRANSLATION

Translation is considered a kind of modification, so you may distribute translations of the Document under the terms of section 4. Replacing Invariant Sections with translations requires special permission from their copyright holders, but you may include translations of some or all Invariant Sections in addition to the original versions of these Invariant Sections. You may include a translation of this License provided that you also include the original English version of this License. In case of a disagreement between the translation and the original English version of this License, the original English version will prevail.

9. TERMINATION

You may not copy, modify, sublicense, or distribute the Document except as expressly provided for under this License. Any other attempt to copy, modify, sublicense or distribute the Document is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.

10. FUTURE REVISIONS OF THIS LICENSE

The Free Software Foundation may publish new, revised versions of the GNU Free Documentation License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns. See http://www.gnu.org/copyleft/.

Each version of the License is given a distinguishing version number. If the Document specifies that a particular numbered version of this License “or any later version” applies to it, you have the option of following the terms and conditions either of that specified version or of any later version that has been published (not as a draft) by the Free Software Foundation. If the Document does not specify a version number of this License, you may choose any version ever published (not as a draft) by the Free Software Foundation.

ADDENDUM: How to use this License for your documents

To use this License in a document you have written, include a copy of the License in the document and put the following copyright and license notices just after the title page:

> Copyright (c) YEAR YOUR NAME.
>
> Permission is granted to copy, distribute and/or modify this document under the terms of the GNU Free Documentation License, Version 1.1 or any later version published by the Free Software Foundation; with the Invariant Sections being LIST THEIR TITLES, with the Front-Cover Texts being LIST, and with the Back-Cover Texts being LIST. A copy of the license is included in the section entitled "GNU Free Documentation License".

If you have no Invariant Sections, write "with no Invariant Sections" instead of saying which ones are invariant. If you have no Front-Cover Texts, write "no Front-Cover Texts" instead of "Front-Cover Texts being LIST"; likewise for Back-Cover Texts.

If your document contains nontrivial examples of program code, we recommend releasing these examples in parallel under your choice of free software license, such as the GNU General Public License, to permit their use in free software.

The GNU General Public License (GPL)

Version 2, June 1991

Copyright (C) 1989, 1991 Free Software Foundation, Inc. 59 Temple Place, Suite 330, Boston, MA 02111-1307 USA

Everyone is permitted to copy and distribute verbatim copies of this license document, but changing it is not allowed.

Machine readable copies of modifications made by Cisco to open source code under GPL are available from Cisco upon request.

Preamble

The licenses for most software are designed to take away your freedom to share and change it. By contrast, the GNU General Public License is intended to guarantee your freedom to share and change free software—to make sure the software is free for all its users. This General Public License applies to most of the Free Software Foundation's software and to any other program whose authors commit to using it. (Some other Free Software Foundation software is covered by the GNU Library General Public License instead.) You can apply it to your programs, too.

When we speak of free software, we are referring to freedom, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for this service if you wish), that you receive source code or can get it if you want it, that you can change the software or use pieces of it in new free programs; and that you know you can do these things.

To protect your rights, we need to make restrictions that forbid anyone to deny you these rights or to ask you to surrender the rights. These restrictions translate to certain responsibilities for you if you distribute copies of the software, or if you modify it.

For example, if you distribute copies of such a program, whether gratis or for a fee, you must give the recipients all the rights that you have. You must make sure that they, too, receive or can get the source code. And you must show them these terms so they know their rights.

We protect your rights with two steps: (1) copyright the software, and (2) offer you this license which gives you legal permission to copy, distribute and/or modify the software.

Also, for each author's protection and ours, we want to make certain that everyone understands that there is no warranty for this free software. If the software is modified by someone else and passed on, we want its recipients to know that what they have is not the original, so that any problems introduced by others will not reflect on the original authors' reputations.

Finally, any free program is threatened constantly by software patents. We wish to avoid the danger that redistributors of a free program will individually obtain patent licenses, in effect making the program proprietary. To prevent this, we have made it clear that any patent must be licensed for everyone's free use or not licensed at all.

The precise terms and conditions for copying, distribution and modification follow.

TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION

0. This License applies to any program or other work which contains a notice placed by the copyright holder saying it may be distributed under the terms of this General Public License. The "Program", below, refers to any such program or work, and a "work based on the Program" means either the Program or any derivative work under copyright law: that is to say, a work containing the Program or a portion of it, either verbatim or with modifications and/or translated into another language. (Hereinafter, translation is included without limitation in the term "modification".) Each licensee is addressed as "you".

Activities other than copying, distribution and modification are not covered by this License; they are outside its scope. The act of running the Program is not restricted, and the output from the Program is covered only if its contents constitute a work based on the Program (independent of having been made by running the Program). Whether that is true depends on what the Program does.

1. You may copy and distribute verbatim copies of the Program's source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice and disclaimer of warranty; keep intact all the notices that refer to this License and to the absence of any warranty; and give any other recipients of the Program a copy of this License along with the Program.

You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty protection in exchange for a fee.

2. You may modify your copy or copies of the Program or any portion of it, thus forming a work based on the Program, and copy and distribute such modifications or work under the terms of Section 1 above, provided that you also meet all of these conditions:

- **a.** You must cause the modified files to carry prominent notices stating that you changed the files and the date of any change.
- **b.** You must cause any work that you distribute or publish, that in whole or in part contains or is derived from the Program or any part thereof, to be licensed as a whole at no charge to all third parties under the terms of this License.
- **c.** If the modified program normally reads commands interactively when run, you must cause it, when started running for such interactive use in the most ordinary way, to print or display an announcement including an appropriate copyright notice and a notice that there is no warranty (or else, saying that you provide a warranty) and that users may redistribute the program under these conditions, and telling the user how to view a copy of this License. (Exception: if the Program itself is interactive but does not normally print such an announcement, your work based on the Program is not required to print an announcement.)

These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived from the Program, and can be reasonably considered independent and separate works in themselves, then this License, and its terms, do not apply to those sections when you distribute them as

separate works. But when you distribute the same sections as part of a whole which is a work based on the Program, the distribution of the whole must be on the terms of this License, whose permissions for other licensees extend to the entire whole, and thus to each and every part regardless of who wrote it.

Thus, it is not the intent of this section to claim rights or contest your rights to work written entirely by you; rather, the intent is to exercise the right to control the distribution of derivative or collective works based on the Program.

In addition, mere aggregation of another work not based on the Program with the Program (or with a work based on the Program) on a volume of a storage or distribution medium does not bring the other work under the scope of this License.

3. You may copy and distribute the Program (or a work based on it, under Section 2) in object code or executable form under the terms of Sections 1 and 2 above provided that you also do one of the following:

- **a.** Accompany it with the complete corresponding machine-readable source code, which must be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or
- **b.** Accompany it with a written offer, valid for at least three years, to give any third party, for a charge no more than your cost of physically performing source distribution, a complete machine-readable copy of the corresponding source code, to be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or
- **c.** Accompany it with the information you received as to the offer to distribute corresponding source code. (This alternative is allowed only for noncommercial distribution and only if you received the program in object code or executable form with such an offer, in accord with Subsection b above.)

The source code for a work means the preferred form of the work for making modifications to it. For an executable work, complete source code means all the source code for all modules it contains, plus any associated interface definition files, plus the scripts used to control compilation and installation of the executable. However, as a special exception, the source code distributed need not include anything that is normally distributed (in either source or binary form) with the major components (compiler, kernel, and so on) of the operating system on which the executable runs, unless that component itself accompanies the executable.

If distribution of executable or object code is made by offering access to copy from a designated place, then offering equivalent access to copy the source code from the same place counts as distribution of the source code, even though third parties are not compelled to copy the source along with the object code.

4. You may not copy, modify, sublicense, or distribute the Program except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense or distribute the Program is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.

5. You are not required to accept this License, since you have not signed it. However, nothing else grants you permission to modify or distribute the Program or its derivative works. These actions are prohibited by law if you do not accept this License. Therefore, by modifying or distributing the Program (or any work based on the Program), you indicate your acceptance of this License to do so, and all its terms and conditions for copying, distributing or modifying the Program or works based on it.

6. Each time you redistribute the Program (or any work based on the Program), the recipient automatically receives a license from the original licensor to copy, distribute or modify the Program subject to these terms and conditions. You may not impose any further restrictions on the recipients' exercise of the rights granted herein. You are not responsible for enforcing compliance by third parties to this License.

