Security Configuration Guide: Protocol Support for Context-Based Access Firewall, Cisco IOS Release 15M&T

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

CONTENTS

Application Firewall-Instant Message Traffic Enforcement

The Application Firewall--Instant Message Traffic Enforcement feature enables users to define and enforce a policy that specifies which instant messenger traffic types are allowed into the network. Thus, the following additional functionality can also be enforced:

- Configuration of firewall inspection rules
- Deep packet inspection of the payload, looking for services such as text chat

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest caveats and feature information, see Bug Search Tool and the release notes for your platform and software release. To find information about the features documented in this module, and to see a list of the releases in which each feature is supported, see the feature information table at the end of this module.

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

Restrictions for Application Firewall-Instant Message Traffic Enforcement

If an instant messenger traffic enforcement policy is configured on a Cisco IOS router with a server command, traffic destined to other services (such as Telnet, FTP, SMTP) that is running on the instant message server's IP address will also be treated as IM traffic by the Cisco IOS router. Thus, access to the other services is prevented through the Cisco IOS firewall; however, this limitation is not a problem for most IM application users who are connecting from a user's network.

Information About Application Firewall-Instant Message Traffic Enforcement

What Is an Application Policy

The application firewall uses an application policy, which consists of a collection of static signatures, to detect security violations. A static signature is a collection of parameters that specify protocol conditions that must be met before an action is taken. These protocol conditions and reactions are defined by the end user via the command-line interface (CLI) to form an application policy.

Instant Messenger Application Policy Overview

Cisco IOS application firewall has been enhanced to support instant native messenger application policies. Thus, the Cisco IOS firewall can now detect and prohibit user connections to instant messenger servers for the AOL Instant Messenger (AIM), Yahoo! Messenger, and MSN Messenger instant messaging services. This functionality controls all connections for supported servies, including text, voice, video, and file-transfer capabilities. The three applications can be individually denied or permitted. Each service may be individually controlled so that text-chat service is allowed, and voice, file transfer, video, and other services are restricted. This functionality augments existing Application Inspection capability to control IM application traffic that has been disguised as HTTP (web) traffic.

Note If an instant messenger application is blocked, the connection will be reset and a syslog message will be generated, as appropriate.

How to Define and Apply an Application Policy to a Firewall for Inspection

Defining an Application Policy to Permit or Deny Instant Messenger Traffic

Use this task to create an instant messenger application firewall policy.

Note If at least one DNS name was not specified for resolution under any of the application policies for IM protocols (AOL, Yahoo, or MSN), you do not need to configure the DNS server IP address in the Cisco IOS router.

Before defining and enabling an application policy for instant messenger traffic, you must have already properly configured your router with a Domain Name System (DNS) server IP address via the **ip domain lookup** command and the **ip name-server** command.

The IP address of the DNS server configured on the Cisco IOS router must be the same as that configured on all PCs connecting to the IM servers from behind the Cisco IOS firewall.

Note Although application firewall policies are defined in global configuration mode, only one global policy for a given protocol is allowed per interface.

>

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. appfw policy-name policy-name
4. application protocol
5. **audit-trail** {**on** | **off**
6. server {permit | deny} {name string | ip-address {ip-address | range ip-address-start ip-address-end}
7. **timeout** *seconds*
8. service {default | text-chat} action {allow [alarm] | reset [alarm] | alarm}
9. **alert** {**on** | **off**}
10. **exit**
11. show appfw {configuration | dns cache} [policy policy-name]

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Router> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Router# configure terminal`	Enters global configuration mode.

	Command or Action	Purpose
Step 3	appfw policy-name policy-name **Example:** `Router(config)# appfw policy-name my_policy`	Defines an application firewall policy and enters application firewall policy configuration mode.
Step 4	application protocol **Example:** `Router(cfg-appfw-policy)# application im aol`	Allows you to configure inspection parameters for a given protocol. • *protocol* -- One of the following options: ◦ **http** (HTTP traffic will be inspected) ◦ **im** {**aol** | **yahoo** | **msn**} (Traffic for the specified instant messenger application will be inspected) This command puts the router in appfw-policy-protocol configuration mode, where "protocol" is dependent upon the specified protocol.
Step 5	**audit-trail** {**on** | **off** **Example:** `Router(cfg-appfw-policy-aim)# audit-trail on`	(Optional) Enables message logging for established or torn-down connections. If this command is not issued, the default value specified via the **ip inspect audit-trail**command will be used.
Step 6	server {permit | deny} {name string | ip-address {ip-address | range ip-address-start ip-address-end} **Example:** **Example:** `Router(cfg-appfw-policy-aim)# server permit name login.cat.aol.com`	Controls access to instant messenger servers. **Note** The **server** command helps the instant messenger application engine to recognize the port-hopping instant messenger traffic and to enforce the security policy for that instant messenger application; thus, if this command is not issued, the security policy cannot be enforced if IM applications use port-hopping techniques. To deploy IM traffic enforcement policies effectively, it is recommended that you issue the appropriate **server** command.
Step 7	**timeout** *seconds* **Example:** `Router(cfg-appfw-policy-aim)# timeout 30`	(Optional) Specifies the elapsed length of time before an inactive connection is torn down. • *seconds* --Available timeout range: 5 to 43200 (12 hours). If this command is not issued, the default value specified via the **ip inspect tcp idle-time**command will be used. **Note** Some IM applications continue to send "keepalive-like" packets that effectively prevent timeout even when the user is idle.

	Command or Action	Purpose
Step 8	service {default | text-chat} action {allow [alarm] | reset [alarm] | alarm} **Example:** `Router(cfg-appfw-policy-aim)# service default action reset`	(Optional) Specifies an action when a specific service is detected in the instant messenger traffic. • If a specific action is not specified for a service, the **service default**command will be performed. • If the **service default** command is not specified for an application, the action is considered “reset” by the system.
Step 9	**alert** {**on** | **off**} **Example:** `Router(cfg-appfw-policy-aim)# alert on`	(Optional) Enables message logging when events, such as the start of a text-chat, begin. If this parameter is not configured, the global setting for the **ip inspect alert-off** command will take effect.

Step 10	**exit**
	
	Example:
	`Router(cfg-appfw-policy-aim)# exit`
	
	Example:
	`Router(cfg-appfw-policy)# exit`
	
	Example:
	`Router(config)# exit`
		(Optional) Exits application firewall policy *protocol* configuration mode, application firewall policy configuration mode, and global configuration mode.

Step 11	show appfw {configuration | dns cache} [policy policy-name] **Example:** `Router# show appfw dns cache policy abc`	(Optional) Displays the IP addresses that have been resolved by the DNS server and stored in the DNS cache of the IM traffic policy enforcement component of the Cisco IOS router. • If you don’t indicate a specific policy via the **policy** *policy-name* option, IP addresses gathered for all DNS names for all policies are displayed.

- Troubleshooting Tips, page 5
- What to Do Next, page 6

Troubleshooting Tips

Resolved IP addresses are never “timed out” and not automatically removed from the DNS cache. Thus, if you find an obsolete IP address in the instant messenger database (DNS cache), you can issue the **clear appfw dns cache** command to remove the IP address and prevent the address from being interpreted by the router as that of an IM server.

Always allow a couple of minutes for the DNS cache to populate after configuring the **server** command (with the **name** *string* option) in an application firewall policy for IM applications.

If you do not want the DNS resolver to send periodic queries, do not use the **server** command (with the **name** *string* option); instead, use the **server** command (with the **ip address** option).

If you issue the **server** command (with the **name** *string* option), ensure that you specify the name of every DNS server for an IM application in your policy. Always be alert to new names.

What to Do Next

After you have successfully defined an application policy for instant message traffic inspection, you must apply the policy to an inspection rule. Thereafter, the inspection rule must be applied to an interface. For information on completing this task, see the section "Applying an Instant Messenger Traffic Application Policy to a Firewall for Inspection."

Applying an Instant Messenger Traffic Application Policy to a Firewall for Inspection

Use this task to apply an IM application policy to an inspection rule, followed by applying the inspection rule to an interface.

You must have already defined an application policy (as shown in the section "Defining an Application Policy to Permit or Deny Instant Messenger Traffic").

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **ip inspect name** *inspection-name* **appfw** *policy-name*
4. **interface** *type number*
5. **ip inspect** *inspection-name* **in** | **out**}
6. **exit**
7. **exit**
8. show appfw configuration [name]

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Router> enable`	Enables privileged EXEC mode. • Enter your password if prompted.

	Command or Action	Purpose
Step 2	**configure terminal** **Example:** `Router# configure terminal`	Enters global configuration mode.
Step 3	**ip inspect name** *inspection-name* **appfw** *policy-name* **Example:** `Router(config)# ip inspect name firewall appfw mypolicy`	Defines a set of inspection rules for the application policy. • *policy-name* --Must match the policy name specified via the **appfw policy-name** command.
Step 4	**interface** *type number* **Example:** `Router#(config)# interface FastEthernet0/0`	Configures an interface type and enters interface configuration mode.
Step 5	**ip inspect** *inspection-name* **in** | **out**} **Example:** `Router#(config-if)# ip inspect firewall in`	Applies the inspection rules (defined in Step 3) to all traffic entering the specified interface. • The *inspection-name* argument must match the inspection name defined via the **ip inspect name** command.
Step 6	**exit** **Example:** `Router#(config-if)# exit`	`Exits interface configuration mode.`
Step 7	**exit** **Example:** `Router(config)# exit`	`Exits global configuration mode.`
Step 8	show appfw configuration [name] **Example:** `Router# show appfw configuration`	`(Optional) Displays application firewall policy configuration information.`

Configuration Examples for Setting Up an Instant Messenger Traffic Inspection Engine

Example Instant Messenger Application Policy Configuration

The following example shows to configure application policy “my-im-policy,” which allows text-chat for Yahoo! instant messenger users and blocks instant messenger traffic for all other users:

```
appfw policy-name my-im-policy
 application http
  port-misuse im reset
!
 application im yahoo
  server permit name scs.msg.yahoo.com
  server permit name scsa.msg.yahoo.com
  server permit name scsb.msg.yahoo.com
  server permit name scsc.msg.yahoo.com
  service text-chat action allow
  service default action reset
!
 application im aol
  server deny name login.oscar.aol.com
!
 application im msn
  server deny name messenger.hotmail.com
!
ip inspect name test appfw my-im-policy
interface FastEthernet0/0
 description Inside interface
 ip inspect test in
```

The **port-misuse im** command blocks all the three IM applications going through the HTTP protocol. It is always recommended that you block IM activity through HTTP and allow IM traffic to pass, if at all, through its native port.

The **server permit**commands help to identify all the servers for Yahoo! messenger services. A connection to any one of the specified servers will be recognized by the firewall as a Yahoo! IM session--even if the Yahoo! client uses port-hopping techniques (which can be accomplished by using server port-numbers such as 25 instead of the standard 5050.)

If a **server permit** command is not issued within the **application im yahoo** command, the Cisco IOS firewall will classify only the traffic going to server port 5050 as Yahoo! messenger traffic. Because the port classification scheme breaks if any of the Yahoo! clients are configured to use a port other than 5050, it is more reliable to have **server permit** command entries instead of relying on the port classification method.

The **server deny** commands under other IM applications deny connection to respective servers. This action operates at the network layer connection level--not at the application session level. When traffic is denied, the TCP connection to the server is denied, no data traffic is allowed, and all packets are dropped in the firewall.

Additional References

Related Documents

Related Topic	Document Title
Cisco IOS commands	Cisco IOS Master Commands List, All Releases
Firewall commands: complete command syntax, command mode, defaults, usage guidelines, and examples	*Cisco IOS Security Command Reference*
Application firewall: configure a firewall to detect and prohibit HTTP connections	"HTTP Inspection Engine"

Standards

Standard	Title
None	--

MIBs

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

RFCs

RFC	Title
None	--

Technical Assistance

Description	Link
The Cisco Support and Documentation website provides online resources to download documentation, software, and tools. Use these resources to install and configure the software and to troubleshoot and resolve technical issues with Cisco products and technologies. Access to most tools on the Cisco Support and Documentation website requires a Cisco.com user ID and password.	http://www.cisco.com/cisco/web/support/index.html

Feature Information for Application Firewall-Instant Message Traffic Enforcement

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

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

Table 1 Feature Information for Application Firewall-Instant Message Traffic Enforcement

Feature Name	Releases	Feature Information
Application Firewall--Instant Message Traffic Enforcement	12.4(4)T	The Application Firewall--Instant Message Traffic Enforcement feature enables users to define and enforce a policy that specifies which instant messenger traffic types are allowed into the network. The following commands were introduced or modified: **alert**, **application (application firewall policy)**, **audit-trail**, **clear appfw dns cache**, **server (application firewall policy)**, **service**, **show appfw**, **timeout**.

E-mail Inspection Engine

The E-mail Inspection Engine feature allows the Cisco IOS Firewall to inspect Post Office Protocol 3 (POP3) and Internet Message Access Protocol (IMAP) e-mail, in addition to Simple Mail Transfer Protocol (SMTP) and Extended Simple Mail Transfer Protocol (ESMTP) e-mail which were previously supported.

The secure-login enhancement allows people to download external POP3 e-mail only if authentication methods are secure.

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest caveats and feature information, see Bug Search Tool and the release notes for your platform and software release. To find information about the features documented in this module, and to see a list of the releases in which each feature is supported, see the feature information table at the end of this module.

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

Prerequisites for E-mail Inspection Engine

- Configure CBAC.
- Enable SSL VPN tunnels.

Information About E-mail Inspection Engine

E-mail Inspection Engine Operation

The client/server communication is validated from the time the TCP connection is initialized until the client is authenticated. The Cisco IOS Firewall uses a state router to track each stage of authentication. After the client is authenticated, the Cisco IOS Firewall allows all the client/server commands without further L7 inspection. TCP L4 inspection continues until the connection is closed. At the end of the e-mail session when the client host quits and before the TCP connection is closed, no further client/server interaction is allowed unless the client is reauthenticated.

During the authentication, any unrecognized command causes the Cisco IOS Firewall to drop the packet and close the connection.

If encryption is negotiated between the client and server control channel, no further validation occurs.