7. If, as a consequence of a court judgment or allegation of patent infringement or for any other reason (not limited to patent issues), conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot distribute so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not distribute the Program at all. For example, if a patent license would not permit royalty-free redistribution of the Program by all those who receive copies directly or indirectly through you, then the only way you could satisfy both it and this License would be to refrain entirely from distribution of the Program.

If any portion of this section is held invalid or unenforceable under any particular circumstance, the balance of the section is intended to apply and the section as a whole is intended to apply in other circumstances.

It is not the purpose of this section to induce you to infringe any patents or other property right claims or to contest validity of any such claims; this section has the sole purpose of protecting the integrity of the free software distribution system, which is implemented by public license practices. Many people have made generous contributions to the wide range of software distributed through that system in reliance on consistent application of that system; it is up to the author/donor to decide if he or she is willing to distribute software through any other system and a licensee cannot impose that choice.

This section is intended to make thoroughly clear what is believed to be a consequence of the rest of this License.

8. If the distribution and/or use of the Program is restricted in certain countries either by patents or by copyrighted interfaces, the original copyright holder who places the Program under this License may add an explicit geographical distribution limitation excluding those countries, so that distribution is permitted only in or among countries not thus excluded. In such case, this License incorporates the limitation as if written in the body of this License.

9. The Free Software Foundation may publish revised and/or new versions of the General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Program specifies a version number of this License which applies to it and “any later version”, you have the option of following the terms and conditions either of that version or of any later version published by the Free Software Foundation. If the Program does not specify a version number of this License, you may choose any version ever published by the Free Software Foundation.

10. If you wish to incorporate parts of the Program into other free programs whose distribution conditions are different, write to the author to ask for permission. For software which is copyrighted by the Free Software Foundation, write to the Free Software Foundation; we sometimes make exceptions for this. Our decision will be guided by the two goals of preserving the free status of all derivatives of our free software and of promoting the sharing and reuse of software generally.

NO WARRANTY

11. BECAUSE THE PROGRAM IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY FOR THE PROGRAM, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE PROGRAM “AS IS” WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE PROGRAM IS WITH YOU. SHOULD THE PROGRAM PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

12. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY AND/OR REDISTRIBUTE THE PROGRAM AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE PROGRAM (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE PROGRAM TO OPERATE WITH ANY OTHER PROGRAMS), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

END OF TERMS AND CONDITIONS

How to Apply These Terms to Your New Programs

If you develop a new program, and you want it to be of the greatest possible use to the public, the best way to achieve this is to make it free software which everyone can redistribute and change under these terms.

To do so, attach the following notices to the program. It is safest to attach them to the start of each source file to most effectively convey the exclusion of warranty; and each file should have at least the "copyright" line and a pointer to where the full notice is found.

one line to give the program's name and a brief idea of what it does. Copyright (C)

This program is free software; you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation; either version 2 of the License, or (at your option) any later version.

This program is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

You should have received a copy of the GNU General Public License along with this program; if not, write to the Free Software Foundation, Inc., 59 Temple Place, Suite 330, Boston, MA 02111-1307 USA

Also add information on how to contact you by electronic and paper mail.

If the program is interactive, make it output a short notice like this when it starts in an interactive mode:

Gnomovision version 69, Copyright (C) year name of author Gnomovision comes with ABSOLUTELY NO WARRANTY; for details type 'show w'. This is free software, and you are welcome to redistribute it under certain conditions; type 'show c' for details.

The hypothetical commands 'show w' and 'show c' should show the appropriate parts of the General Public License. Of course, the commands you use may be called something other than `show w' and 'show c'; they could even be mouse-clicks or menu items—whatever suits your program.

You should also get your employer (if you work as a programmer) or your school, if any, to sign a "copyright disclaimer" for the program, if necessary. Here is a sample; alter the names:

Yoyodyne, Inc., hereby disclaims all copyright interest in the program 'Gnomovision' (which makes passes at compilers) written by James Hacker.

signature of Ty Coon, 1 April 1989 Ty Coon, President of Vice

This General Public License does not permit incorporating your program into proprietary programs. If your program is a subroutine library, you may consider it more useful to permit linking proprietary applications with the library. If this is what you want to do, use the GNU Library General Public License instead of this License.

GNU LESSER GENERAL PUBLIC LICENSE

As a special exception, if other files instantiate generics from this library, or you link this library with other files to produce an executable, this library does not by itself cause the resulting executable to be covered by the GNU Lesser General Public License. This exception does not however invalidate any other reasons why the executable file might be covered by the GNU Lesser General Public License.

Version 2.1, February 1999

Copyright (C) 1991, 1999 Free Software Foundation, Inc. 59 Temple Place, Suite 330, Boston, MA 02111-1307 USA Everyone is permitted to copy and distribute verbatim copies of this license document, but changing it is not allowed.

[This is the first released version of the Lesser GPL. It also counts as the successor of the GNU Library Public License, version 2, hence the version number 2.1.]

Preamble

The licenses for most software are designed to take away your freedom to share and change it. By contrast, the GNU General Public Licenses are intended to guarantee your freedom to share and change free software to make sure the software is free for all its users.

This license, the Lesser General Public License, applies to some specially designated software packages typically libraries of the Free Software Foundation and other authors who decide to use it. You can use it too, but we suggest you first think carefully about whether this license or the ordinary General Public License is the better strategy to use in any particular case, based on the explanations below.

When we speak of free software, we are referring to freedom of use, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for this service if you wish); that you receive source code or can get it if you want it; that you can change the software and use pieces of it in new free programs; and that you are informed that you can do these things.

To protect your rights, we need to make restrictions that forbid distributors to deny you these rights or to ask you to surrender these rights. These restrictions translate to certain responsibilities for you if you distribute copies of the library or if you modify it.

For example, if you distribute copies of the library, whether gratis or for a fee, you must give the recipients all the rights that we gave you. You must make sure that they, too, receive or can get the source code. If you link other code with the library, you must provide complete object files to the recipients, so that they can relink them with the library after making changes to the library and recompiling it. And you must show them these terms so they know their rights.

We protect your rights with a two-step method: (1) we copyright the library, and (2) we offer you this license, which gives you legal permission to copy, distribute and/or modify the library.

To protect each distributor, we want to make it very clear that there is no warranty for the free library. Also, if the library is modified by someone else and passed on, the recipients should know that what they have is not the original version, so that the original author's reputation will not be affected by problems that might be introduced by others.

Finally, software patents pose a constant threat to the existence of any free program. We wish to make sure that a company cannot effectively restrict the users of a free program by obtaining a restrictive license from a patent holder. Therefore, we insist that any patent license obtained for a version of the library must be consistent with the full freedom of use specified in this license.

Most GNU software, including some libraries, is covered by the ordinary GNU General Public License. This license, the GNU Lesser General Public License, applies to certain designated libraries, and is quite different from the ordinary General Public License. We use this license for certain libraries in order to permit linking those libraries into non-free programs.

When a program is linked with a library, whether statically or using a shared library, the combination of the two is legally speaking a combined work, a derivative of the original library. The ordinary General Public License therefore permits such linking only if the entire combination fits its criteria of freedom. The Lesser General Public License permits more lax criteria for linking other code with the library.

We call this license the "Lesser" General Public License because it does Less to protect the user's freedom than the ordinary General Public License. It also provides other free software developers Less of an advantage over competing non-free programs. These disadvantages are the reason we use the ordinary General Public License for many libraries. However, the Lesser license provides advantages in certain special circumstances.

For example, on rare occasions, there may be a special need to encourage the widest possible use of a certain library, so that it becomes a de-facto standard. To achieve this, non-free programs must be allowed to use the library. A more frequent case is that a free library does the same job as widely used non-free libraries. In this case, there is little to gain by limiting the free library to free software only, so we use the Lesser General Public License.

In other cases, permission to use a particular library in non-free programs enables a greater number of people to use a large body of free software. For example, permission to use the GNU C Library in non-free programs enables many more people to use the whole GNU operating system, as well as its variant, the GNU/Linux operating system.

Although the Lesser General Public License is Less protective of the users' freedom, it does ensure that the user of a program that is linked with the Library has the freedom and the wherewithal to run that program using a modified version of the Library.