An e-mail client logging in from a nonsecure location may need to use encryption for authentication. For information about secure logins, see the description of the **secure-login** keyword of the **ip inspect name** command.

Inspection

Context Based Access Control (CBAC) inspects traffic that travels through the firewall to discover and manage state information for TCP and User Datagram Protocol (UDP) sessions. This state information is used to create temporary openings in the firewall's access lists to allow return traffic and additional data connections for permissible sessions.

Inspecting packets at the application layer and maintaining TCP and UDP session information provides CBAC with the ability to detect and prevent certain types of network attacks such as SYN-flooding. A SYN-flood attack occurs when a network attacker floods a server with a barrage of requests for connection and does not complete the connection. The resulting volume of half-open connections can overwhelm the server, causing it to deny service to valid requests. Network attacks that deny access to a network device are called denial-of-service (DoS) attacks.

POP3

The Post Office Protocol, Version 3 (POP3) is used to receive e-mail that is stored on a mail server. Unlike IMAP, POP only retrieves mail from a remote host.

POP3 works best when there is only one computer because it supports "offline" message access where messages are downloaded and then deleted from the mail server. This mode of access is not compatible with access from multiple computers because it tends to sprinkle messages across all the computers used for mail access.

With POP3-based e-mail clients, messages are downloaded to the user's local message store and can also be deleted from the mail server. Deletion is optional in most clients. When a new voice message arrives, the subscriber's only immediate notification is the activation of the MWI on the phone. New messages are displayed in the Inbox only after the client's local message store is updated with the Exchange message store. After the subscriber downloads new messages, the message state automatically changes from "new" to "read" on the server, even though the subscriber has not actually listened to the voice messages. MWIs

on the subscriber's phone are extinguished, and the message state between the TUI and the subscriber's Inbox are not synchronized.

IMAP Protocol

The Internet Message Access Protocol (IMAP) is a method of accessing electronic mail or bulletin board messages that are kept on a mail server that may be shared. It permits a "client" e-mail program to access remote messages as though they were local. For example, e-mail stored on an IMAP server can be retrieved, sent, and managed from a desktop computer at home, from a workstation at the office, or from a laptop without transferring messages or files back and forth between the computers.

Only the message header and sender information are displayed in the Inbox until the user downloads the entire message, including attachments, from the server. When a new voice message arrives, the subscriber's only immediate notification is the activation of the Message Waiting Indication (MWI) on the phone. New messages are displayed in the Inbox only after the client's local message store is updated with the Exchange message store. When the subscriber listens to a new message by using the telephone user interface (TUI), the MWI is extinguished. In this case again, the message state is not updated in the Inbox until the client's message store is refreshed. However, if the subscriber uses an installed multimedia player to listen to the WaveForm Audio (WAV) attachment from the e-mail client's Inbox, message state changes are automatically synchronized with the TUI.

How message state changes are conveyed to the Cisco Unity subscriber, and how these changes are synchronized with the TUI, depend on whether the subscriber's e-mail client is configured to use POP3 or IMAP4 to access Exchange.

Client Command Validation

The Cisco IOS Firewall authenticates an e-mail client accessing an IMAP or POP3 server before allowing complete access into the server. The firewall searches the IMAP/POP3 TCP stream for valid protocol commands. If the client's commands are outside the protocol's definition, the Cisco IOS Firewall drops the packets and resets the connection.

Client command validation is typically needed in a DeMilitarized Zone (DMZ). Client access is allowed into the DMZ only if the e-mail server validates the user authentication. After the client is authenticated, the client becomes a trusted user and access is permitted.

SMTP

The Simple Mail Transfer Protocol (SMTP) is used to transfer e-mail between servers and clients on the Internet. E-mail clients and mail servers that use protocols other than Message Application Programming Interface (MAPI) can use the SMTP protocol to transfer a message from a client to the server, and then forward it to a message recipient's server. To retrieve, send, and manage these messages from the e-mail client use POP3 or IMAP4.

Cisco Unity uses SMTP to route voice messages via the Internet Voice Connector (IVC) gateway between other Exchange servers that are not connected by using a Site Message Connector. There is an IVC gateway on either end of the SMTP connection between Exchange servers. This ensures that MAPI message attributes survive the outbound transit between SMTP connections. It also ensures that the MIME-encoded attributes survive the inbound transit, and are included with the message stored in the Exchange message store.

SSL

The Secure Socket Layer (SSL) protocol is the standard protocol that delivers secure content over the Internet. It is a point-to-point security protocol that secures communication between a client and a server. SSL usually does not require a special client (that is, a Web browser often will suffice) and it does not require any additional operating system software.

SSL includes client and server authentication and data encryption for a limited set of applications (for example, the Web, e-mail, news, and file transfer). SSL is useful for securing e-commerce transactions over the Internet, and the protocol is well suited for extranets and remote access because it is relatively simple to deploy.

How to Configure E-mail Inspection Engine

Configuring Firewall Inspection of POP3 or IMAP E-mail

To allow the Cisco IOS Firewall to inspect POP3 or IMAP e-mail, perform the following task.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **ip inspect name** *inspection-name protocol* [**alert** {**on** | **off**}] [**audit-trail** {**on** | **off**}][**reset**] [**secure-login**] [**timeout** *seconds*]
4. **interface** *type slot/port*
5. **ip inspect name** *inspection-name* {**in** | **out**}
6. **exit**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Router> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Router# configure terminal`	Enters global configuration mode.

	Command or Action	Purpose
Step 3	**ip inspect name** *inspection-name protocol* [**alert** {**on** | **off**}] [**audit-trail** {**on** | **off**}][**reset**] [**secure-login**] [**timeout** *seconds*] **Example:** `Router(config)# ip inspect name mail-guard pop3`	Defines a set of inspection rules.
Step 4	**interface** *type slot/port* **Example:** `Router(config-if)# interface 1/0`	Configures an interface type.
Step 5	**ip inspect name** *inspection-name* {**in** | **out**} **Example:** `Router(config-if)# ip inspect name mail-guard in`	Enables the Cisco IOS Firewall on an interface.
Step 6	**exit** **Example:** `Router(config-if)# exit`	Exits interface configuration mode and returns to global configuration mode.

Verifying the E-mail Inspection Engine Configuration

To verify the E-mail Inspection Engine configuration, perform the following task.

Note Effective with Cisco IOS Release 12.4(20)T, the **debug ip inspect** command is replaced by the **debug policy-firewall** command. See the Cisco IOS Debug Command Reference for more information.

SUMMARY STEPS

1. **debug ip inspect imap**
2. **debug ip inspect pop3**
3. **show ip inspect** {**name** *inspection-name* \| **config** \| **interfaces** \| **session** [**detail**] \| **all**}

DETAILED STEPS

Step 1 **debug ip inspect imap**

Use this command to display messages about Cisco IOS Firewall events related to IMAP protocol e-mail messages.

Example:

```
Router# debug ip inspect imap
```

Step 2 **debug ip inspect pop3**

Use this command to display messages about Cisco IOS Firewall events related to POP3 protocol e-mail messages.

Example:

```
Router# debug ip inspect pop3
```

Step 3 **show ip inspect** {**name** *inspection-name* | **config** | **interfaces** | **session** [**detail**] | **all**}

Use this command to view CBAC configuration and session information.

Example:

```
Router# show ip inspect
Session audit trail is disabled
one-minute (sampling period) thresholds are [400:500] connections
max-incomplete sessions thresholds are [400:500]
max-incomplete tcp connections per host is 50. Block-time 0 minute.
tcp synwait-time is 30 sec -- tcp finwait-time is 5 sec
tcp idle-time is 3600 sec -- udp idle-time is 30 sec
dns-timeout is 5 sec
Inspection Rule Configuration
 Inspection name mail-guard
  tcp timeout 3600
  tdp timeout 30
  ftp timeout 3600
```

Configuration Examples for E-mail Inspection Engine

Example Configuring IMAP and POP3 Protocol E-mail

The following example configures the Cisco IOS Firewall inspection of IMAP and POP3 protocol e-mail:

```
configure terminal
ip inspect name mail-guard pop3
ip inspect name mail-guard imap
exit
```

The following commands enable this functionality on an interface:

```
configure terminal
interface 1/0
ip inspect name mail-guard in
exit
```

Additional References

Related Documents

Related Topic	Document Title
Cisco IOS commands	Cisco IOS Master Commands List, All Releases
Security commands	Cisco IOS Security Command Reference
IMAP and POP3	White Paper: *Deploying Cisco Unity in Diverse Messaging Environments (All Versions with Microsoft Exchange)*
CBAC	*Cisco IOS Security Command Reference* Configuring Context-based Access Control

Standards

Standards	Title
No new or modified standards are supported by this feature, and support for existing standards has not been modified by this feature.	--

MIBs

MIBs	MIBs Link
No new or modified MIBs are supported by this feature, and support for existing MIBs has not been modified by this feature.	To locate and download MIBs for selected platforms, Cisco IOS releases, and feature sets, use Cisco MIB Locator found at the following URL: http://www.cisco.com/go/mibs

RFCs

RFCs	Title
RFC 1939	J Myers and M. Rose, “Post Office Protocol, Version 3 (POP3),” May 1996.
RFC 3501	M. Crispin, *“Internet Message Access Protocol (IMAP4rev1,”* March 2003.

Technical Assistance

Description	Link
The Cisco Support and Documentation website provides online resources to download documentation, software, and tools. Use these resources to install and configure the software and to troubleshoot and resolve technical issues with Cisco products and technologies. Access to most tools on the Cisco Support and Documentation website requires a Cisco.com user ID and password.	http://www.cisco.com/cisco/web/support/index.html

Feature Information for E-mail Inspection Engine

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

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

Table 2 Feature Information for E-mail Inspection Engine

Feature Name	Releases	Feature Information
E-mail Inspection Engine	12.3(14)T	The E-mail Inspection Engine feature allows the Cisco IOS Firewall to inspect Post Office Protocol 3 (POP3) and Internet Message Access Protocol (IMAP) e-mail, in addition to Simple Mail Transfer Protocol (SMTP) and Extended Simple Mail Transfer Protocol (ESMTP) e-mail which were previously supported. The secure-login enhancement allows people to download external POP3 e-mail only if authentication methods are secure. The following commands were introduced or modified: **debug ip inspect**, **ip inspect name**, **show ip inspect**.

Glossary

authentication --Process during which any unrecognized command causes the Cisco IOS Firewall to drop the packet and close the connection.

CBAC --Context-Based Access Control. A Cisco IOS Firewall set feature that scrutinizes source and destination addresses to enhance security for TCP and UDP applications that use well-known ports, such as FTP and e-mail traffic.

ESMTP --Extended Simple Mail Transfer Protocol. An extended version of the Simple Mail Transfer Protocol (SMTP), which includes additional functionality, such as delivery notification and session delivery.

IMAP --Internet Message Access Protocol. A method of accessing e-mail or bulletin board messages kept on a mail server that can be shared. IMAP permits client e-mail applications to access remote message stores as if they were local without actually transferring the message.

POP --Post Office Protocol. A protocol that client e-mail applications use to retrieve mail from a mail server.

SMTP --Simple Mail Transfer Protocol. An Internet protocol providing e-mail services.

SSL --Secure Socket Layer Protocol. This protocol is used to deliver secure information over the Internet.

state router --A router that tracks the client/server commands until the client is authenticated.

TCP --Transmission Control Protocol. A connection-oriented transport layer protocol that provides reliable full-duplex data transmission. TCP is part of the TCP/IP protocol stack.

UDP --User Datagram Protocol. A connectionless transport-layer protocol for exchanging datagrams without acknowledgments or guaranteed delivery.

VPN --Virtual Private Network. A network that enables IP traffic to travel securely over a public TCP/IP network by encrypting all traffic from one network to another. A VPN network uses “tunneling” to encrypt all information at the IP level.

ESMTP Support for Cisco IOS Firewall

The ESMTP Support for Cisco IOS Firewall feature enhances the Cisco IOS Firewall to support Extended Simple Mail Transport Protocol (ESMTP), allowing customers who install mail servers behind Cisco IOS firewalls to install their servers on the basis of ESMTP (instead of Simple Mail Transport Protocol [SMTP]).

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest caveats and feature information, see Bug Search Tool and the release notes for your platform and software release. To find information about the features documented in this module, and to see a list of the releases in which each feature is supported, see the feature information table at the end of this module.

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

Prerequisites for ESMTP Support for Cisco IOS Firewall

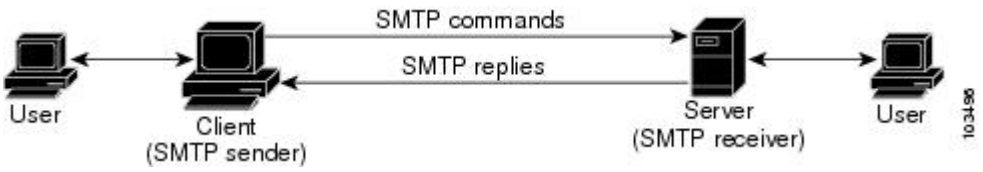
To enable this feature, your Cisco IOS image must contain the Cisco IOS firewall.

Information About ESMTP Support for Cisco IOS Firewall

SMTP Functionality Overview

SMTP inspection provides a basic method for exchanging e-mail messages. The figure below and the following steps outline a basic SMTP session.

Figure 1 Sample SMTP Exchange Topology



After a user sends an e-mail request to the client (the "SMTP sender"), the client established a TCP channel with the server (the "SMTP receiver"). Thereafter, the client and the server exchange SMTP commands and responses until the mail transaction is complete. The steps of typical SMTP transaction are as follows:

1. The client establishes a TCP connection with the server.
2. The client sends a HELO command with its domain name. If the server can accept mail from that domain name, it responds with a 250 reply code, which allows the client to continue with the mail transaction. (If the server does not respond with a 250 reply code, the client will send a QUIT command and terminate the TCP session.)
3. The client sends the MAIL command, indicating who initiated the mail. If the server accepts the mail, it responds with an OK reply. Then, the client sends the RCPT command, identifying the recipient of the mail. If the server accepts mail for the specified recipient, it responds with an OK reply; if the server cannot accept mail for the specified recipient, it rejects the recipient but not the entire transaction. (Several recipients can be negotiated.)
4. After the list of recipients has been negotiated between the client and the server, the client sends a DATA command. If the server is ready to receive data, it responds with a 354 reply code. If the server is not ready to receive data, it responds with a error reply, and the client terminates the transaction.
5. The client sends mail data ending with a special sequence. When the server sees the end of the message, it sends a 250 code reply.
6. The client sends a QUIT command, waits for the server to respond, then terminates the session.