The precise terms and conditions for copying, distribution and modification follow. Pay close attention to the difference between a "work based on the library" and a "work that uses the library". The former contains code derived from the library, whereas the latter must be combined with the library in order to run.

GNU LESSER GENERAL PUBLIC LICENSE TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION

0. This License Agreement applies to any software library or other program which contains a notice placed by the copyright holder or other authorized party saying it may be distributed under the terms of this Lesser General Public License (also called "this License"). Each licensee is addressed as "you".

A "library" means a collection of software functions and/or data prepared so as to be conveniently linked with application programs (which use some of those functions and data) to form executables.

The "Library", below, refers to any such software library or work which has been distributed under these terms. A "work based on the Library" means either the Library or any derivative work under copyright law: that is to say, a work containing the Library or a portion of it, either verbatim or with modifications and/or translated straightforwardly into another language. (Hereinafter, translation is included without limitation in the term "modification".)

"Source code" for a work means the preferred form of the work for making modifications to it. For a library, complete source code means all the source code for all modules it contains, plus any associated interface definition files, plus the scripts used to control compilation and installation of the library.

Activities other than copying, distribution and modification are not covered by this License; they are outside its scope. The act of running a program using the Library is not restricted, and output from such a program is covered only if its contents constitute a work based on the Library (independent of the use of the Library in a tool for writing it). Whether that is true depends on what the Library does and what the program that uses the Library does.

1. You may copy and distribute verbatim copies of the Library's complete source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice and disclaimer of warranty; keep intact all the notices that refer to this License and to the absence of any warranty; and distribute a copy of this License along with the Library.

You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty protection in exchange for a fee.

2. You may modify your copy or copies of the Library or any portion of it, thus forming a work based on the Library, and copy and distribute such modifications or work under the terms of Section 1 above, provided that you also meet all of these conditions:

- **a.** The modified work must itself be a software library.
- **b.** You must cause the files modified to carry prominent notices stating that you changed the files and the date of any change.
- **c.** You must cause the whole of the work to be licensed at no charge to all third parties under the terms of this License.
- **d.** If a facility in the modified Library refers to a function or a table of data to be supplied by an application program that uses the facility, other than as an argument passed when the facility is invoked, then you must make a good faith effort to ensure that, in the event an application does not supply such function or table, the facility still operates, and performs whatever part of its purpose remains meaningful.

(For example, a function in a library to compute square roots has a purpose that is entirely well-defined independent of the application. Therefore, Subsection 2d requires that any application-supplied function or table used by this function must be optional: if the application does not supply it, the square root function must still compute square roots.)

These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived from the Library, and can be reasonably considered independent and separate works in themselves, then this License, and its terms, do not apply to those sections when you distribute them as separate works. But when you distribute the same sections as part of a whole which is a work based on the Library, the distribution of the whole must be on the terms of this License, whose permissions for other licensees extend to the entire whole, and thus to each and every part regardless of who wrote it.

Thus, it is not the intent of this section to claim rights or contest your rights to work written entirely by you; rather, the intent is to exercise the right to control the distribution of derivative or collective works based on the Library.

In addition, mere aggregation of another work not based on the Library with the Library (or with a work based on the Library) on a volume of a storage or distribution medium does not bring the other work under the scope of this License.

3. You may opt to apply the terms of the ordinary GNU General Public License instead of this License to a given copy of the Library. To do this, you must alter all the notices that refer to this License, so that they refer to the ordinary GNU General Public License, version 2, instead of to this License. (If a newer version than version 2 of the ordinary GNU General Public License has appeared, then you can specify that version instead if you wish.) Do not make any other change in these notices.

Once this change is made in a given copy, it is irreversible for that copy, so the ordinary GNU General Public License applies to all subsequent copies and derivative works made from that copy.

This option is useful when you wish to copy part of the code of the Library into a program that is not a library.

4. You may copy and distribute the Library (or a portion or derivative of it, under Section 2) in object code or executable form under the terms of Sections 1 and 2 above provided that you accompany it with the complete corresponding machine-readable source code, which must be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange.

If distribution of object code is made by offering access to copy from a designated place, then offering equivalent access to copy the source code from the same place satisfies the requirement to distribute the source code, even though third parties are not compelled to copy the source along with the object code.

5. A program that contains no derivative of any portion of the Library, but is designed to work with the Library by being compiled or linked with it, is called a "work that uses the Library". Such a work, in isolation, is not a derivative work of the Library, and therefore falls outside the scope of this License.

However, linking a "work that uses the Library" with the Library creates an executable that is a derivative of the Library (because it contains portions of the Library), rather than a "work that uses the library". The executable is therefore covered by this License. Section 6 states terms for distribution of such executables.

When a "work that uses the Library" uses material from a header file that is part of the Library, the object code for the work may be a derivative work of the Library even though the source code is not. Whether this is true is especially significant if the work can be linked without the Library, or if the work is itself a library. The threshold for this to be true is not precisely defined by law.

If such an object file uses only numerical parameters, data structure layouts and accessors, and small macros and small inline functions (ten lines or less in length), then the use of the object file is unrestricted, regardless of whether it is legally a derivative work. (Executables containing this object code plus portions of the Library will still fall under Section 6.)

Otherwise, if the work is a derivative of the Library, you may distribute the object code for the work under the terms of Section 6. Any executables containing that work also fall under Section 6, whether or not they are linked directly with the Library itself.

6. As an exception to the Sections above, you may also combine or link a "work that uses the Library" with the Library to produce a work containing portions of the Library, and distribute that work under terms of your choice, provided that the terms permit modification of the work for the customer's own use and reverse engineering for debugging such modifications.

You must give prominent notice with each copy of the work that the Library is used in it and that the Library and its use are covered by this License. You must supply a copy of this License. If the work during execution displays copyright notices, you must include the copyright notice for the Library among them, as well as a reference directing the user to the copy of this License. Also, you must do one of these things:

a. Accompany the work with the complete corresponding machine-readable source code for the Library including whatever changes were used in the work (which must be distributed under Sections 1 and 2 above); and, if the work is an executable linked with the Library, with the complete machine-readable "work that uses the Library", as object code and/or source code, so that the user can modify the Library and then relink to produce a modified executable containing the modified Library. (It is understood that the user who changes the contents of definitions files in the Library will not necessarily be able to recompile the application to use the modified definitions.)

b. Use a suitable shared library mechanism for linking with the Library. A suitable mechanism is one that (1) uses at run time a copy of the library already present on the user's computer system, rather than copying library functions into the executable, and (2) will operate properly with a modified version of the library, if the user installs one, as long as the modified version is interface-compatible with the version that the work was made with.

c. Accompany the work with a written offer, valid for at least three years, to give the same user the materials specified in Subsection 6a, above, for a charge no more than the cost of performing this distribution.

d. If distribution of the work is made by offering access to copy from a designated place, offer equivalent access to copy the above specified materials from the same place.

e. Verify that the user has already received a copy of these materials or that you have already sent this user a copy.

For an executable, the required form of the "work that uses the Library" must include any data and utility programs needed for reproducing the executable from it. However, as a special exception, the materials to be distributed need not include anything that is normally distributed (in either source or binary form) with the major components (compiler, kernel, and so on) of the operating system on which the executable runs, unless that component itself accompanies the executable.

It may happen that this requirement contradicts the license restrictions of other proprietary libraries that do not normally accompany the operating system. Such a contradiction means you cannot use both them and the Library together in an executable that you distribute.

7. You may place library facilities that are a work based on the Library side-by-side in a single library together with other library facilities not covered by this License, and distribute such a combined library, provided that the separate distribution of the work based on the Library and of the other library facilities is otherwise permitted, and provided that you do these two things:

a. Accompany the combined library with a copy of the same work based on the Library, uncombined with any other library facilities. This must be distributed under the terms of the Sections above.

b. Give prominent notice with the combined library of the fact that part of it is a work based on the Library, and explaining where to find the accompanying uncombined form of the same work.

8. You may not copy, modify, sublicense, link with, or distribute the Library except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense, link with, or distribute the Library is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.

9. You are not required to accept this License, since you have not signed it. However, nothing else grants you permission to modify or distribute the Library or its derivative works. These actions are prohibited by law if you do not accept this License. Therefore, by modifying or distributing the Library (or any work based on the Library), you indicate your acceptance of this License to do so, and all its terms and conditions for copying, distributing or modifying the Library or works based on it.

10. Each time you redistribute the Library (or any work based on the Library), the recipient automatically receives a license from the original licensor to copy, distribute, link with or modify the Library subject to these terms and conditions. You may not impose any further restrictions on the recipients' exercise of the rights granted herein. You are not responsible for enforcing compliance by third parties with this License.

11. If, as a consequence of a court judgment or allegation of patent infringement or for any other reason (not limited to patent issues), conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot distribute so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not distribute the Library at all. For example, if a patent license would not permit royalty-free redistribution of the Library by all those who receive copies directly or indirectly through you, then the only way you could satisfy both it and this License would be to refrain entirely from distribution of the Library.

If any portion of this section is held invalid or unenforceable under any particular circumstance, the balance of the section is intended to apply, and the section as a whole is intended to apply in other circumstances.

It is not the purpose of this section to induce you to infringe any patents or other property right claims or to contest validity of any such claims; this section has the sole purpose of protecting the integrity of the free software distribution system which is implemented by public license practices. Many people have made generous contributions to the wide range of software distributed through that system in reliance on consistent application of that system; it is up to the author/donor to decide if he or she is willing to distribute software through any other system and a licensee cannot impose that choice.

This section is intended to make thoroughly clear what is believed to be a consequence of the rest of this License.

12. If the distribution and/or use of the Library is restricted in certain countries either by patents or by copyrighted interfaces, the original copyright holder who places the Library under this License may add an explicit geographical distribution limitation excluding those countries, so that distribution is permitted only in or among countries not thus excluded. In such case, this License incorporates the limitation as if written in the body of this License.

13. The Free Software Foundation may publish revised and/or new versions of the Lesser General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Library specifies a version number of this License which applies to it and “any later version”, you have the option of following the terms and conditions either of that version or of any later version published by the Free Software Foundation. If the Library does not specify a license version number, you may choose any version ever published by the Free Software Foundation.

14. If you wish to incorporate parts of the Library into other free programs whose distribution conditions are incompatible with these, write to the author to ask for permission. For software which is copyrighted by the Free Software Foundation, write to the Free Software Foundation; we sometimes make exceptions for this. Our decision will be guided by the two goals of preserving the free status of all derivatives of our free software and of promoting the sharing and reuse of software generally.

NO WARRANTY

15. BECAUSE THE LIBRARY IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY FOR THE LIBRARY, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE LIBRARY “AS IS” WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE LIBRARY IS WITH YOU. SHOULD THE LIBRARY PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

16. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY AND/OR REDISTRIBUTE THE LIBRARY AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE LIBRARY (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE LIBRARY TO OPERATE WITH ANY OTHER SOFTWARE), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

END OF TERMS AND CONDITIONS

How to Apply These Terms to Your New Libraries

If you develop a new library, and you want it to be of the greatest possible use to the public, we recommend making it free software that everyone can redistribute and change. You can do so by permitting redistribution under these terms (or, alternatively, under the terms of the ordinary General Public License).

To apply these terms, attach the following notices to the library. It is safest to attach them to the start of each source file to most effectively convey the exclusion of warranty; and each file should have at least the "copyright" line and a pointer to where the full notice is found.

<one line to give the library's name and a brief idea of what it does.> Copyright (C) <year> <name of author>

This library is free software; you can redistribute it and/or modify it under the terms of the GNU Lesser General Public License as published by the Free Software Foundation; either version 2.1 of the License, or (at your option) any later version.

This library is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU Lesser General Public License for more details.

You should have received a copy of the GNU Lesser General Public License along with this library; if not, write to the Free Software Foundation, Inc., 59 Temple Place, Suite 330, Boston, MA 02111-1307 USA

Also add information on how to contact you by electronic and paper mail.

You should also get your employer (if you work as a programmer) or your school, if any, to sign a "copyright disclaimer" for the library, if necessary. Here is a sample; alter the names:

Yoyodyne, Inc., hereby disclaims all copyright interest in the library 'Frob' (a library for tweaking knobs) written by James Random Hacker.

<signature of Ty Coon>, 1 April 1990 Ty Coon, President of Vice

That's all there is to it!

Copyright (c) 1995, 1996, 1997, 1998, 1999 Kungliga Tekniska Högskolan (Royal Institute of Technology, Stockholm, Sweden). All rights reserved. (remainder is BSD-type license; see uploaded file).

libtecla License

Copyright (c) 2000, 2001 by Martin C. Shepherd.

All rights reserved.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, provided that the above copyright notice(s) and this permission notice appear in all copies of the Software and that both the above copyright notice(s) and this permission notice appear in supporting documentation.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT OF THIRD PARTY RIGHTS. IN NO EVENT SHALL THE COPYRIGHT HOLDER OR HOLDERS INCLUDED IN THIS NOTICE BE LIABLE FOR ANY CLAIM, OR ANY SPECIAL INDIRECT OR CONSEQUENTIAL

DAMAGES, OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE.

Except as contained in this notice, the name of a copyright holder shall not be used in advertising or otherwise to promote the sale, use or other dealings in this Software without prior written authorization of the copyright holder.

Linux-PAM License

Unless otherwise explicitly stated the following text describes the licensed conditions under which the contents of this Linux-PAM release may be distributed:

Redistribution and use in source and binary forms of Linux-PAM, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain any existing copyright notice, and this entire permission notice in its entirety, including the disclaimer of warranties.
2. Redistributions in binary form must reproduce all prior and current copyright notices, this list of conditions, and the following disclaimer in the documentation and/or other materials provided with the distribution.
3. The name of any author may not be used to endorse or promote products derived from this software without their specific prior written permission.

ALTERNATIVELY, this product may be distributed under the terms of the GNU General Public License, in which case the provisions of the GNU GPL are required INSTEAD OF the above restrictions. (This clause is necessary due to a potential conflict between the GNU GPL and the restrictions contained in a BSD-style copyright.)

THIS SOFTWARE IS PROVIDED “AS IS” AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE AUTHOR(S) BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Makefile.in License

Copyright (c) 1998 Free Software Foundation, Inc.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the “Software”), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, distribute with modifications, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE ABOVE COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Except as contained in this notice, the name(s) of the above copyright holders shall not be used in advertising or otherwise to promote the sale, use or other dealings in this Software without prior written authorization.

Modified BSD License

This software is copyrighted by the Regents of the University of California, Sun Microsystems, Inc., Scriptics Corporation, ActiveState Corporation and other parties. The following terms apply to all files associated with the software unless explicitly disclaimed in individual files. The authors hereby grant permission to use, copy, modify, distribute, and license this software and its documentation for any purpose, provided that existing copyright notices are retained in all copies and that this notice is included verbatim in any distributions. No written agreement, license, or royalty fee is required for any of the authorized uses. Modifications to this software may be copyrighted by their authors and need not follow the licensing terms described here, provided that the new terms are clearly indicated on the first page of each file where they apply. IN NO EVENT SHALL THE AUTHORS OR DISTRIBUTORS BE LIABLE TO ANY PARTY FOR DIRECT, INDIRECT, SPECIAL, INCIDENTAL, OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OF THIS SOFTWARE, ITS DOCUMENTATION, OR ANY DERIVATIVES THEREOF, EVEN IF THE AUTHORS HAVE BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGE. THE AUTHORS AND DISTRIBUTORS SPECIFICALLY DISCLAIM ANY WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, AND NON-INFRINGEMENT. THIS SOFTWARE IS PROVIDED ON AN "AS IS" BASIS, AND THE AUTHORS AND DISTRIBUTORS HAVE NO OBLIGATION TO PROVIDE MAINTENANCE, SUPPORT, UPDATES, ENHANCEMENTS, OR MODIFICATIONS. GOVERNMENT USE: If you are acquiring this software on behalf of the U.S. government, the Government shall have only "Restricted Rights" in the software and related documentation as defined in the Federal Acquisition Regulations (FARs) in Clause 52.227.19 (c) (2). If you are acquiring the software on behalf of the Department of Defense, the software shall be classified as "Commercial Computer Software" and the Government shall have only "Restricted Rights" as defined in Clause 252.227-7013 (c) (1) of DFARs. Notwithstanding the foregoing, the authors grant the U.S. Government and others acting in its behalf permission to use and distribute the software in accordance with the terms specified in this license.