ESMTP Overview

Like SMTP, ESMTP inspection provides a basic method for exchanging e-mail messages. Although an ESMTP session is similar to SMTP, there is one difference--the EHLO command.

After the TCP connection has been established between the client (the ESMTP sender) and the server (the ESMTP receiver), the client sends the EHLO command (instead of the HELO command that is used for SMTP). If the server does not support ESMTP, it sends a failure reply to the client because it did not recognize the EHLO command. If it supports ESMTP, the server responds with the code 250 and a list of extensions that the server supports. (Refer to RFC 1869 for an explanation of the extensions that your server may support.)

The server may send any of the following error codes if it supports ESMTP but is unable to function as normal:

- Error code 501--The server recognizes the EHLO command but is unable to accept it.
- Error code 502--The server recognizes the EHLO command but does not implement it.
- Error code 554--The server is unable to list the service extensions it supports.

If the client receives any of these error codes, it should issue the HELO command to revert to SMTP mode or issue the QUIT command to end the session.

After the client receives a successful response to the EHLO command, it will work the same way as SMTP, except that the client may issue new extended commands, and it may add a few parameters to the MAIL FROM and REPT TO commands.

SMTP Firewall and ESMTP Firewall Comparison

Although a SMTP firewall and an ESMTP firewall support the same functionality--command inspection, session conversion, and Intrusion Detection System (IDS) detection--slight variations exist between the protocols. The table below explains the firewall functionality and protocol-specific differences.

Table 3 SMTP and ESMTP Firewalls Functionality Comparison

Functionality	SMTP Firewall Description	ESMTP Firewall Description
Command Inspection	The SMTP firewall inspects commands for illegal commands. Illegal commands found in a packet are modified to an “xxxx” pattern and forwarded to the server. This process causes the server to send a negative reply, forcing the client to issue a valid command. An illegal SMTP command is any command except the following: DATA, HELO, HELP, MAIL, NOOP, QUIT, RCPT, RSET, SAML, SEND, SOML, and VRFY. **Note** Prior to Cisco IOS Release 12.3(7)T, an SMTP firewall will reset the TCP connection upon detection of an illegal command. That is, an SMTP firewall no longer resets the TCP connection upon detecting an illegal command.	ESMTP command inspection is the same as SMTP command inspection, except that ESMTP supports three additional commands--AUTH, EHLO, and ETRN. An illegal ESMTP command is any command except the following: AUTH, DATA, EHLO, ETRN, HELO, HELP, MAIL, NOOP, QUIT, RCPT, RSET, SAML, SEND, SOML, and VRFY.

Functionality	SMTP Firewall Description	ESMTP Firewall Description
Parameter Inspection	Not applicable.	The ESMTP firewall inspects the following extensions by performing deeper command inspection: • Message Size Declaration (SIZE) • Remote Queue Processing Declaration (ETRN) • Binary MIME (BINARYMIME) • Command Pipelining • Authentication • Delivery Status Notification (DSN) • Enhanced Status Code (ENHANCEDSTATUSCODE) • 8bit-MIMEtransport (8BITMIME) **Note** All other extensions, including private extensions, are not supported.
EHLO Reply Inspection	Not applicable.	The ESMTP firewall inspects the EHLO reply, which contains a list of SMTP extensions that the server supports. Any unsupported extension that is found in the server's reply will be replaced with the "XXXX" pattern, which labels that extension "private." Thus, the client will no longer use the unsupported extension.

Functionality	SMTP Firewall Description	ESMTP Firewall Description
ESMTP to SMTP Session Conversion	The SMTP firewall forces a client that initiates an ESMTP session to use SMTP. When a client attempts to initiate an ESMTP session by sending the ELHO command, the firewall treats the EHLO command as an illegal command and modified it to the "xxxx" pattern. This response causes the server to send a 5xx code reply, forcing the client to revert to SMTP mode. **Note** Prior to Cisco IOS Release 12.3(7)T, the firewall intercepts the EHLO command and changes it to the NOOP command. The server responds with a 250 code reply. The firewall intercepts the response and modifies it to 502 code reply, which tells the client that the EHLO command is not supported.	Not applicable (because EHLO is supported in ESMTP).
IDS Signature Detection	The SMTP and ESMTP firewalls scan for a set of hard-coded IDS signatures. There are 11 signatures--6 are hard coded in the firewall and are enabled by default. The other 5 signatures remain in the IDS code and are disabled by default.	
Command Pipelining	Not available. (The client sends a command to the server and must wait for a reply before sending another command.)	An ESMTP firewall can inspect commands that are in the pipeline. That is, commands that are sent before a response is received are inspected.

Functionality	SMTP Firewall Description	ESMTP Firewall Description
Resetting a Connection	Both SMTP and ESMTP firewalls will always send a "5xx" error code and close the connection upon detection of an unsupported parameter or an IDS signature in a command. That is, the firewall sends an appropriate reply code and closes the connection with proper TCP closing sequence packets (such as FIN or FIN+ACK) so the client does not continually attempt to send the same message. **Note** Prior to Cisco IOS Release 12.3(7)T, an SMTP firewall will reset the TCP connection upon detection of an illegal command or IDs signature. This behavior causes the client to keep trying to send the same message for up to 4 days (which is when the original message is bounced back to the user).	

How to Configure a Firewall to Support ESMTP

Configuring a Firewall for ESMTP Inspection

Use this task to configure a Cisco IOS Firewall to inspect an ESMTP session and command sequence.

Note SMTP and ESMTP cannot exist simultaneously. If SMTP is already configured, an attempt to configure ESMTP will result in the error message, "%ESMTP cannot coexist with SMTP, please unconfigure SMTP and try again...." If ESMTP is already configured, an attempt to configure SMTP will result in the error message, "%SMTP cannot coexist with ESMTP, please unconfigure ESMTP and try again...."

The following example illustrates how the router will react if you attempt to configure both protocols:

```
Router(config)# ip inspect name mail-guard smtp
Router(config)# ip inspect name mail-guard esmtp
ESMTP cannot coexist with SMTP, please unconfigure SMTP and try again...

Router(config)# end
Router# show running-config
.
.
.
ip inspect name mail-guard smtp
.
.
.

>
```

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. ip inspect name inspection-name {smtp | esmtp} [alert {on | off}] [audit-trail {on | off}] [max-data number] [timeout seconds]
4. **interface** *type number*
5. **ip inspect** *inspection-name* {**in** | **out**}

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Router> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Router# configure terminal`	Enters global configuration mode.

	Command or Action	Purpose
Step 3	ip inspect name inspection-name {smtp | esmtp} [alert {on | off}] [audit-trail {on | off}] [max-data number] [timeout seconds] **Example:** `Router(config)# ip inspect name test esmtp`	Configures inspection of a SMTP or an ESMTP session.
Step 4	**interface** *type number* **Example:** `Router(config)# interface ethernet0`	Configures an interface type and enters interface configuration mode.

Step 5	**ip inspect** *inspection-name* {**in** | **out**}
	
	Example:
	
	`Router(config-if)#`
	`ip inspect test in`
		Applies an inspection rule to an interface.

- Troubleshooting Tips, page 28
- What to Do Next, page 29

Troubleshooting Tips

To view and verify the inspection configuration, status, or session information, you can use any of the following EXEC commands:

- **show ip inspect name** *inspection-name* --Shows a particular configured inspection rule.
- **show ip inspect session** --Shows existing sessions that are currently being tracked and inspected by the firewall.
- **show ip inspect all** --Shows all inspection configuration and all existing sessions that are currently being tracked and inspected by the firewall.

Alert Messages

The existing SMTP-related alert message will not change. This message is logged every time the firewall detects an illegal or unsupported command. The message format is as follows:

```
FW-3-SMTP_INVALID_COMMAND: Invalid SMTP command (%s) (total %d chars) from initiator (%i:
%d)
```

A new alert message is added. This message is logged whenever the firewall detects an illegal parameter in an SMTP command. The message includes the address and port of the sender as well as the illegal parameter. The message format is as follows:

FW-3-SMTP_INVALID_PARAMETER: Invalid SMTP parameter (%s) from initiator (%i:%d)

What to Do Next

To provide a record of network access through the firewall, including illegitimate access attempts, and inbound and outbound services, you should turn on logging and audit trail. For information on completing this task, refer to the section “Configuring Logging and Audit Trail ” in the cha pter “Configuring Context-Based Access Control ” in the *Cisco IOS Security Configuration Guide*

Configuration Examples for Firewall ESMTP Support

Example ESMTP Inspection Configuration

The following example shows how to configure inspection of ESMTP traffic:

```
Router# configure terminal
Router(config)# ip inspect name mail-guard esmtp timeout 30
```

Additional References

Related Documents

Related Topic	Document Title
Cisco IOS commands	Cisco IOS Master Commands List, All Releases
Security commands	*Cisco IOS Security Command Reference*

Standards

Standards	Title
None	--

MIBs

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

RFCs

RFCs	Title
RFC 821	Simple Mail Transfer Protocol
RFC 1652	SMTP Service Extension for 8bit-MIMEtransport
RFC 1845	SMTP Service Extension for Checkpoint/Restart
RFC 1869	SMTP Service Extensions
RFC 1870	SMTP Service Extension for Message Size Declaration
RFC 1891	SMTP Service Extension for Delivery Status Notifications
RFC 1985	SMTP Service Extension for Remote Message Queue Starting
RFC 2034	SMTP Service Extension for Returning Enhanced Error Codes
RFC 2554	SMTP Service Extension for Authentication
RFC 2645	ON-DEMAND MAIL RELAY (ODMR) SMTP with Dynamic IP Addresses
RFC 2920	SMTP Service Extension for Command Pipelining
RFC 3030	SMTP Service Extensions for Transmission of Large and Binary MIME Messages
RFC 3207	SMTP Service Extension for Secure SMTP over Transport Layer Security

Technical Assistance

Description	Link
The Cisco Support and Documentation website provides online resources to download documentation, software, and tools. Use these resources to install and configure the software and to troubleshoot and resolve technical issues with Cisco products and technologies. Access to most tools on the Cisco Support and Documentation website requires a Cisco.com user ID and password.	http://www.cisco.com/cisco/web/support/index.html

Feature Information for ESMTP Support for Cisco IOS Firewall

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

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

Table 4 Feature Information for ESMTP Support for Cisco IOS Firewall

Feature Name	Releases	Feature Information
ESMTP Support for Cisco IOS Firewall	12.3(7)T	The ESMTP Support for Cisco IOS Firewall feature enhances the Cisco IOS Firewall to support Extended Simple Mail Transport Protocol (ESMTP), allowing customers who install mail servers behind Cisco IOS firewalls to install their servers on the basis of ESMTP (instead of Simple Mail Transport Protocol [SMTP]). The following commands were introduced or modified: **ip inspect name**.

Firewall N2H2 Support

The Firewall N2H2 Support feature provides users with an additional option when choosing the URL filter vendor. Just like the Websense URL filtering server, N2H2 interacts with your Cisco IOS firewall (also known as Cisco Secure Integrated Software [CSIS]) to allow you to prevent users from accessing specified websites on the basis of some policy. The Cisco IOS firewall works with the N2H2 Internet Filtering Protocol (IFP) server to know whether a particular URL should be allowed or denied (blocked).

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest caveats and feature information, see Bug Search Tool and the release notes for your platform and software release. To find information about the features documented in this module, and to see a list of the releases in which each feature is supported, see the feature information table at the end of this module.

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

Restrictions for Firewall N2H2 Support

N2H2 IFP Server Requirement

To enable this feature, you must have at least one N2H2 server; however, two or more N2H2 servers are preferred. Although there is no limit to the number of N2H2 servers you may have, and you can configure as many servers as you wish, only one server will be active at any given time--the primary server. URL lookup requests will be sent only to the primary server.

URL Filtering Support Restriction

This feature supports only one active URL filtering scheme at a time. (Before enabling N2H2 URL filtering, you should always ensure that there is not another URL filtering scheme configured, such as Websense.)

Username Restriction

N2H2 requires the username to be supplied with the URL lookup request. Thus, the user-based policy will not work with N2H2 because the current Cisco IOS software does not retrieve the username.

Protocol Used to Communicate Between Firewall and N2H2 Server Restriction

TCP is currently the only protocol used to communicate between the Cisco IOS firewall (UNIX FileSystem [UFS]) and the N2H2 server.

Information About Cisco N2H2 Support

Benefits of Firewall N2H2 Support

The Cisco IOS Firewall N2H2 Support feature provides an Internet management application that allows you to control web traffic for a given host or user on the basis of a specified security policy. In addition, the following functions are available in this feature:

Primary and Secondary Servers

When users configure multiple N2H2 servers, the firewall will use only one server at a time--the primary server; all other servers are called secondary servers. When the primary server becomes unavailable for any reason, it becomes a secondary server and one of the secondary servers becomes the primary server.

A firewall marks a primary server as down when sending a request to or receiving a response from the server fails. When a primary server goes down, the system will go to the beginning of the configured servers list and try to activate the first server on the list. If the first server on the list is unavailable, it will try the second server on the list; the system will keep trying to activate a server until it is successful or until it reaches the end of the server list. If the system reaches the end of the server list, it will set a flag indicating that all of the servers are down, and it will enter allowmode.

When all the servers are down and the system is in allow mode, a periodic event that occurs for each minute will trace through the server list, trying to bring up a server by opening a TCP connection. If the TCP connection is successfully opened, the server is considered to be up, and the system will return to operational mode.

IP Cache Table

This function provides an IP cache table that contains the IP addresses of web servers whose underlying URLs can be accessed by all users and hosts.