Network Time Protocol Version 4 Distribution License

This file is automatically generated from html/copyright.html

Copyright Notice

jpg "Clone me," says Dolly sheepishly

Last update: 15:44 UTC Tuesday, July 15, 2003

The following copyright notice applies to all files collectively called the Network Time Protocol Version 4 Distribution. Unless specifically declared otherwise in an individual file, this notice applies as if the text was explicitly included in the file.

Copyright (c) David L. Mills 1992-2003

Permission to use, copy, modify, and distribute this software and its documentation for any purpose and without fee is hereby granted, provided that the above copyright notice appears in all copies and that both the copyright notice and this permission notice appear in supporting documentation, and that the name University of Delaware not be used in advertising or publicity pertaining to distribution of the software without specific, written prior permission. The University of Delaware makes no representations about the suitability this software for any purpose. It is provided "as is" without express or implied warranty.

The following individuals contributed in part to the Network Time Protocol Distribution Version 4 and are acknowledged as authors of this work.

1. [1] Mark Andrews <mark_andrews@isc.org> Leitch atomic clock controller
2. [2] Bernd Altmeier <altmeier@atlsoft.de> hopf Elektronik serial line and interface-bus devices
3. [3] Viraj Bais <vbais@mailman1.intel.com> and [4]Clayton Kirkwood <kirkwood@striderfm.intel.com> port to WindowsNT 3.5
4. [5] Michael Barone <michael,barone@lmco.com> GPSVME fixes
5. [6] Jean-Francois Boudreault <Jean-Francois.Boudreault@viagenie.qc.ca>IPv6 support
6. [7] Karl Berry <karl@owl.HQ.ileaf.com> syslog to file option
7. [8] Greg Brackley <greg.brackley@bigfoot.com> Major rework of WINNT port. Clean up recvbuf and iosignal code into separate modules.
8. [9] Marc Brett <Marc.Brett@westgeo.com> Magnavox GPS clock driver
9. [10] Piete Brooks <Piete.Brooks@cl.cam.ac.uk> MSF clock driver, Trimble PARSE support
10. [11] Reg Clemens <reg@dwf.com> Oncore driver (Current maintainer)
11. [12]Steve Clift <clift@ml.csiro.au> OMEGA clock driver
12. [13]Casey Crellin <casey@csc.co.za> vxWorks (Tornado) port and help with target configuration
13. [14]Sven Dietrich <sven_dietrich@trimble.com> Palisade reference clock driver, NT adj. residuals, integrated Greg's Winnt port.
14. [15]John A. Dundas III <dundas@salt.jpl.nasa.gov> Apple A/UX port
15. [16]Torsten Duwe <duwe@immd4.informatik.uni-erlangen.de> Linux port
16. [17] Dennis Ferguson <dennis@mrbill.canet.ca> foundation code for NTP Version 2 as specified in RFC-1119
17. [18] John Hay <jhay@@icomtek.csir.co.za> IPv6 support and testing
18. [19] Glenn Hollinger <glenn@herald.usask.ca> GOES clock driver
19. [20] Mike Iglesias <iglesias@uci.edu> DEC Alpha port
20. [21] Jim Jagielski <jim@jagubox.gsfc.nasa.gov> A/UX port
21. [22] Jeff Johnson <jbj@chatham.usdesign.com> massive prototyping overhaul
22. [23] Hans Lambermont <Hans.Lambermont@nl.origin-it.com> or [24]<H.Lambermont@chello.nl> ntpsweep
23. [25] Poul-Henning Kamp <phk@FreeBSD.ORG> Oncore driver (Original author)
24. [26] Frank Kardel [27]<Frank.Kardel@informatik.uni-erlangen.de> PARSE <GENERIC> driver (14 reference clocks), STREAMS modules for PARSE, support scripts, syslog cleanup
25. [28]William L. Jones <jones@hermes.chpc.utexas.edu> RS/6000 AIX modifications, HPUX modifications

26. [29]Dave Katz <dkatz@cisco.com> RS/6000 AIX port
27. [30]Craig Leres <leres@ee.lbl.gov> 4.4BSD port, ppsclock, Magnavox GPS clock driver
28. [31] George Lindholm <lindholm@ucs.ubc.ca> SunOS 5.1 port
29. [32] Louis A. Mamakos <louie@ni.umd.edu> MD5-based authentication
30. [33] Lars H. Mathiesen <thorinn@diku.dk> adaptation of foundation code for Version 3 as specified in RFC-1305
31. [34] Danny Mayer <mayer@ntp.org>Network I/O, Windows Port, Code Maintenance
32. [35] David L. Mills <mills@udel.edu> Version 4 foundation: clock discipline, authentication, precision kernel; clock drivers: Spectracom, Austron, Arbiter, Heath, ATOM, ACTS, KSI/Odetics; audio clock drivers: CHU, WWV/H, IRIG
33. [36] Wolfgang Moeller <moeller@gwdgv1.dnet.gwdg.de> VMS port
34. [37] Jeffrey Mogul <mogul@pa.dec.com> ntptrace utility
35. [38] Tom Moore <tmoore@fievel.daytonoh.ncr.com> i386 svr4 port
36. [39] Kamal A Mostafa <kamal@whence.com> SCO OpenServer port
37. [40] Derek Mulcahy <derek@toybox.demon.co.uk> and [41]Damon Hart-Davis <d@hd.org> ARCRON MSF clock driver
38. [42] Rainer Pruy <Rainer.Pruy@informatik.uni-erlangen.de> monitoring/trap scripts, statistics file handling
39. [43] Dirce Richards <dirce@zk3.dec.com> Digital UNIX V4.0 port
40. [44] Wilfredo Sánchez <wsanchez@apple.com> added support for NetInfo
41. [45]Nick Sayer <mrapple@quack.kfu.com> SunOS streams modules
42. [46]Jack Sasportas <jack@innovativeinternet.com> Saved a Lot of space on the stuff in the html/pic/ subdirectory
43. [47]Ray Schnitzler <schnitz@unipress.com> Unixware1 port
44. [48]Michael Shields <shields@tembel.org> USNO clock driver
45. [49]Jeff Steinman <jss@pebbles.jpl.nasa.gov> Datum PTS clock driver
46. [50]Harlan Stenn <harlan@pfcs.com> GNU automake/autoconfigure makeover, various other bits (see the ChangeLog)
47. [51]Kenneth Stone <ken@sdd.hp.com> HP-UX port
48. [52]Ajit Thyagarajan <ajit@ee.udel.edu>IP multicast/anycast support
49. [53]Tomoaki TSURUOKA <tsuruoka@nc.fukuoka-u.ac.jp>TRAK clock driver
50. [54]Paul A Vixie <vixie@vix.com> TrueTime GPS driver, generic TrueTime clock driver
51. [55]Ulrich Windl <Ulrich.Windl@rz.uni-regensburg.de> corrected and validated HTML documents according to the HTML DTD

References

1. mailto:%20mark_andrews@isc.org
2. mailto:%20altmeier@atlsoft.de
3. mailto:%20vbais@mailman1.intel.co
4. mailto:%20kirkwood@striderfm.intel.com

5. mailto:%20michael.barone@lmco.com
6. mailto:%20Jean-Francois.Boudreault@viagenie.qc.ca
7. mailto:%20karl@owl.HQ.ileaf.com
8. mailto:%20greg.brackley@bigfoot.com
9. mailto:%20Marc.Brett@westgeo.com
10. mailto:%20Piete.Brooks@cl.cam.ac.uk
11. mailto:%20reg@dwf.com
12. mailto:%20clift@ml.csiro.au
13. mailto:casey@csc.co.za
14. mailto:%20Sven_Dietrich@trimble.COM
15. mailto:%20dundas@salt.jpl.nasa.gov
16. mailto:%20duwe@immd4.informatik.uni-erlangen.de
17. mailto:%20dennis@mrbill.canet.ca
18. mailto:%20jhay@icomtek.csir.co.za
19. mailto:%20glenn@herald.usask.ca
20. mailto:%20iglesias@uci.edu
21. mailto:%20jagubox.gsfc.nasa.gov
22. mailto:%20jbj@chatham.usdesign.com
23. mailto:Hans.Lambermont@nl.origin-it.com
24. mailto:H.Lambermont@chello.nl
25. mailto:%20phk@FreeBSD.ORG
26. http://www4.informatik.uni-erlangen.de/%7ekardel
27. mailto:%20Frank.Kardel@informatik.uni-erlangen.de
28. mailto:%20jones@hermes.chpc.utexas.edu
29. mailto:%20dkatz@cisco.com
30. mailto:%20leres@ee.lbl.gov
31. mailto:%20lindholm@ucs.ubc.ca
32. mailto:%20louie@ni.umd.edu
33. mailto:%20thorinn@diku.dk
34. mailto:%20mayer@ntp.org
35. mailto:%20mills@udel.edu
36. mailto:%20moeller@gwdgv1.dnet.gwdg.de
37. mailto:%20mogul@pa.dec.com
38. mailto:%20tmoore@fievel.daytonoh.ncr.com
39. mailto:%20kamal@whence.com
40. mailto:%20derek@toybox.demon.co.uk
41. mailto:%20d@hd.org
42. mailto:%20Rainer.Pruy@informatik.uni-erlangen.de

43. mailto:%20dirce@zk3.dec.com
44. mailto:%20wsanchez@apple.com
45. mailto:%20mrapple@quack.kfu.com
46. mailto:%20jack@innovativeinternet.com
47. mailto:%20schnitz@unipress.com
48. mailto:%20shields@tembel.org
49. mailto:%20pebbles.jpl.nasa.gov
50. mailto:%20harlan@pfcs.com
51. mailto:%20ken@sdd.hp.com
52. mailto:%20ajit@ee.udel.edu
53. mailto:%20tsuruoka@nc.fukuoka-u.ac.jp
54. mailto:%20vixie@vix.com
55. mailto:%20Ulrich.Windl@rz.uni-regensburg.de

Open SSL License

This product includes software developed by the OpenSSL Project for use in the OpenSSL Toolkit. (http://www.openssl.org/)

This product includes cryptographic software written by Eric Young (eay@cryptsoft.com).

This product includes software written by Tim Hudson (tjh@cryptsoft.com).

LICENSE ISSUES

The OpenSSL toolkit stays under a dual license, i.e. both the conditions of the OpenSSL License and the original SSLeay license apply to the toolkit. See below for the actual license texts. Actually both licenses are BSD-style Open Source licenses. In case of any license issues related to OpenSSL please contact openssl-core@openssl.org.

OpenSSL License

Copyright (c) 1998-2000 The OpenSSL Project. All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
3. All advertising materials mentioning features or use of this software must display the following acknowledgment: “This product includes software developed by the OpenSSL Project for use in the OpenSSL Toolkit. (http://www.openssl.org/)”
4. The names “OpenSSL Toolkit” and “OpenSSL Project” must not be used to endorse or promote products derived from this software without prior written permission. For written permission, please contact openssl-core@openssl.org.
5. Products derived from this software may not be called “OpenSSL” nor may “OpenSSL” appear in their names without prior written permission of the OpenSSL Project.

6. Redistributions of any form whatsoever must retain the following acknowledgment: "This product includes software developed by the OpenSSL Project for use in the OpenSSL Toolkit (http://www.openssl.org/)"

   THIS SOFTWARE IS PROVIDED BY THE OpenSSL PROJECT "AS IS" AND ANY EXPRESSED OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE OpenSSL PROJECT OR ITS CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

This product includes cryptographic software written by Eric Young (eay@cryptsoft.com). This product includes software written by Tim Hudson (tjh@cryptsoft.com).

Original SSLeay License

Copyright (C) 1995-1998 Eric Young (eay@cryptsoft.com) All rights reserved.

This package is an SSL implementation written by Eric Young (eay@cryptsoft.com). The implementation was written so as to conform with Netscape's SSL.

This library is free for commercial and non-commercial use as long as the following conditions are adhered to. The following conditions apply to all code found in this distribution, be it the RC4, RSA, lhash, DES, etc., code; not just the SSL code. The SSL documentation included with this distribution is covered by the same copyright terms except that the holder is Tim Hudson (tjh@cryptsoft.com).

Copyright remains Eric Young's, and as such any Copyright notices in the code are not to be removed. If this package is used in a product, Eric Young should be given attribution as the author of the parts of the library used. This can be in the form of a textual message at program startup or in documentation (online or textual) provided with the package.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
3. All advertising materials mentioning features or use of this software must display the following acknowledgement: "This product includes cryptographic software written by Eric Young (eay@cryptsoft.com)" The word 'cryptographic' can be left out if the routines from the library being used are not cryptographic related :-).
4. If you include any Windows specific code (or a derivative thereof) from the apps directory (application code) you must include an acknowledgement: "This product includes software written by Tim Hudson (tjh@cryptsoft.com)"

   THIS SOFTWARE IS PROVIDED BY ERIC YOUNG "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE AUTHOR OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS

INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

The licence and distribution terms for any publicly available version or derivative of this code cannot be changed. i.e. this code cannot simply be copied and put under another distribution licence [including the GNU Public Licence.]

PATENTS

Various companies hold various patents for various algorithms in various locations around the world. **YOU** are responsible for ensuring that your use of any algorithms is legal by checking if there are any patents in your country. The file contains some of the patents that we know about or are rumored to exist. This is not a definitive list.

RSA Data Security holds software patents on the RSA and RC5 algorithms. If their ciphers are used inside the USA (and Japan?), you must contact RSA Data Security for licensing conditions. Their web page is http://www.rsa.com/.

RC4 is a trademark of RSA Data Security, so use of this label should perhaps only be used with RSA Data Security's permission.

The IDEA algorithm is patented by Ascom in Austria, France, Germany, Italy, Japan, Netherlands, Spain, Sweden, Switzerland, UK and the USA. They should be contacted if that algorithm is to be used, their web page is http://www.ascom.ch/.

5. The name of the Copyright Holder may not be used to endorse or promote products derived from this software without specific prior written permission.
6. THIS PACKAGE IS PROVIDED "AS IS" AND WITHOUT ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, WITHOUT LIMITATION, THE IMPLIED WARRANTIES OF MERCHANTIBILITY AND FITNESS FOR A PARTICULAR PURPOSE.

UCD Net-SNMP Version 5.1 License

Various copyrights apply to this package, listed in 4 separate parts below. Please make sure that you read all the parts. Up until 2001, the project was based at UC Davis, and the first part covers all code written during this time. From 2001 onwards, the project has been based at SourceForge, and Networks Associates Technology, Inc hold the copyright on behalf of the wider Net-SNMP community, covering all derivative work done since then. An additional copyright section has been added as Part 3 below also under a BSD license for the work contributed by Cambridge Broadband Ltd. to the project since 2001. An additional copyright section has been added as Part 4 below also under a BSD license for the work contributed by Sun Microsystems, Inc. to the project since 2003.

Code has been contributed to this project by many people over the years it has been in development, and a full list of contributors can be found in the README file under the THANKS section.

Part 1: CMU/UCD copyright notice: (BSD like)

Copyright 1989, 1991, 1992 by Carnegie Mellon University

Derivative Work - 1996, 1998-2000

Copyright 1996, 1998-2000 The Regents of the University of California

All Rights Reserved

Permission to use, copy, modify and distribute this software and its documentation for any purpose and without fee is hereby granted, provided that the above copyright notice appears in all copies and that both that copyright notice and this permission notice appear in supporting documentation, and that the name of CMU and The Regents of the University of California not be used in advertising or publicity pertaining to distribution of the software without specific written permission.

CMU AND THE REGENTS OF THE UNIVERSITY OF CALIFORNIA DISCLAIM ALL WARRANTIES WITH REGARD TO THIS SOFTWARE, INCLUDING ALL IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS. IN NO EVENT SHALL CMU OR THE REGENTS OF THE UNIVERSITY OF CALIFORNIA BE LIABLE FOR ANY SPECIAL, INDIRECT OR CONSEQUENTIAL DAMAGES OR ANY DAMAGES WHATSOEVER RESULTING FROM THE LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE.

Part 2: Networks Associates Technology, Inc copyright notice (BSD)

Copyright (c) 2001-2003, Networks Associates Technology, Inc. All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

- Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
- Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
- Neither the name of the Networks Associates Technology, Inc nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS “AS IS” AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT HOLDERS OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Part 3: Cambridge Broadband Ltd. copyright notice (BSD)

Portions of this code are copyright (c) 2001-2003, Cambridge Broadband Ltd. All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

- Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
- Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
- The name of Cambridge Broadband Ltd. may not be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDER "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT HOLDER BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Part 4: Sun Microsystems, Inc. copyright notice (BSD)

Copyright © 2003 Sun Microsystems, Inc., 4150 Network Circle, Santa Clara, California 95054, U.S.A. All rights reserved.