The caching algorithm involves three parameters--the maximum number of IP addresses that can be cached, an idle time, and an absolute time. The algorithm also involves two timers--idle timer and absolute

timer. The idle timer is a small periodic timer (1 minute) that checks to see whether the number of cached IP addresses in the cache table exceeds 80 percent of the maximum limit. If the cached IP addresses have exceeded 80 percent, it will start removing idle entries; if it has not exceeded 80 percent, it will quit and wait for the next cycle. The absolute timer is a large periodic timer (1 hour) that is used to remove all of the elapsed entries. (The age of an elapsed entry is greater than the absolute time.) An elapsed entry will also be removed during cache lookup.

The idle time value is fixed at 10 minutes. The absolute time value is taken from the N2H2 lookup response, which is often greater than 15 hours. The absolute value for cache entries made out of exclusive-domains is 12 hours. The maximum number of cache entries is configurable.

To configure cache table parameters, use the **ip urlfilter cache** command.

Packet Buffering

This function allows you to increase the maximum number of HTTP responses that a Cisco IOS firewall can hold. If the HTTP responses arrive prior to an N2H2 server reply, this buffering scheme allows your firewall to store a maximum of 200 HTTP responses. (After 200 responses have been reached, the firewall will drop further responses.) The responses will remain in the buffer until an allow or deny message is received from N2H2: if the status indicates that the URL is allowed, the firewall will release the HTTP responses in the buffer to the browser of the end user; if the status indicates that the URL is blocked, the firewall will discard the HTTP responses in the buffer and close the connection to both ends. This function prevents numerous HTTP responses from overwhelming your system.

To configure the maximum number of HTTP responses for your firewall, use the **ip urlfilter max-resp-pak** command.

Exclusive Domains

This function provides a configurable list of domain names so that the Cisco IOS firewall does not have to send a lookup request to the N2H2 server for the HTTP traffic that is destined for one of the domains in the exclusive list. Thus, the N2H2 server does not have to deal with look-up requests for HTTP traffic that is destined for a host that has already been marked as “allowed.”

Flexibility when entering domain names is also provided; that is, the user can enter the complete domain name or a partial domain name. If the user adds a complete domain name such as “www.cisco.com” to the exclusive domain list, all HTTP traffic whose URLs are destined for this domain (such as www.cisco.com/news and www.cisco.com/index) will be excluded from the N2H2 URL filtering policies and, on the basis of the configuration, the URLs will be permitted or blocked (denied).

If the user adds only a partial domain name to the exclusive domain list, such as “.cisco.com,” all URLs whose domain names end with this partial domain name (such as www.cisco.com/products and www.cisco.com/eng) will be excluded from the N2H2 URL filtering policies and, based upon the configuration, the URLs will be permitted or blocked (denied).

To configure an exclusive domain list, use the **ip urlfilter exclusive-domain** command.

Allow Mode

The system will go into allow mode when connections to all the N2H2 servers are down. The system will return to normal mode when a connection to at least one web N2H2 server is up. Allow mode directs your system to forward or drop all packets on the basis of the configurable allow mode setting. By default, allow mode is off, so all HTTP requests are forbidden if all N2H2 servers are down.

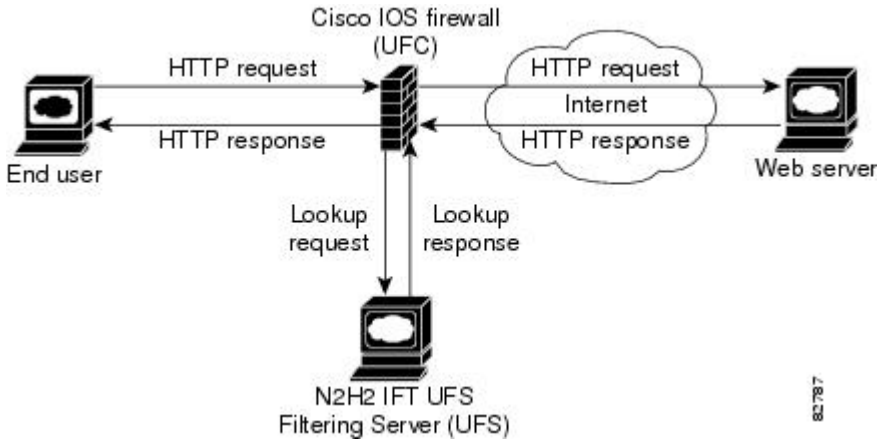
To configure allow mode for your system, use the **ip urlfilter allowmode** command.

Feature Design of Firewall N2H2 Support

Note This feature assumes that the N2H2 server will be part of a protected network and that requests from the Cisco IOS firewall will not travel over any unprotected network to reach the N2H2 server.

The figure below and the corresponding steps explain a sample URL filtering network topology.

Figure 2 Cisco IOS Firewall N2H2 URL Filtering Sample Topology



1. The end user browses a page on the web server, and the browser sends an HTTP request.
2. After the Cisco IOS firewall receives this request, it forwards the request to the web server, while simultaneously extracting the URL and sending a look-up request to the N2H2 server.
3. After the N2H2 server receives the look-up request, it checks its database to see whether it should permit or deny the URL; it returns a permit or deny status via a look-up response to the Cisco IOS firewall.
4. After the Cisco IOS Firewall receives this look-up response, it performs one of the following functions:
5. If the look-up response permits the URL, it sends the HTTP response to the end user.
6. If the look-up response denies the URL, the N2H2 server redirects the user to its own internal web server, which displays a message that describes the category under which the URL is blocked; thereafter, the connection is reset to both ends.

Supported N2H2 Filtering Methods

The Cisco IOS firewall supports most of the filtering methods that are supported by the N2H2 server. The table below lists N2H2 filtering methods and identifies which methods are supported by Cisco.

Table 5 N2H2 Filtering Methods Supported on Cisco IOS Firewall

N2H2 Filtering Method	Description	Supported by Cisco IOS Firewall?
Client-IP-based filtering	Filtering is applied to specified client IP addresses	Yes

N2H2 Filtering Method	Description	Supported by Cisco IOS Firewall?
Global filtering	Filtering is applied to all users, groups, and IP addresses	Yes
User-based filtering	Filtering is applied to a specified user	No

How to Configure N2H2 URL Support

Configuring Cisco IOS Firewall N2H2 URL Filtering

N2H2 is based on a pass-through filtering technology, which is the most accurate, reliable, and scalable method of Internet filtering. Pass-through filtering requires all requests for web pages to pass through an Internet control point, such as a firewall, proxy server, or caching device. N2H2 is integrated with these control points and checks each request to determine whether it should be allowed or denied. All responses are logged for reporting purposes.

- Before enabling N2H2 URL filtering, you should always ensure that there is not another URL filtering scheme configured, such as Websense. If you try to enter a new filtering scheme when one already exists, the new scheme will be ignored, and the system will display an error message that says, “different URL filtering scheme cannot co-exist.”
- URL filtering does not have an interface-specific command. It relies on Cisco IOS firewall C HTTP inspection to classify the traffic that needs filtering. This makes the configuration of Cisco IOS firewall inspection mandatory for the URL filtering feature to work. For more details on Cisco IOS firewall configuration, refer to the chapter “Cisco IOS Firewall Overview” in the IOS IOS Security Configuration Guide, Release 12.2.

Note Enabling HTTP inspection (via the **ip inspect name**command) triggers the Java applet scanner, which is very CPU intensive. The only way to stop the Java applet scanner is to specify the **java-list** *access-list* option and configure a standard access-list to allow any traffic. Configuring URL filtering without enabling the **java-list** *access-list*option will severely impact performance.

>

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **ip inspect name** *inspection-name* http [**urlfilter**] [**java-list** *access-list*] [**alert** {**on** | **off**}] [**timeout** *seconds*] [**audit-trail** {**on** | **off**}]
4. **ip urlfilter server vendor websense** | **n2h2**} *ip-address* [**port** *port-number*] [**timeout** *seconds*] [**retransmit** *number*]
5. **ip urlfilter alert**
6. **ip urlfilter audit-trail**
7. **ip urlfilter urlf-server-log**
8. **ip urlfilter exclusive-domain permit** | **deny**} *domain-name*
9. **ip urlfilter cache** *number*
10. **ip urlfilter allowmode** [**on** | **off**]
11. **ip urlfilter max-resp-pak** *number*
12. **ip urlfilter max-request** *number*
13. **interface** *type slot* / *port*
14. **ip inspect inspection-name** {**in** | **out**}

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Router> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Router# configure terminal`	Enters global configuration mode.
Step 3	**ip inspect name** *inspection-name* http [**urlfilter**] [**java-list** *access-list*] [**alert** {**on** | **off**}] [**timeout** *seconds*] [**audit-trail** {**on** | **off**}] **Example:** `Router(config)# ip inspect name fw_urlf http urlfilter java-list 51 timeout 30`	Turns on HTTP inspection. The **urlfilter** keyword associates URL filtering with HTTP inspection. **Note** You may configure two or more inspections in a router, but URL filtering will work only with the inspections in which the **urlfilter** keyword is enabled. **Note** Enabling HTTP inspection with or without any options triggers the Java applet scanner, which is CPU intensive. The only way to stop the Java applet scanner is to specify the **java-list** *access-list* option. Configuring URL filtering without enabling the **java-list** *access-list*option will severely impact performance.

	Command or Action	Purpose
Step 4	**ip urlfilter server vendor websense** | **n2h2**} *ip-address* [**port** *port-number*] [**timeout** *seconds*] [**retransmit** *number*] **Example:** `Router(config)# ip urlfilter server vendor websense 10.201.6.202`	Configures an N2H2 server to interact with the firewall to filter HTTP requests based on a specified policy. • *ip-address* --IP address of the vendor server. • **port** *port-number* --Port number that the vendor server listens on. The default port number is 4005. • **timeout** *seconds* --Length of time that the firewall will wait for a response from the vendor server. The default timeout is 5 minutes. • **retransmit** *number* --Number of times the firewall will retransmit the request when a response does not arrive. The default value is 2 times.
Step 5	**ip urlfilter alert** **Example:** `Router(config)# ip urlfilter alert`	(Optional) Enables the system alert, which displays system messages such as a server entering allow mode or going down. • The system alert is enabled by default.
Step 6	**ip urlfilter audit-trail** **Example:** `Router(config)# ip urlfilter audit-trail`	(Optional) Enables the logging of messages into the syslog server of router. • This function is disabled by default.
Step 7	**ip urlfilter urlf-server-log** **Example:** `Router(config)# ip urlfilter urlf-server-log`	(Optional) Enables the logging of system messages on the URL filtering server (the N2H2 server). This function is disabled by default.
Step 8	**ip urlfilter exclusive-domain permit** | **deny**} *domain-name* **Example:** `Router(config)# ip urlfilter exclusive-domain permit www.cisco.com`	(Optional) Adds a domain name to or from the exclusive domain list so the firewall does not have to send look-up requests to the N2H2 server. • **permit** --Permits all traffic destined for the specified domain name. • **deny** --Denies all traffic destined for the specified domain name. • *domain-name* --Domain name that is added or removed from the exclusive domain list.
Step 9	**ip urlfilter cache** *number* **Example:** `Router(config)# ip urlfilter cache 4500`	(Optional) Configures cache table parameters. • *number* --Specifies the maximum number of destination IP addresses that can be cached into the cache table; the default is 5000.

Command or Action	Purpose
Step 10 **ip urlfilter allowmode** [**on** | **off**] **Example:** `Router(config)# ip urlfilter allowmode on`	(Optional) Turns on the default mode of the filtering systems. • **on** --Allows HTTP requests to pass to the end user if all N2H2 servers are down. • **off** --Blocks all HTTP requests if all N2H2 servers are down; **off** is the default setting.
Step 11 **ip urlfilter max-resp-pak** *number* **Example:** `Router(config)# ip urlfilter max-resp-pak 150`	(Optional) Configures the maximum number of HTTP responses that the firewall can keep in its packet buffer. • The default value is 200. The maximum value is 20000, so you may set the **max-resp-pak** *number* to a value up to 20000.
Step 12 **ip urlfilter max-request** *number* **Example:** `Router(config)# ip urlfilter max-request 500`	(Optional) Sets the maximum number of outstanding requests that can exist at any given time. • The default value is 1000.
Step 13 **interface** *type slot / port* **Example:** `Router(config)# interface FastEthernet 0/0`	Configures an interface type and enters interface configuration mode
Step 14 **ip inspect inspection-name** {**in** | **out**} **Example:** `Router(config-if)# ip inspect inspection-name out`	Applies a set of inspection rules to an interface. • URL filtering is associated with inspection, and inspection is an interface-specific command. Hence, the **ip inspect** command needs to be configured on an interface.

- Troubleshooting Tips, page 40

Troubleshooting Tips

This feature introduces the following alert messages:

- "%URLF-3-SERVER_DOWN: Connection to the URL filter server 10.92.0.9 is down"

This level three LOG_ERR-type message is displayed when a configured UFS goes down. When this happens, the firewall will mark the configured server as secondary, try to bring up one of the other secondary servers, and mark that server as the primary server. If there is no other server configured, the firewall will enter allow mode and display the "URLF-3-ALLOW_MODE" message.

- %URLF-3-ALLOW_MODE: Connection to all URL filter servers are down and ALLOW MODE is OFF

This LOG_ERR type message is displayed when all UFSs are down and the system enters allow mode.

Note Whenever the system goes into allow mode (all filter servers are down), a periodic keepalive timer will be triggered which will try to bring up a server by opening a TCP connection.

- "%URLF-5-SERVER_UP: Connection to an URL filter server 10.92.0.9 is made, the system is returning from ALLOW MODE"

This LOG_NOTICE-type message is displayed when the UFSs are detected as being up and the system is returning from allow-mode.

- "%URLF-4-URL_TOO_LONG:URL too long (more than 3072 bytes), possibly a fake packet?"

This LOG_WARNING-type message is displayed when the URL in a look-up request is too long; any URL longer than 3K will be dropped.

- "%URLF-4-MAX_REQ: The number of pending request exceeds the maximum limit <1000>"

This LOG_WARNING-type message is displayed when the number of pending requests in the system exceeds the maximum limit and all further requests are dropped.

To display these alert messages, use the **ip urlfilter alert** command.