Use is subject to license terms below.

This distribution may include materials developed by third parties.

Sun, Sun Microsystems, the Sun logo and Solaris are trademarks or registered trademarks of Sun Microsystems, Inc. in the U.S. and other countries.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

- Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
- Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
- Neither the name of the Sun Microsystems, Inc. nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT HOLDERS OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Wietse Venema License

Copyright 1995 by Wietse Venema. All rights reserved. Some individual files may be covered by other copyrights. This material was originally written and compiled by Wietse Venema at Eindhoven University of Technology, The Netherlands, in 1990, 1991, 1992, 1993, 1994 and 1995. Redistribution and use in source and binary forms are permitted provided that this entire copyright notice is duplicated in all such copies. This software is provided "as is" and without any expressed or implied warranties, including, without limitation, the implied warranties of merchantibility and fitness for any particular purpose.

zlib License

zlib 1.1.4 is a general purpose data compression library. All the code is thread safe. The data format used by the zlib library is described by RFCs (Request for Comments) 1950 to 1952 in the files http://www.ietf.org/rfc/rfc1950.txt (zlib format), rfc1951.txt (deflate format) and rfc1952.txt (gzip format). These documents are also available in other formats from ftp://ftp.uu.net/graphics/png/documents/zlib/zdoc-index.html

All functions of the compression library are documented in the file zlib.h (volunteer to write man pages welcome, contact jloup@gzip.org). A usage example of the library is given in the file example.c which also tests that the library is working correctly. Another example is given in the file minigzip.c. The compression library itself is composed of all source files except example.c and minigzip.c.

To compile all files and run the test program, follow the instructions given at the top of Makefile. In short "make test; make install" should work for most machines. For Unix: "./configure; make test; make install" For MSDOS, use one of the special makefiles such as Makefile.msc. For VMS, use Make_vms.com or descrip.mms.

Questions about zlib should be sent to <zlib@gzip.org>, or to Gilles Vollant <info@winimage.com> for the Windows DLL version. The zlib home page is http://www.zlib.org or http://www.gzip.org/zlib/. Before reporting a problem, please check this site to verify that you have the latest version of zlib; otherwise get the latest version and check whether the problem still exists or not.

PLEASE read the zlib FAQ http://www.gzip.org/zlib/zlib_faq.html before asking for help.

Mark Nelson <markn@ieee.org> wrote an article about zlib for the Jan. 1997 issue of Dr. Dob's Journal; a copy of the article is available in http://dogma.net/markn/articles/zlibtool/zlibtool.htm

The changes made in version 1.1.4 are documented in the file ChangeLog. The only changes made since 1.1.3 are bug corrections:

- ZFREE was repeated on same allocation on some error conditions.

  This creates a security problem described in http://www.zlib.org/advisory-2002-03-11.txt
- Returned incorrect error (Z_MEM_ERROR) on some invalid data -
- Avoid accesses before window for invalid distances with inflate window less than 32K.
- force windowBits > 8 to avoid a bug in the encoder for a window size of 256 bytes. (A complete fix will be available in 1.1.5).

The beta version 1.1.5beta includes many more changes. A new official version 1.1.5 will be released as soon as extensive testing has been completed on it.

Unsupported third party contributions are provided in directory "contrib".

A Java implementation of zlib is available in the Java Development Kit http://www.javasoft.com/products/JDK/1.1/docs/api/Package-java.util.zip.html. See the zlib home page http://www.zlib.org for details.

A Perl interface to zlib written by Paul Marquess <pmarquess@bfsec.bt.co.uk> is in the CPAN (Comprehensive Perl Archive Network) sites http://www.cpan.org/modules/by-module/Compress/

A Python interface to zlib written by A.M. Kuchling <amk@magnet.com> is available in Python 1.5 and later versions, see http://www.python.org/doc/lib/module-zlib.html

A zlib binding for TCL written by Andreas Kupries <a.kupries@westend.com> is available at http://www.westend.com/~kupries/doc/trf/man/man.html

An experimental package to read and write files in .zip format, written on top of zlib by Gilles Vollant <info@winimage.com>, is available at http://www.winimage.com/zLibDll/unzip.html and also in the contrib/minizip directory of zlib.

Notes for some targets:

- To build a Windows DLL version, include in a DLL project zlib.def, zlib.rc and all .c files except example.c and minigzip.c; compile with -DZLIB_DLL The zlib DLL support was initially done by Alessandro Iacopetti and is now maintained by Gilles Vollant <info@winimage.com>. Check the zlib DLL home page at http://www.winimage.com/zLibDll

  From Visual Basic, you can call the DLL functions which do not take a structure as argument: compress, uncompress and all gz* functions. See contrib/visual-basic.txt for more information, or get http://www.tcfb.com/dowseware/cmp-z-it.zip

- For 64-bit Irix, deflate.c must be compiled without any optimization.

  With -O, one libpng test fails. The test works in 32 bit mode (with the -n32 compiler flag). The compiler bug has been reported to SGI.

- zlib doesn't work with gcc 2.6.3 on a DEC 3000/300LX under OSF/1 2.1 it works when compiled with cc.
- on Digital Unix 4.0D (formerly OSF/1) on AlphaServer, the cc option -std1 is necessary to get gzprintf working correctly. This is done by configure.
- zlib doesn't work on HP-UX 9.05 with some versions of /bin/cc. It works with other compilers. Use “make test” to check your compiler.
- gzdopen is not supported on RISCOS, BEOS and by some Mac compilers.
- For Turbo C the small model is supported only with reduced performance to avoid any far allocation; it was tested with -DMAX_WBITS=11 -DMAX_MEM_LEVEL=3
- For PalmOs, see http://www.cs.uit.no/~perm/PASTA/pilot/software.html Per Harald Myrvang <perm@stud.cs.uit.no>

Acknowledgments:

The deflate format used by zlib was defined by Phil Katz. The deflate and zlib specifications were written by L. Peter Deutsch. Thanks to all the people who reported problems and suggested various improvements in zlib; they are too numerous to cite here.

Copyright notice:

(C) 1995-2002 Jean-loup Gailly and Mark Adler

This software is provided “as-is”, without any express or implied warranty. In no event will the authors be held liable for any damages arising from the use of this software.

Permission is granted to anyone to use this software for any purpose, including commercial applications, and to alter it and redistribute it freely, subject to the following restrictions:

1. The origin of this software must not be misrepresented; you must not claim that you wrote the original software. If you use this software in a product, an acknowledgment in the product documentation would be appreciated but is not required.
2. Altered source versions must be plainly marked as such, and must not be misrepresented as being the original software.
3. This notice may not be removed or altered from any source distribution.

Jean-loup Gailly Mark Adler

jloup@gzip.org madler@alumni.caltech.edu

If you use the zlib library in a product, we would appreciate **not** receiving lengthy legal documents to sign. The sources are provided for free but without warranty of any kind. The library has been entirely written by Jean-loup Gailly and Mark Adler; it does not include third-party code.

If you redistribute modified sources, we would appreciate that you include in the file ChangeLog history information documenting your changes.

APPENDIX E

CLI Error Messages

This appendix lists the CLI error messages and CLI validation error messages. It contains the following sections:

CLI Error Messages

Table E-1 describes CLI error messages.

Table E-1 CLI Error Messages

Error Message	Reason	Command
`getVirtualSensorStatistics : Analysis Engine is busy`	Analysis Engine is busy because the virtual sensor has not finished initializing.	**show statistics virtual-sensor**
`getVirtualSensorStatistics : Analysis Engine is busy rebuilding regex tables. This may take a while.`	Analysis Engine is busy building cache files immediately after the sensor has been imaged.	**show statistics virtual-sensor**
`editConfigDeltaSignatureDefinition : Analysis Engine is busy rebuilding regex tables. This may take a while.`	Analysis Engine is busy building cache files immediately after the sensor has been imaged.	**service signature-definition**
`Invalid command received.`	The .conf file and code are out of synchronization, which should never occur in the field.	All commands
`Invalid port number was entered.`	An out-of-range port number was entered in URI.	**copy**, **upgrade**, **show tech-support**
`Invalid scheme was entered.`	Internal tables are out of synchronization, which should never occur in the field.	**copy**, **upgrade**, **show tech-support**
`Unknown scheme was entered.`	An invalid scheme was entered in URI.	**copy**, **upgrade**, **show tech-support**

Table E-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**
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
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**

Table E-1 CLI Error Messages (continued)

Error Message	Reason	Command
`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**
`Log file exists but an error occurred during read.`	The user was displaying or copying an iplog file that was overwritten. The partial file contents should still be viewable.	**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**
`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**
`The log file was 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**

Table E-1 CLI Error Messages (continued)

Error Message	Reason	Command
`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 E-2 describes the validation error messages.