This feature introduces the following syslog messages:

- "%URLF-6-SITE_ALLOWED: Client 10.0.0.2:12543 accessed server 10.76.82.21:8080"

This message is logged for each request whose destination IP address is found in the cache. It includes the source IP address, source port number, destination IP address, and destination port number. The URL is not logged because the IP address of the request is found in the cache, so parsing the request and extracting the URL is a waste of time.

- "%URLF-4-SITE-BLOCKED: Access denied for the site 'www.sports.com'; client 10.54.192.6:34557 server 172.24.50.12:80"

This message is logged when a request finds a match against one of the blocked domains in the exclusive-domain list or the corresponding entry in the IP cache.

- "%URLF-6-URL_ALLOWED: Access allowed for URL http://www.n2h2.com/; client 10.54.192.6:54123 server 192.168.0.1:80"

This message is logged for each URL request that is allowed by a UFS. It includes the allowed URL, source IP address, source port number, destination IP address, and destination port number. Longer URLs will be truncated to 300 bytes and logged.

- "%URLF-6-URL_BLOCKED: Access denied URL http://www.google.com; client 10.54.192.6:54678 server 172.19.14.2:80"

This message is logged for each URL request that is blocked by a UFS. It includes the blocked URL, source IP address, source port number, destination IP address, and destination port number. Longer URLs will be truncated to 300 bytes and then logged.

To display these syslog messages, use the **ip urlfilter audit-trail** command.

Verifying Firewall and N2H2 URL Filtering

To verify that the Firewall N2H2 Support feature is working, perform any of the following optional steps:

Command or Action	Purpose
enable `Router> enable`	Enables privileged EXEC mode. • Enter your password if prompted.

show ip urlfilter cache
 	
`Router#`
`show ip urlfilter cache`
	Displays the destination IP addresses that are cached into the cache table.

show ip urlfilter config Router# show ip urlfilter config	Displays the size of the cache, the maximum number of outstanding requests, the allow mode state, and the list of configured N2H2 servers.
show ip urlfilter statistics Router# show ip urlfilter statistics	Displays information such as the number of requests that are sent to the N2H2 server, the number of responses received from the N2H2 server, the number pending requests in the system, the number of failed requests, the number of blocked URLs.

Maintaining the Cache Table

To clear the cache table of a specified or all IP addresses, perform the following optional steps:

Command or Action	Purpose
enable `Router> enable`	Enables privileged EXEC mode. • Enter your password if prompted.

Command or Action	Purpose

clear ip urlfilter cache {*ip-address* | **all**}
 	
`Router#`
`clear ip urlfilter cache all`
	Clears the cache table.

Monitoring the URL Filter Subsytems

To monitor the URL filter subsystems, perform the following optional steps:

Command or Action	Purpose
enable `Router> enable`	Enables privileged EXEC mode. • Enter your password if prompted.

debug ip urlfilter **function-trace** **detailed** **events**
 	
`Router#`
`debug ip urlfilter detailed`
	Enables debugging information of URL filter subsystems.
	• **function-trace** --Prints a sequence of important functions that are called when configuring URL filtering.
	• **detailed** --Prints detailed information about various activities that occur during URL filtering.
	• **events** --Prints various events such as queue event, timer event, and socket event.

Configuration Examples for Firewall and Webserver

Example URL Filter Client Firewall Configuration

The following example shows how to configure the Cisco IOS firewall (also known as the URL filter client [UFC]) for N2H2 URL filtering:

```
Topology:
```

```
End User------LAN-----Fa0/0 -- Firewall -- S2/0----- Internet ---- Web Server
               |               Router
               |
N2H2           |
Server --------+

Router Configuration:
Example 1:
hostname fw9-7200b
!
!-----------------------------------------------------------------
! The following commands define the inspection rule "myfw," allowing
! the specified protocols to be inspected. Note that the "urlfilter"
! keyword entered for HTTP protocol enables URL filtering on HTTP
! traffic that are bound to this inspection.
!-----------------------------------------------------------------
!
ip inspect name myfw http urlfilter
ip inspect name myfw ftp
ip inspect name myfw smtp
ip inspect name myfw h323
!
!-----------------------------------------------------------------------
! The following command sets the URL filtering cache table size to 12000.
!-----------------------------------------------------------------------
ip urlfilter cache 12000
!
!-----------------------------------------------------------------
! The following commands configure three exclusive domains--
! two partial domains and one complete domain.
!-----------------------------------------------------------------
ip urlfilter exclusive-domain permit .weapons.com
ip urlfilter exclusive-domain deny .nbc.com
ip urlfilter exclusive-domain permit www.cisco.com
!
!---------------------------------------------------------------
! The following two commands enable URL filtering Audit Trail and
! Alert messages.
!---------------------------------------------------------------
ip urlfilter audit-trail
ip urlfilter alert
!
!-------------------------------------------------------------
! The command configures the N2H2 URL filtering server installed
! on 192.168.3.1.
!-------------------------------------------------------------
ip urlfilter server vendor n2h2 192.168.3.1
!
!-----------------------------------------------------------------
! Create Access Control List 102:
! ACL 102 denies all IP protocol traffic except for ICMP traffic.
! This means that only the return traffic for protocols defined in the
! inspection rule and the ICMP traffic is allowed access through the
! interface where this rule is applied.
!
! Note that ACL is given here for an example; it is not relevant
! to the URL filtering. The URL filtering will work without ACL also.
!-----------------------------------------------------------------
!
access-list 102 permit icmp any any
access-list 102 deny tcp any any
access-list 102 deny udp any any
access-list 102 deny ip any any
!
!
interface FastEthernet0/0
ip address 192.168.3.254 255.255.255.0
ip nat inside
no ip route-cache
no ip mroute-cache
!
interface Ethernet1/0
no ip route-cache
```

```
no ip mroute-cache
shutdown
duplex half
!
interface Ethernet1/1
no ip address
no ip mroute-cache
shutdown
duplex half
!
!--------------------------------------------------------------------------
! The ACL and CBAC inspection rules are applied to the Serial2/0 interface.
!  In this example, the ACL is applied IN, meaning that it applies to traffic
!  inbound from the internet. The CBAC inspection rule myfw is applied OUT,
!  meaning that CBAC inspects the traffic that goes out through the interface
!  and controls return traffic to the router for an existing connection.
!--------------------------------------------------------------------------
interface Serial2/0
ip address 10.6.9.7 255.255.0.0
ip access-group 102 in
ip nat outside
ip inspect myfw out
no ip directed-broadcast
no ip mroute-cache
!
ip nat inside source static 192.168.3.1 10.6.243.1
ip nat inside source static 192.168.3.2 10.6.243.2
ip nat inside source static 192.168.3.3 10.6.243.3
ip classless
ip route 192.168.0.30 255.255.255.255 10.6.0.1
!
!
line con 0
exec-timeout 0 0
stopbits 1
line aux 0
stopbits 1
line vty 0 4
password letmein
login
!
end
Example 2:
! In the above example, the CBAC can also be configured on the inbound
! FastEthernet0/0 interface as IN, in which case the CBAC inspects all
! the traffic that comes in on FastEthernet0/0 and controls return traffic
! that leaves out of this interface for an existing connection.

interface FastEthernet0/0
ip address 192.168.3.254 255.255.255.0
ip access-group 102 out
ip nat inside
ip inspect myfw in
no ip route-cache
no ip mroute-cache
!
!
hostname fw9-7200b
!
logging buffered 64000 debugging
enable secret 5 $1$qMOf$umPb75mb3sV27JpNbW//7.
!
clock timezone PST -8
clock summer-time PDT recurring
ip subnet-zero
ip cef
no ip domain lookup
!
ip inspect name test http urlfilter
ip urlfilter cache 5
ip urlfilter exclusive-domain permit .weapons.com
ip urlfilter exclusive-domain deny .nbc.com
ip urlfilter exclusive-domain permit www.cisco.com
```

```
ip urlfilter audit-trail
ip urlfilter alert
ip urlfilter server vendor n2h2 192.168.3.1
ip audit notify log
ip audit po max-events 100
ip port-map http port 8080
!
no voice hpi capture buffer
no voice hpi capture destination
!
mta receive maximum-recipients 0
!
interface FastEthernet0/0
 ip address 192.168.3.254 255.255.255.0
 ip access-group 101 out
 ip nat inside
 ip inspect test in
 no ip route-cache
 no ip mroute-cache
!
interface Ethernet1/0
 ip address 10.6.9.7 255.255.0.0
 ip nat outside
 no ip route-cache
 no ip mroute-cache
 duplex half
!
interface Ethernet1/1
 no ip address
 no ip mroute-cache
 shutdown
 duplex half
!
interface Ethernet1/2
 no ip address
 no ip mroute-cache
 shutdown
 duplex half
!
interface Ethernet1/3
 no ip address
 no ip mroute-cache
 shutdown
 duplex half
!
interface Serial2/0
 no ip address
 no ip mroute-cache
 shutdown
 dsu bandwidth 44210
 framing c-bit
 cablelength 10
 serial restart_delay 0
 fair-queue
!
ip nat pool devtest 10.6.243.21 10.6.243.220 netmask 255.255.0.0
ip nat inside source list 1 pool devtest
ip nat inside source static 192.168.3.1 10.6.243.1
ip nat inside source static 192.168.3.2 10.6.243.2
ip nat inside source static 192.168.3.3 10.6.243.3
ip classless
ip route 192.168.0.30 255.255.255.255 10.6.0.1
no ip http server
no ip http secure-server
!
ip pim bidir-enable
!
!
access-list 101 deny   tcp any any
access-list 101 deny   udp any any
access-list 101 permit ip any any
access-list 102 deny   tcp any any
access-list 102 deny   udp any any
```

```
access-list 102 permit ip any any
dialer-list 1 protocol ip permit
dialer-list 1 protocol ipx permit
!
!
call rsvp-sync
!
!
mgcp profile default
!
dial-peer cor custom
!
!
!gatekeeper
 shutdown
!
!
line con 0
 exec-timeout 0 0
 stopbits 1
line aux 0
 stopbits 1
line vty 0 4
 password letmein
 login
!
exception core-file sisu-devtest/coredump/fw9-7200b.core
exception dump 192.168.0.1
no scheduler max-task-time
!
end
```

Additional References

Related Documents

Related Topic	Document Title
Cisco IOS commands	Cisco IOS Master Commands List, All Releases
Websense URL filtering information	*Firewall Websense URL Filtering*

Standards

Standards	Title
None	--

MIBs

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

RFCs

RFCs[1]	Title
RFC 1945	*Hypertext Transfer Protocol -- HTTP/1.0*
RFC 2616	*Hypertext Transfer Protocol -- HTTP/1.1*

Technical Assistance

Description	Link
The Cisco Support and Documentation website provides online resources to download documentation, software, and tools. Use these resources to install and configure the software and to troubleshoot and resolve technical issues with Cisco products and technologies. Access to most tools on the Cisco Support and Documentation website requires a Cisco.com user ID and password.	http://www.cisco.com/cisco/web/support/index.html

Feature Information for Firewall N2H2 Support

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

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

1 Not all supported RFCs are listed.

Table 6 Feature Information for Firewall N2H2 Support

Feature Name	Releases	Feature Information
Firewall N2H2 Support	12.2(11)YU 12.2(15)T	The Firewall N2H2 Support feature provides users with an additional option when choosing the URL filter vendor. Just like the Websense URL filtering server, N2H2 interacts with your Cisco IOS firewall (also known as Cisco Secure Integrated Software [CSIS]) to allow you to prevent users from accessing specified websites on the basis of some policy. The Cisco IOS firewall works with the N2H2 Internet Filtering Protocol (IFP) server to know whether a particular URL should be allowed or denied (blocked). The following commands were introduced or modified: **clear ip urlfilter cache**, **debug ip urlfilter**, **ip inspect name**, **ip urlfilter alert**, **ip urlfilter allowmode**, **ip urlfilter audit-trail**, **ip urlfilter cache**, **ip urlfilter exclusive-domain**, **ip urlfilter max-request**, **ip urlfilter max-resp-pak**, **ip urlfilter server vendor**, **ip urlfilter urlf-server-log**, **show ip urlfilter cache**, **show ip urlfilter config**, **show ip urlfilter statistics**.

Glossary

ACL--Access Control List.

CSIS--Cisco Secure Integrated Software. CSIS is a content-based firewall that currently inspects application data, checks for protocol conformance, extracts the relevant port information to create dynamic access list entries that successfully allows return traffic, and closes the ports at the end of the session.

ICMP --Internet Control Message Protocol. ICMP is a network layer Internet protocol that reports errors and provides other information relevant to IP packet processing. ICMP is documented in RFC 792.

UFC--URL filter client. UFC is a separate process that accepts URLs from CSIS, forwards the URL to the Websense server, and process the replies from the vendor server (Websense or N2H2).

UFS--URL filter server. UFS is a generic name given to the vendor server (Websense or N2H2), which processes URLs and decides whether to allow or deny web traffic based on a given policy.

Note Refer to the *Internetworking Terms and Acronyms* for terms not included in this glossary.

Firewall Support for SIP

The Firewall Support for SIP feature integrates Cisco IOS firewalls, Voice over IP (VoIP) protocol, and Session Initiation Protocol (SIP) within a Cisco IOS-based platform, enabling better network convergence.

Note Some Cisco IOS versions earlier than 12.2(11)YU and 12.2(15)T may accept the configuration commands for SIP that are shown in this document; however, those earlier versions will not function properly.

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest caveats and feature information, see Bug Search Tool and the release notes for your platform and software release. To find information about the features documented in this module, and to see a list of the releases in which each feature is supported, see the feature information table at the end of this module.

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

Restrictions for Firewall Support for SIP

DNS Name Resolution

Although SIP methods can have Domain Name System (DNS) names instead of raw IP addresses, this feature currently does not support DNS names.

SIP UDP Support Only

This feature supports only the SIP User Datagram Protocol (UDP) format for signaling; the TCP format is not supported.

SIP Abbreviated Header

This feature does not support the compact form of SIP header fields.

Earlier Versions of Cisco IOS

Some Cisco IOS versions earlier than 12.2(11)YU and 12.2(15)T may accept the configuration commands for SIP that are shown in this document; however, those earlier versions will not function properly.

Information About Firewall Support for SIP

Cisco IOS Firewall

The Cisco IOS firewall extends the concept of static access control lists (ACLs) by introducing dynamic ACL entries that open on the basis of the necessary application ports on a specific application and close these ports at the end of the application session. The Cisco IOS firewall achieves this functionality by inspecting the application data, checking for conformance of the application protocol, extracting the relevant port information to create the dynamic ACL entries, and closing these ports at the end of the session. The Cisco IOS firewall is designed to easily allow a new application inspection whenever support is needed.

SIP - Session Initiation Protocol

SIP is an ASCII-based, application-layer control protocol that can be used to establish, maintain, and terminate calls between two or more endpoints. Like other VoIP protocols, SIP is designed to address the functions of signaling and session management within a packet telephony network. Signaling allows call information to be carried across network boundaries. Session management provides the ability to control the attributes of an end-to-end call.

SIP Messages

SIP has two types of messages--requests and responses--that have the following generic structure:

generic-message = Request-Line | Status-Line

* (general-header | request-header

| response-header | entity-header)

CRLF

[message-body]

Note Any of these message components may contain embedded IP addresses.

The table below identifies the six available SIP request messages.

Table 7 SIP Request Messages

SIP Message	Purpose
ACK	Confirms receipt of a final response to INVITE
BYE	Is sent by either side to end the call
CANCEL	Is sent to end a call that has not yet been connected
INVITE	Is a request from a User Agent Client (UAC) to initiate a session
OPTIONS	Are sent to query capabilities of the user agents and network servers
REGISTER	Is sent by the client to register the address with a SIP proxy

The table below identifies the available SIP response methods.

Table 8 SIP Response Messages

SIP Message	Purpose
1xx Informational	• 100 = Trying • 180 = Ringing • 181 = Call Is Being Forwarded • 182 = Queued • 183 = Session Progress
2xx Successful	• 200 = OK
3xx Redirection	• 300 = Multiple Choices • 301 = Moved Permanently • 302 = Moved Temporarily • 303 = See Other • 305 = Use Proxy • 380 = Alternative Service

SIP Message	Purpose
4xx Request Failure	• 400 = Bad Request • 401 = Unauthorized • 402 = Payment Required • 403 = Forbidden • 404 = Not Found • 405 = Method Not Allowed • 406 = Not Acceptable • 407 = Proxy Authentication Required • 408 = Request Timeout • 409 = Conflict • 410 = Gone • 411 = Length Required • 413 = Request Entity Too Large • 414 = Request URI Too Large • 415 = Unsupported Media Type • 420 = Bad Extension • 480 = Temporarily Not Available • 481 = Call Leg/Transaction Does Not Exist
4xx Request Failure (continued)	• 482 = Loop Detected • 483 = Too Many Hops • 484 = Address Incomplete • 485 = Ambiguous • 486 - Busy Here
5xx Server Failure	• 500 = Internal Server Error • 501 = Not Implemented • 502 = Bad Gateway • 503 = Service Unavailable • 504 = Gateway Timeout • 505 = SIP Version Not Supported
6xx Global Failure	• 600 = Busy Anywhere • 603 = Decline • 604 = Does Not Exist Anywhere • 606 = Not Acceptable

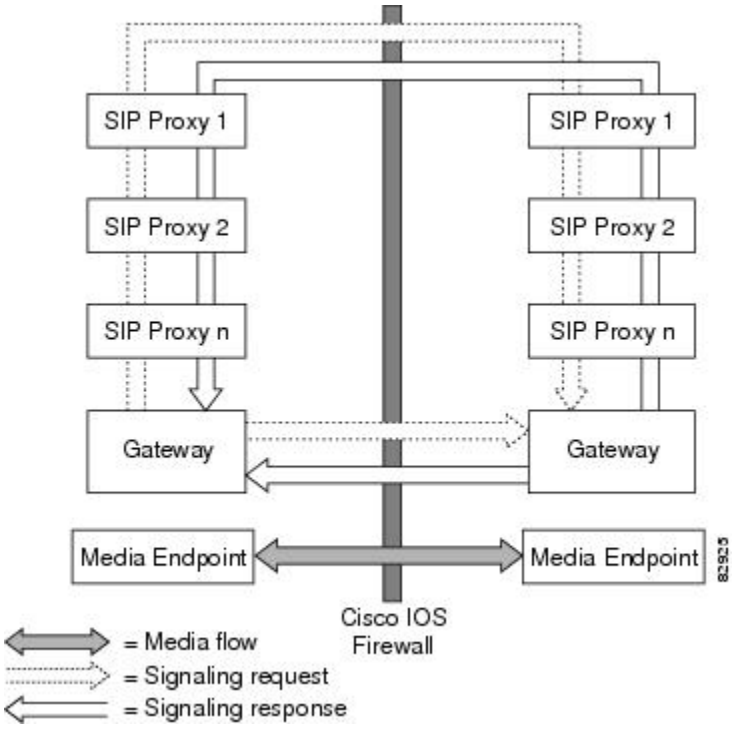
Firewall for SIP Functionality Description

The Firewall for SIP Support feature allows SIP signaling requests to traverse directly between gateways or through a series of proxies to the destination gateway or phone. After the initial request, if the Record-Route header field is not used, subsequent requests can traverse directly to the destination gateway address as specified in the Contact header field. Thus, the Cisco IOS firewall is aware of all surrounding proxies and gateways and allows the following functionality:

- SIP signaling responses can travel the same path as SIP signaling requests.
- Subsequent signaling requests can travel directly to the endpoint (destination gateway).
- Media endpoints can exchange data between each other.

See the figure below for a sample topology that displays these functionalities.

Figure 3 Cisco IOS Firewall for SIP Awareness Sample Topology



SIP Message Treatment by the Firewall

See the table below for information on the treatment of SIP methods by the Cisco IOS firewall.

Table 9 Treatment of SIP Methods by the Cisco IOS Firewall

SIP Message	Purpose
200 OK	Signifies the end of the call creation phase. The packet is checked for validity against the call database, and the contact information of the server is taken from it. Temporary call-flow-based openings in the firewall are created for allowing the BYE message, which can be initiated from the inside or outside.
200 OK for BYE	Signifies the graceful termination of the call and is in response to the BYE message. The same action as the CANCEL message is taken.

SIP Message	Purpose
ACK	Signifies that the message is passed after checking for validity.
BYE	Signifies the intent to terminate the call. The database state is updated and temporary openings in the firewall are created for response to the BYE message.
CANCEL	Signifies abnormal data termination. The signaling sessions, media sessions, pregenerated temporary openings in the firewall, and the call database entry for the call are removed.
INVITE	Occurs typically at the start of the call. The firewall will create a database entry upon receipt of this method and fill the database with relevant information extracted from this message. Temporary openings in the firewall will allow for a series of responses to the INVITE request. The temporary openings will be call-flow sensitive and will allow for responses for a fixed amount of time (t = 30 secs).
NO MATCH	Signifies a signaling message that is not present in the database.
Other Methods	Signifies that the message is passed if the call ID is present in the call database.
REGISTER	Results in the creation of an entry in the call database. Time-based, flow-control ACL firewall openings will allow for the response to the REGISTER and subsequent INVITE messages.
SESSION PROGRESS	Contains a response to the INVITE message, and it is a packet during the call creation phase. The packet is checked against the call database for validity of call ID and the media ports; the server proxy information is gathered from the packet. Media channels should be created in this phase.

Call Database

A call database, which contains the details of a call leg, is maintained for all call flows. A call database is created and maintained because there can be numerous signaling sessions for each call. The table below identifies the information available in the call database.

Table 10 Call Database Information

Type	Purpose
call_int_over	Checks to see whether or not call initialization is over, and if so, checks to see of the call is in the teardown phase
C con ip & C con port	Signifies the IP address and port in the contact field of the initiator; for example, “Contact:<sip: 1111@172.16.0.3:5060;user=phone>”
C media ip & C media port	Signifies the IP address in the media field of the initiator; for example, “c=IN IP4 172.16.0.3”
C media port	Signifies the port in the media field of the initiator; for example, “m=audio 20758 RTP/AVP 0”
C src ip & C src port	Signifies the actual IP address and port of the initiator
C via ip & C via port	Signifies the IP address and port in the via field of the initiator (the first via line); for example, “Via: SIP/2.0/UDP 172.16.0.3:5060”
current sip state	Is the current state of the call (which helps to avoid retransmission)
from/to/callid	Is extracted from the “INVITE” SIP request message to identify the call
media header	Keeps the list of media sessions for the call
media opened	Signifies multiple messages that may have media information, so you need to check to see whether or not the media has been opened for the call
prev sip state	Signifies the previous state of the call (which helps to avoid retransmission)
S con ip & S con port	Signifies the IP address and port in the contact field for the responder
S media ip	Signifies the IP address in the media field for the responder
S media port	Signifies the port in the media field for the responder
S src ip & S src port	Signifies the actual IP address and port of the responder
S via ip & S via port	Signifies the IP address and port in the via field for the responder

Type	Purpose
signal header	Keeps the list of signaling sessions for the call
sip_proxy_traversed	Makes the firewall topologically aware of whether the call has traversed through proxies

How to Configure Your Firewall for SIP

Configuring Firewall for SIP Support

Before you configure Cisco IOS firewall support for Session Initiation Protocol (SIP) on your device, you first need to configure access lists. The purpose of access lists is to block SIP traffic from unprotected networks for which the firewall will create temporary openings for specific traffic.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **ip inspect name** *inspection-name* **sip** [**alert** {**on** | **off**}] [**audit-trail on** |**off**}] [**timeout** *seconds*]
4. **interface** *type number*
5. **ip inspect** *inspection-name* {**in** | **out**}
6. If SIP calls are coming from other interfaces, repeat Steps 3 through 5 and apply SIP inspections for the calls that are coming from those interfaces.

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Device> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.

	Command or Action	Purpose
Step 3	**ip inspect name** *inspection-name* **sip** [**alert** {**on** | **off**}] [**audit-trail on** |**off**}] [**timeout** *seconds*] **Example:** `Device(config)# ip inspect name voip sip`	Enables inspection for SIP.
Step 4	**interface** *type number* **Example:** `Device(config)# interface FastEthernet 0/0`	Configures an interface type and enters interface configuration mode.
Step 5	**ip inspect** *inspection-name* {**in** | **out**} **Example:** `Device(config-if)# ip inspect voip in`	Applies inspection configurations to an interface and for a particular traffic direction.
Step 6	If SIP calls are coming from other interfaces, repeat Steps 3 through 5 and apply SIP inspections for the calls that are coming from those interfaces.	**Note** The inspection of protocols other than SIP may not be desirable for traffic that comes from external networks, so it may be necessary to configure an additional inspection rule specifying only SIP.

Verifying Firewall for SIP Support

To verify Cisco IOS firewall session information, perform the following optional steps:

SUMMARY STEPS

1. **enable**
2. show ip inspect name *inspection-name*
3. **show ip inspect session detail**
4. show ip access-list

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Router> enable`	Enables privileged EXEC mode. • Enter your password if prompted.

	Command or Action	Purpose
Step 2	show ip inspect name *inspection-name* **Example:** `Router# show ip inspect name voip`	(Optional) Displays the configured inspection rule.
Step 3	**show ip inspect session detail** **Example:** `Router# show ip inspect session`	(Optional) Displays existing sessions that are currently being tracked and inspected by the Cisco IOS firewall. • The optional **detail** keyword causes additional details about these sessions to be shown.
Step 4	show ip access-list **Example:** `Router# show ip access-list`	(Optional) Displays the contents of all current IP access lists.

Monitoring Firewall for SIP Support

To monitor firewall events, perform the following optional steps:

Note Effective with Cisco IOS Release 12.4(20)T, the **debug ip inspect** command is replaced by the **debug policy-firewall** command. See the *Cisco IOS Debug Command Reference* for more information.

SUMMARY STEPS

1. **enable**
2. **debug ip inspect sip**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Router> enable`	Enables privileged EXEC mode. • Enter your password if prompted.

	Command or Action	Purpose
Step 2	**debug ip inspect sip** **Example:** `Router# debug ip inspect sip`	(Optional) Displays the operations of the SIP inspection engine for debugging purposes.

Configuration Examples for Firewall SIP Support

Example Firewall and SIP Configuration

The following example shows how to allow outside initiated calls and internal calls. For outside initiated calls, an ACL needs to be punched to allow for the traffic from the initial signaling packet from outside. Subsequent signaling and media channels will be allowed by the inspection module.

```
ip inspect name voip sip
interface FastEthernet0/0
 ip inspect voip in
!
!
interface FastEthernet0/1
 ip inspect voip in
 ip access-group 100 in
!
!
access-list 100 permit udp host <gw ip> any eq 5060
access-list 100 permit udp host <proxy ip> any eq 5060
access-list deny ip any any
```

Additional References

Related Documents

Related Topic	Document Title
Cisco IOS firewall information and configuration tasks	“Configuring Context-Based Access Control”
Cisco IOS firewall commands	*Cisco IOS Security Command Reference*

Standards

Standards	Title
None	--

MIBs

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

RFCs

RFCs[2]	Title
RFC 2543	SIP: Session Initiation Protocol

Technical Assistance

Description	Link
The Cisco Support and Documentation website provides online resources to download documentation, software, and tools. Use these resources to install and configure the software and to troubleshoot and resolve technical issues with Cisco products and technologies. Access to most tools on the Cisco Support and Documentation website requires a Cisco.com user ID and password.	http://www.cisco.com/cisco/web/support/index.html

Feature Information for Firewall SIP Support

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

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

2 Not all supported RFCs are listed.

Table 11 Feature Information for Firewall SIP Support

Feature Name	Releases	Feature Information
Firewall SIP Support	12.2(11)YU 12.2(15)T	The Firewall Support for SIP feature integrates Cisco IOS firewalls, Voice over IP (VoIP) protocol, and Session Initiation Protocol (SIP) within a Cisco IOS-based platform, enabling better network convergence. The following commands were introduced or modified: **debug ip inspect**, **ip inspect name**.