Table E-2 Validation Error Messages

Error Message	Reason/Location
`Interface 'name' has not been subdivided.`	The physical interface or inline interface *name* subinterface type is none (service interface submode).
`Interface 'name' subinterface 'num' does not exist.`	The physical interface *name* 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 *name* is the command and control interface (service interface submode).
`Interface 'name' has been subdivided.`	The physical interface *name* subinterface type is inline VLAN pair or VLAN group. Or the inline interface *name* subinterface type is VLAN group (service interface submode).
`Interface 'name' is assigned to inline-interfaces 'inlinename.'`	The physical interface *name* is assigned to an inline interface entry's interface1 or interface2 (service interface submode).
`Vlan 'vlannum' is assigned to subinterface 'subnum.'`	The VLAN *vlannum* is already assigned to a different subinterface *subnum* entry's vlan1 or vlan2 (service interface submode).
`Vlan range 'vlanrange' overlaps with vlans assigned to subinterface 'subnum.'`	The VLAN range *vlanrange* contains values that are already used in a different subinterface *subnum* entry's *vlans range* (service interface submode).
`Unassigned vlans already assigned to subinterface 'subnum.'`	Unassigned VLANs have already been selected in a different subinterface *subnum* entry.
`Inline-interface 'inlinename' does not exist.`	The inline interface *inlinename* 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 *name* subinterface type is not inline VLAN pair (service interface submode).

Table E-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 *vsname*.

GLOSSARY

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. The primary and recommended method for access control in Cisco devices.

AAA authentication, authorization, and accounting. Pronounced “triple a.”

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 acknowledgement. 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.

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.

adaptive security appliance Combines firewall, VPN concentrator, and intrusion prevention software functionality into one software image. You can configure the adaptive security appliance in single mode or multi-mode.

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.

AIM IPS Advanced Integration Module. A type of IPS network module installed in Cisco routers.

AIP SSM Advanced Inspection and Prevention Security Services Module. The IPS plug-in module in the Cisco ASA 5500 series adaptive security appliance. See ASA.

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.

ASDM Adaptive Security Device Manager. A web-based application that lets you configure and manage your ASA.

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.

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.

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.

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. It verifies that users have the correct permissions to perform CLI, IDM, IME, or RDEP actions.

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.

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.

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.

bootloader A small set of system software that runs when the system first powers up. It loads the operating system (from the disk, network, external compact flash, or external USB flash), which loads and runs the IPS application. For AIM IPS, it boots the module from the network and assists in software installation and upgrades, disaster recovery, and other operations when the module cannot access its software.

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.

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.

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 An IPS message containing a command addressed to a specific application instance. Example control transactions include *start*, *stop*, *getConfig*.

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.

CSA MC Cisco Security Agent Management Center. 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.

CSM Cisco Security Manager, the provisioning component of the Cisco Self-Defending Networks solution. CS-Manager is fully integrated with CS-MARS.

CS-Manager See CSM.

CS-MARS Cisco Security Monitoring, Analysis and Reporting System. The monitoring component of the Cisco Self-Defending Networks solution. CS-MARS is fully integrated with CS-Manager

CVE Common Vulnerabilities and Exposures. A list of standardized names for vulnerabilities and other information security exposures maintained at http://cve.mitre.org/.

D

Database Processor 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 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 Server One of the components of the IPS.

Event Store One of the components of the IPS. A fixed-size, indexed store (30 MB) used to store IPS events.

evIdsAlert 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.

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.

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 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.

FWSM Firewall Security Module. A module that can be installed in a Catalyst 6500 series switch. It uses the **shun** command to block. You can configure the FWSM in either single mode or multi-mode.

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 *Catalyst Switch Cable, Connector, and AC Power Cord Guide.*

Gigabit Ethernet Standard for a high-speed Ethernet, approved by the IEEE (Institute of Electrical and Electronics Engineers) 802.3z standards committee in 1996.

GMT Greenwich Mean Time. Time zone at zero degrees longitude. Now called Coordinated Universal Time (UTC).

GRUB Grand Unified Bootloader.

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 NIC 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.

IDSM2 Intrusion Detection System Module. A switching module that performs intrusion detection in the Catalyst 6500 series switch.

IDS MC Management Center for IDS Sensors. A web-based IDS manager that can manage configurations for up to 300 sensors.

IME IPS Manager Express. A network management application that provides system health monitoring, events monitoring, reporting, and configuration for up to five 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.

intrusion detection system 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 Processes layer 2-related events. It also identifies malformed packets and removes them from the processing path.

Logger A component of the IPS.

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.

maintenance partition The bootable disk partition on IDSM2, from which an IPS image can be installed on the application partition. No IPS capability is available while the IDSM2 is booted into the maintenance partition.

maintenance partition image The bootable software image installed on the maintenance partition on an IDSM2. You can install the maintenance partition image only while booted into the application partition.

major update A base version that contains major new functionality or a major architectural change in the product.

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.

MEG Mega Event Generator. Signature based on the META engine. The META engine takes alerts as input rather than packets.

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.

module A removable card in a switch, router, or security appliance chassis. AIM IPS, AIP SSM, IDSM2, and NME IPS are IPS modules.

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.

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.

ND Neighbor Discovery. Neighbor 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 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.

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.

NME IPS Network Module Enhanced. An IPS module that you can install in any network module slot in the Cisco 2800 and 3800 series integrated services routers.

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, an IDSM2, 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.

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

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.

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 date 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.

PIX Firewall Private Internet Exchange Firewall. A Cisco network security device that can be programmed to block/enable addresses and ports between networks.

PKI Public Key Infrastructure. Authentication of HTTP clients using the clients X.509 certificates.

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.

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.

RDEP2 Remote Data Exchange Protocol version 2. The published specification for remote data exchange over the command and control network using HTTP and TLS.

reassembly 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.

repackage release Used to address defects in the packaging or the installer.

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.

repackage release A release that addresses defects in the packaging or the installer.

risk rating RR. An 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.

RSM Router Switch Module. A router module that is installed in a Catalyst 5000 switch. It functions exactly like a standalone router.

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 acknowledgement 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

SCP Switch Configuration Protocol. Cisco control protocol that runs directly over the Ethernet.

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.

SDEE Security Device Event Exchange. A product-independent standard for communicating security device events. It is an enhancement to RDEP. It adds extensibility features that are needed for communicating events generated by various types of security devices.

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.

Security Monitor Monitoring Center for Security. Provides event collection, viewing, and reporting capability for network devices. Used with the IDS MC.

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 SL. NTP, RPC, SMB, SNMP, and SSH.

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.

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 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.The 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 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.

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 Keeps track of system statistics such as packet counts and packet arrival rates.

Stream Reassembly 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.

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.

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/.

TCP reset interface The interface on IDSM2 that can send TCP resets. On most sensors the TCP resets are sent out on the same sensing interface on which the packets are monitored, but on IDSM2 the sensing interfaces cannot be used for sending TCP resets. On the IDSM2 the TCP reset interface is designated as port 1 with Catalyst software, and is not visible to the user in Cisco IOS software. The TCP reset action is only appropriate as an action selection on those signatures that are associated with a TCP-based service.

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 alert is sent.

Time 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.

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.

Transaction Server A component of the IPS.

Transaction Source A component of the IPS.

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.

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.

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.

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.

virus update A signature update specifically addressing viruses.

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.

VMS CiscoWorks VPN/Security Management Solution. A suite of network security applications that combines web-based tools for configuring, monitoring, and troubleshooting enterprise VPN, firewalls, network intrusion detection systems and host-based intrusion prevention systems.

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.

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.

Z

zone A set of destination IP addresses sorted into an internal, illegal, or external zone used by anomaly detection.

INDEX

Numerics

A

B

C

D

E

F

G

H

I

K

L

M

N

O

P

Q

R

S

T

U

V

W

Z