Firewall Support of Skinny Client Control Protocol

The Firewall Support of Skinny Client Control Protocol (SCCP) feature enables Context-Based Access Control (CBAC) inspection to support the Voice over IP (VoIP) protocol, Skinny Client Control Protocol (SCCP). That is, CBAC inspects Skinny control packets that are exchanged between a Skinny client and the Call Manager (CM); CBAC then configures the router (also known as the Cisco IOS Firewall) to enable the Skinny data channels to traverse through the router. In addition,the Firewall Support of Skinny Client Control Protocol (SCCP) feature extends the support of SCCP to accomodate video channels.

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest caveats and feature information, see Bug Search Tool and the release notes for your platform and software release. To find information about the features documented in this module, and to see a list of the releases in which each feature is supported, see the feature information table at the end of this module.

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

Prerequisites for Firewall Support of Skinny Client Control Protocol

The Skinny inspection module is part of the inspection subsystem; thus, your router must be running an image that has firewall support.

Restrictions for Firewall Support of Skinny Client Control Protocol

This feature has the following restrictions:

- Skinny inspection will inspect only the SCCP sessions that have been established after the firewall is configured with Skinny inspection. That is, any SCCP sessions that were established through the firewall before the Skinny inspection was configured will not be inspected.
- This feature does not support Music on Hold (MOH) when a device other than the CM is the music server. (This feature does support MOH when the CM is the music server.)
- This feature does not address either the multicast functionality of SCCP or the functionality of multiple active calls on a single Skinny client.

This feature does not support the following Skinny and firewall configurations:

- The firewall and CM cannot be in the same router. Skinny inspection does not support this configuration because the current firewall implementation does not inspect sessions that start or terminate at the router. Thus, Skinny inspection will work only with an external CM.
- The CM and the Skinny client cannot be on three different networks that are separated at the firewall. The current firewall implementation does not inspect sessions that have devices residing on more than two distinct networks that are segregated at the firewall. That is, if there are more than two interfaces at the firewall, session inspection is not supported.

Information About Firewall Support of Skinny Client Control Protocol

Context-Based Access Control Overview

CBAC extends the concept of static access control lists (ACLs) by introducing dynamic ACL entries that open the necessary application ports on the basis of a specific application and close these ports at the end of the application session. CBAC achieves this functionality by inspecting the application data, checking for conformance of the application protocol, extracting the relevant port information to create the dynamic ACL entries, and closing these ports at the end of the session. CBAC is designed to easily allow a new application inspection whenever support is needed.

Skinny Overview

Skinny enables voice communication between two Skinny clients through the use of a CM. Typically, the CM provides service to the Skinny clients on TCP Port 2000. Initially, a Skinny client connects to the CM by establishing a TCP connection; the client will also establish a TCP connection with a secondary CM, if

available. After the TCP connection is established, the client will register with the primary CM, which will be used as the controlling CM until it reboots or there is a keepalive failure. Thus, the Skinny TCP connection between the client and the CM exists forever and is used to establish calls coming to or from the client. If a TCP connection failure is detected, the secondary CM is used. All data channels established with the previous CM remain active and will be closed after the end parties hang up the call.

The table below lists the set of messages that are necessary for the data sessions to open and close. Skinny inspection will examine the data sessions that are deemed for opening and closing the access list pin holes.

Table 12 Skinny Data Session Messages

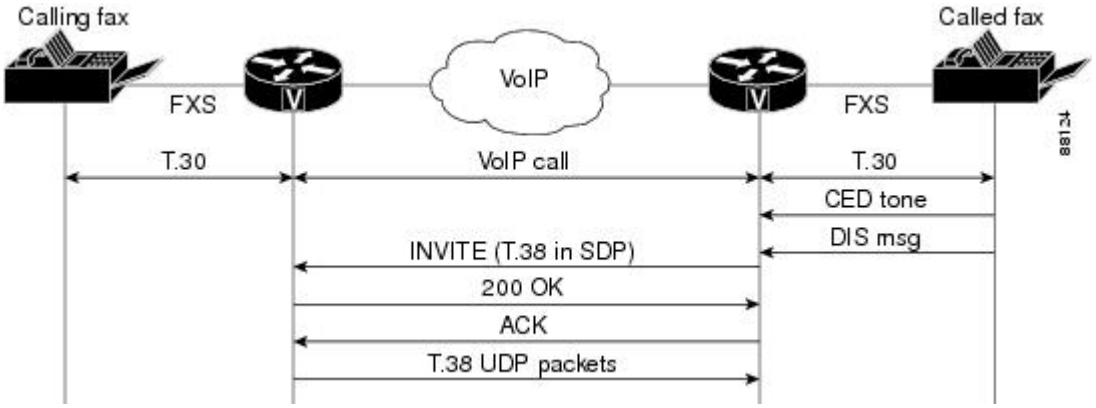
Skinny Inspection Message	Description
StationOpenReceiveChannelAckMessage	Contains the IP address and port information of the Skinny client sending this message. This message also contains the status of whether or not the client is willing to receive the voice traffic.
StationStartMediaTransmissionMessage	Contains the IP address and port information of the remote Skinny client.
StationCloseReceiveChannelMessage	CM instructs the Skinny client (on the basis of the information in this message) to close the receiving channel.
StationStopMediaTransmissionMessage	CM instructs the Skinny client (on the basis of the information in this message) to stop transmitting voice traffic.
StationStopSessionTransmissionMessage	CM instructs the Skinny client (on the basis of the information in this message) to end an indicated session.
StationOpenMultiMediaReceiveChannelAckMessage	Contains the IP address and port information of the skinny Client sending this message. It also contains the status of whether client is willing to receive the video and data channels.
StationCloseMultiMediaReceiveChannel	This message is sent from the Cisco Unified Communications Manager to the Skinny endpoint to request closing the receiving video or data channel.
StationStartMultiMediaTransmitMessage	This message is sent from Cisco Unified Communications Manager to the Skinny endpoint whenever Cisco Unified Communications Manager gets back OpenLogicalChannelAck for video or data channel.
StationStopMultiMediaTransmission	This message is sent to Skinny endpoints to request transmission of video\data channel to stop.

CBAC and Skinny Functionality Overview

The figure below depicts typical deployment solutions that are supported by CBAC inspection for Skinny. According to Figure 1, a firewall with Skinny inspection can be configured on Cisco IOS Router A, Cisco IOS Router B, or both routers, thereby addressing the following three scenarios:

- A Cisco IOS router with a firewall on the customer premises equipment (CPE) side, supporting Skinny VoIP phone
- A Cisco IOS router with a firewall on the CM side
- A Cisco IOS router with a firewall at both ends of the connection

Figure 4 CBAC Inspection for Skinny Sample Topology



SCCP Video Call Flow

The figure below illustrates the communication paths between the clients and the Call Manager (CM). The firewall resides either a) in the path from Client A to CM and from Client A to Client B as indicated by Firewall-1 or b) in the path from Client B to CM and from Client B to Client A as indicated by Firewall-2.

Figure 5 Skinny Client to Skinny Client Communication

Setting Skinny CBAC Session Timeouts

Session timeouts are triggered when traffic is not seen on a particular session for a configured amount of time. (This value is configured via the **ip inspect name** command.) After the inactivity timeout is triggered, the firewall will clean up the session and deallocate all of the session data.

You must set the inactivity timeout value for Skinny to a greater value than the keepalive timeout value that is configured between the CM and Skinny clients. Otherwise, the Skinny connection may become inaccessible for inspection because the firewall might delete the session-related information due to inactivity.

After the inactivity timeout is triggered, the inspection module will send reset (RST packets) to both ends of the connection. Any data channels that are associated with the control channel will not be closed. After both end parties hang up, there will not be any traffic on the data channels and the connection will eventually timeout.

Note If the inactivity timeout of the control channel that is connected to the primary CM is less than the keepalive timeout that is sent by the CM to the Skinny client, the firewall will set the inactivity timeout to three times the keepalive timeout. If a timeout is not configured, the default value of 3600 seconds will be used.

How to Configure Your Firewall for Skinny Support

Configuring Basic Skinny CBAC Inspection

Perform the following required steps to configure a basic Skinny CBAC configuration:

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **ip inspect name** *inspection-name protocol* **alert on off** }] [**audit-trail on**| **off**}] [**timeout** *seconds*
4. ip inspect name *inspection-name protocol* **alert on**| **off**}] [**audit-trail on**| **off**}] [**timeout** *seconds*
5. **interface** *type number*
6. **ip access-group** {*access-list-number*} {**in** | **out**}
7. **ip inspect** *inspection-name* **in** | **out**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Router> enable`	Enables higher privilege levels, such as privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Router# configure terminal`	Enters global configuration mode.

	Command or Action	Purpose
Step 3	**ip inspect name** *inspection-name protocol* **alert on off** }] [**audit-trail on**| **off**}] [**timeout** *seconds* **Example:** `Router(config)# ip inspect name firewall skinny`	Enables CBAC Skinny inspections.
Step 4	ip inspect name *inspection-name protocol* **alert on**| **off**}] [**audit-trail on**| **off**}] [**timeout** *seconds* **Example:** `Router(config)# ip inspect name firewall tftp`	(Optional. Required if the TFTP server is outside the firewall.) Defines a set of inspection rules.
Step 5	**interface** *type number* **Example:** `Router(config)# interface FastEthernet 0/0`	Configures an interface type and enters interface configuration mode.
Step 6	**ip access-group** {*access-list-number*} {**in** | **out**} **Example:** `Router(config-if)# ip access-group 100 in`	Control access to an interface. Number of the access list that is blocking incoming traffic.
Step 7	**ip inspect** *inspection-name* **in** | **out** **Example:** `Router(config-if)# ip inspect firewall out`	Applies a set of inspection rules to an interface.

Configuring Port to Application Mapping

By default, the Skinny inspection will inspect SCCP messages to or from the CM on TCP port 2000. If you prefer to configure the CM to use a different port, the port to application mapping (PAM) feature should be used to specify the desired port to the Cisco IOS firewall. Thus, the firewall will inspect the SCCP messages in the desired port and in port 2000. To configure the CM to use a different port via PAM, use the ip port-map command.

Before you can configure PAM, you must first configure the steps in the section, “Configuring Basic Skinny CBAC Inspection.”

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **ip port map** *appl_name* **port** *port_num* [**list** *acl_num*]

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Router> enable`	Enables higher privilege levels, such as privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Router# configure terminal`	Enters global configuration mode.
Step 3	**ip port map** *appl_name* **port** *port_num* [**list** *acl_num*] **Example:** `Router(config)# ip port map skinny port 2100`	(Optional) Creates a port to address mapping for SCCP. This command allows you to indicate additional ports that need to be monitored for SCCP.

Verifying Cisco IOS Firewall for Skinny Support

To display active Skinny session information, perform the following optional steps:

SUMMARY STEPS

1. **enable**
2. **show ip inspect** {**name** *inspection-name* | **config** | **interfaces** | **session** [**detail**] | **all**}
3. **show ip access-list**
4. show ip port-map [*appl_name* | **port** *port_num*

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Router> enable`	Enables higher privilege levels, such as privileged EXEC mode. • Enter your password if prompted.

	Command or Action	Purpose
Step 2	**show ip inspect** {**name** *inspection-name* | **config** | **interfaces** | **session** [**detail**] | **all**} **Example:** `Router# show ip inspect session detail`	(Optional) Displays existing sessions that are currently being tracked and inspected by CBAC. The optional **detail** keyword causes additional details about these sessions to be shown.
Step 3	**show ip access-list** **Example:** `Router# show ip access-list`	(Optional) Displays the contents of all current IP access lists, which includes the dynamic access lists created by Skinny inspection.

Step 4	show ip port-map [*appl_name* | **port** *port_num*
	
	Example:
	`Router#`
	`show ip port-map skinny`
		(Optional) Displays information about the active port to application mappings on the router.
		Use this command to view Skinny port map information.
		• *appl_name* --Displays Skinny-specific PAM information. (You must specify the *skinny* argument.)

Monitoring Cisco IOS Firewall for Skinny Support

Note Effective with Cisco IOS Release 12.4(20)T, the **debug ip inspect** command is replaced by the **debug policy-firewall** command. See the *Cisco IOS Debug Command Reference* for more information.

To monitor debugging messages related to Skinny inspection, perform the following optional steps:

SUMMARY STEPS

1. **enable**
2. **debug ip inspect** {*sccp* | **detailed**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Router> enable`	Enables higher privilege levels, such as privileged EXEC mode. • Enter your password if prompted.

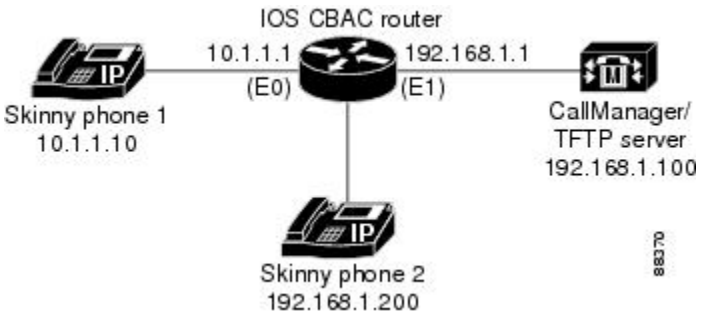
	Command or Action	Purpose
Step 2	**debug ip inspect** {*sccp* | **detailed**	(Optional) Displays and logs the debugging messages related to SCCP inspection.

	Example:
	`Router#`
	`debug ip inspect sccp`	

Configuration Examples for Firewall Skinny Support

Example Firewall and Skinny Configuration

Figure 6 Skinny and CBAC Configuration



The following is an example of how to configure a Cisco IOS firewall for Skinny support (see the figure above):

```
!  Define the name of the router as “CBAC-Firewall.”
!
host CBAC-Firewall
!
!  Create a DHCP server process to offer out 10.1.1.x addresses on the
!  inside network. Option 150 is used by Cisco IP phones as where to
!  look for their configuration file. A default router is required so that all
!  the IP phones can talk to networks other than just to the local 10.1.1.x.
!
ip dhcp pool localnetwork
   network 10.1.1.0 255.255.255.0
   option 150 ip 192.168.1.100
   default-router 10.1.1.1
!
!  Prevent the DHCP server process from assigning 10.1.1.1 -.9 as an IP
!  address on the local network. This is done to hold the addresses .2 - .9 as static-
!  defined addresses.
!
ip dhcp excluded-address 10.1.1.1 10.1.1.9
!
!  Define firewall rules to all Skinny traffic in/out along with TFTP
!  services.
```

```
!
ip inspect name fwout tftp
ip inspect name fwout skinny
!
!  Prevent any traffic from coming in.
!
access-list 100 deny ip any any
!
interface ethernet 1
  ip access-group 100 in
  ip inspect firewall out
If the CallManager is requiring Skinny registration to happen on port tcp/2100, you will
still need the above configuration plus the following additional step.
ip port map skinny port 2100
```

Additional References

Related Documents

Related Topic	Document Title
Additional CBAC information and configuration tasks	*Configuring Context-Based Access Control*
Security commands	• *Cisco IOS Security Command Reference: Commands A to C* • *Cisco IOS Security Command Reference: Commands D to L* • *Cisco IOS Security Command Reference: Commands M to R* • *Cisco IOS Security Command Reference: Commands S to Z*
PAM information and configuration tasks	*Configuring Port to Application Mapping*

Standards

Standards	Title
None	—

MIBs

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

RFCs

RFCs	Title
None	--

Technical Assistance

Description	Link
Technical Assistance Center (TAC) home page, containing 30,000 pages of searchable technical content, including links to products, technologies, solutions, technical tips, and tools. Registered Cisco.com users can log in from this page to access even more content.	http://www.cisco.com/public/support/tac/home.shtml

Firewall Support of Skinny Client Control Protocol

The following table lists the features in this module and provides links to specific configuration information.

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

Note The following table lists only the software release that introduced support for a given feature in a given software release train. Unless noted otherwise, subsequent releases of that software release train also support that feature.

Table 13 Feature Information for Zone-Based Policy Firewall

Feature Name	Releases	Feature Information
Firewall Support of Skinny Client Control Protocol	12.3(1)	The Firewall Support of Skinny Client Control Protocol (SCCP) feature enables Context-Based Access Control (CBAC) inspection to support the Voice over IP (VoIP) protocol, Skinny Client Control Protocol (SCCP).

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

Firewall Stateful Inspection of ICMP

The Firewall Stateful Inspection of ICMP feature addresses the limitation of qualifying Internet Control Management Protocol (ICMP) messages into either a malicious or benign category by allowing the Cisco IOS firewall to use stateful inspection to “trust” ICMP messages that are generated within a private network and to permit the associated ICMP replies. Thus, network administrators can debug network issues by using ICMP without concern that possible intruders may enter the network.

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest caveats and feature information, see Bug Search Tool and the release notes for your platform and software release. To find information about the features documented in this module, and to see a list of the releases in which each feature is supported, see the feature information table at the end of this module.

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

Restrictions for Firewall Stateful Inspection of ICMP

- To enable this feature, your Cisco IOS image must contain the Cisco IOS firewall.
- This feature does not work for the User Datagram Protocol (UDP) traceroute, in which UDP datagrams are sent instead of ICMP packets. The UDP traceroute is typically the default for UNIX systems. To use ICMP inspection with a UNIX host, use the “I” option with the traceroute command. This functionality will cause the UNIX host to generate ICMP traceroute packets, which will be inspected by the Cisco IOS firewall ICMP.

Information About Firewall Stateful Inspection of ICMP

Feature Design of Firewall Stateful Inspection of ICMP

ICMP is used to report errors and information about a network. It is a useful tool for network administrators who are trying to debug network connectivity issues. Unfortunately, intruders can also use ICMP to discover the topology of a private network. To guard against a potential intruder, ICMP messages can be blocked from entering a private network; however, a network administrator may then be unable to debug the network. Although a Cisco IOS router can be configured using access lists to selectively allow certain ICMP messages through the router, the network administrator must still guess which messages are potentially malicious and which messages are benign. With the introduction of this feature, a user can now configure a Cisco IOS firewall for stateful inspection to “trust” that the ICMP messages are generated within the private network and to permit the associated ICMP replies.

Note Access lists can still be used to allow unsolicited error messages along with Cisco IOS firewall inspection. Access lists complement Cisco IOS firewall ICMP inspection.

Stateful inspection of ICMP packets is limited to the most common types of ICMP messages that are useful to network administrators who are trying to debug their networks. That is, ICMP messages that do not provide a valuable tool for the internal network administrator will not be allowed. For the Cisco IOS firewall-supported ICMP message request types, see the table below.

Table 14 ICMP Packet Types Supported by CBAC

ICMP Packet Type	Name	Description
0	Echo Reply	Reply to Echo Request (Type 8)
3	Destination Unreachable	Possible reply to any request **Note** This packet is included because it is a possible response to any ICMP packet request.
8	Echo Request	Ping or traceroute request
11	Time Exceeded	Reply to any request if the time to live (TTL) packet is 0
13	Timestamp Request	Request
14	Timestamp Reply	Reply to Timestamp Request (type 13)

Note ICMP packet types 0 and 8 are used for pinging: the source sends out an Echo Request packet, and the destination responds with an Echo Reply packet. Packet types 0, 8, and 11 are used for ICMP traceroute: Echo Request packets are sent out starting with a TTL packet of 1, and the TTL is incremented for each hop. The intermediate hops respond to the Echo Request packet with a Time Exceeded packet; the final destination responds with an Echo Reply packet.

ICMP Inspection Checking

Return packets are checked by the inspect code, not by ACLs. The inspect code tracks each destination address from outgoing packets and checks each return packet. For ECHO REPLY and TIMESTAMP REPLY packets, the return address is checked. For UNREACHABLE and TIME EXCEEDED packets, the intended destination address is extracted from the packet data and checked.

For more information, see "Example Checking for ICMP Inspection".

How to Use Firewall Stateful Inspection of ICMP

Configuring Firewall Stateful Inspection for ICMP

To enable the Cisco IOS Firewall to start inspection ICMP messages, perform the following steps:

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **ip inspect name** *inspection-name* **icmp alert** {**on** | **off**}] [**audit-trail on** |**off**}] [**timeout** *seconds*

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Router> enable`	Enables privileged EXEC mode. • Enter your password if prompted.

	Command or Action	Purpose
Step 2	**configure terminal** **Example:** `Router# configure terminal`	Enters global configuration mode.

Step 3	**ip inspect name** *inspection-name* **icmp alert** {**on** | **off**}] [**audit-trail on** |**off**}] [**timeout** *seconds*
	
	Example:
	`Router(config)#`
	`ip inspect name test icmp alert on audit-trail on timeout 30`
		Turns on inspection for ICMP.
		• **alert** --Alert messages are generated. This function is **on** by default.
		• **audit-trail** --Audit trail messages are generated. This function is **off** by default.
		• **timeout** --Overrides the global channel inactivity timeout value. The default value of the *seconds* argument is 10.

Verifying Firewall and ICMP Session Information

To display active ICMP session and IP access list information, perform the following optional steps:

SUMMARY STEPS

1. **enable**
2. **show ip inspect session detail**
3. **show ip access-list**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Router> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**show ip inspect session detail** **Example:** `Router# show ip inspect session`	(Optional) Displays existing sessions that are currently being tracked and inspected by the Cisco IOS firewall. • The optional **detail** keyword causes additional details about these sessions to be shown.
Step 3	**show ip access-list** **Example:** `Router# show ip access-list`	(Optional) Displays the contents of all current IP access lists. For a sample output example, see the section “Example ICMP Session Verification.”

Monitoring Firewall and ICMP Session Information

To monitor debugging messages related to ICMP inspection, perform the following optional steps:

Note Effective with Cisco IOS Release 12.4(20)T, the **debug ip inspect** command is replaced by the **debug policy-firewall** command. See the Cisco IOS Debug Command Reference for more information.

SUMMARY STEPS

1. **enable**
2. **debug ip inspect icmp**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Router> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**debug ip inspect icmp** **Example:** `Router# debug ip inspect icmp`	(Optional) Displays the operations of the ICMP inspection engine for debugging purposes. For an example of sample output, see the command debug ip inspect in the Command Reference section.

Configuration Examples for Stateful Inspection of ICMP

Example Firewall Stateful Inspection for ICMP Configuration

The default ICMP timeout is deliberately short (10 seconds) due to the security hole that is opened by allowing ICMP packets with a wild-carded source address back into the inside network. The timeout will occur 10 seconds after the last outgoing packet from the originating host. For example, if you send a set of 10 ping packets spaced 1 second apart, the timeout will expire in 20 seconds or 10 seconds after the last outgoing packet. However, the timeout is not extended for return packets. If a return packet is not seen within the timeout window, the hole will be closed and the return packet will not be allowed in. Although the default timeout can be made longer if desired, it is recommended that this value be kept relatively short.

The following example shows how to configure a firewall for stateful inspection of ICMP packets:

```
no service pad
service timestamps debug uptime
service timestamps log uptime
service password-encryption
!
hostname UUT
!
ip subnet-zero
no ip domain lookup
!
ip inspect audit-trail
ip inspect name test icmp alert on audit-trail on timeout 30
!
interface Ethernet0
ip address 192.168.10.2 255.255.255.0
ip inspect test in
!
interface Ethernet1
ip address 192.168.20.2 255.255.255.0
ip access-group 101 in
!
ip classless
ip route 0.0.0.0 0.0.0.0 192.168.20.3
no ip http server
!
access-list 101 deny ip any any
!
line con 0
exec-timeout 0 0
!
end
```

Example Checking for ICMP Inspection

In the following example, three destinations were pinged. The example shows that the inspect code tracked each destination address in the inspect session information.

```
fw_1751#sh ip insp sess detail
Established Sessions
 Session 813A1808 (192.168.156.5:0)=>(0.0.0.0:0) icmp SIS_OPEN
   Created 00:04:20, Last heard 00:00:00
   Destinations: 3
       Dest addr [192.168.131.3]
       Dest addr [192.168.131.7]
       Dest addr [192.168.131.31]
   Bytes sent (initiator:responder) [8456:5880] acl created 4
   Inbound access-list 102 applied to interface Ethernet0/0
   Inbound access-list 102 applied to interface Ethernet0/0
   Inbound access-list 102 applied to interface Ethernet0/0
   Inbound access-list 102 applied to interface Ethernet0/0
```

Example ICMP Session Verification

The following example is sample output from the **show ip access-list** command. In this example, Access Control Lists (ACLs) are created for an ICMP session on which only ping packets were issued from the host.

```
Router# show ip access-list 101
Extended IP access list 101
     permit icmp any host 192.168.133.3 time-exceeded
     permit icmp any host 192.168.133.3 unreachable
     permit icmp any host 192.168.133.3 timestamp-reply
     permit icmp any host 192.168.133.3 echo-reply (4 matches)
```

Additional References

Related Documents

Related Topic	Document Title
Cisco IOS commands	Cisco IOS Master Commands List, All Releases
CBAC information and configuration tasks	"Configuring Context-based Access Control"
Additional CBAC commands	*Cisco IOS Security Command Reference*

Standards

Standards	Title
None	--

MIBs

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

RFCs

RFCs[3]	Title
RFC 792	*Internet Control Message Protocol*
RFC 950	*Internet Standard Subnetting Procedure*
RFC 1700	*Assigned Numbers*

3 **Not all supported RFCs are listed.**

Technical Assistance

Description	Link
The Cisco Support and Documentation website provides online resources to download documentation, software, and tools. Use these resources to install and configure the software and to troubleshoot and resolve technical issues with Cisco products and technologies. Access to most tools on the Cisco Support and Documentation website requires a Cisco.com user ID and password.	http://www.cisco.com/cisco/web/support/index.html

Feature Information for Firewall Stateful Inspection of ICMP

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

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

Table 15 Feature Information for Firewall Stateful Inspection of ICMP

Feature Name	Releases	Feature Information
Firewall Stateful Inspection of ICMP	12.2(11)YU 12.2(15)T	The Firewall Stateful Inspection of ICMP feature addresses the limitation of qualifying Internet Control Management Protocol (ICMP) messages into either a malicious or benign category by allowing the Cisco IOS firewall to use stateful inspection to "trust" ICMP messages that are generated within a private network and to permit the associated ICMP replies. Thus, network administrators can debug network issues by using ICMP without concern that possible intruders may enter the network. The following commands were introduced or modified: **debug ip inspect**, **ip inspect name**.

Glossary

ACL --access control list. An ACL is a list kept by routers to control access to or from the router for a number of services (for example, to prevent packets with a certain IP address from leaving a particular interface on the router).

CBAC --Context-Based Access Control. CBAC is the name given to the Cisco IOS Firewall subsystem.

firewall --A firewall is a networking device that controls access to the network assets of your organization. Firewalls are positioned at the entrance points into your network. If your network has multiple entrance points, you must position a firewall at each point to provide effective network access control.

The most basic function of a firewall is to monitor and filter traffic. Firewalls can be simple or elaborate, depending on your network requirements. Simple firewalls are usually easier to configure and manage. However, you might require the flexibility of a more elaborate firewall.

ICMP --Internet Control Message Protocol. An ICMP is a network layer Internet protocol that reports errors and provides other information relevant to IP packet processing.

RPC --remote-procedure call. A RPC is the technological foundation of client or 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.

RTSP --Real Time Streaming Protocol. RTSP enables the controlled delivery of real-time data, such as audio and video. Sources of data can include both live data feeds, such as live audio and video, and stored content, such as prerecorded events. RTSP is designed to work with established protocols, such as RTP and HTTP.

SIP --Session Initiation Protocol. SIP is a protocol developed by the IETF MUSIC Working Group as an alternative to H.323. SIP features are compliant with IETF RFC 2543, published in March 1999. SIP equips platforms to signal the setup of voice and multimedia calls over IP networks.

SMTP --simple mail transfer protocol. SMTP is an Internet protocol providing e-mail services.

UDP --User Datagram Protocol. A UDP is a 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.

Note Refer to the *Internetworking Terms and Acronyms* for terms not included in this glossary.

Granular Protocol Inspection

The Granular Protocol Inspection feature adds flexibility to the Cisco IOS Firewall by allowing it to perform a higher degree of inspection of TCP and User Data Protocol (UDP) traffic for most RFC 1700 application types.

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest caveats and feature information, see Bug Search Tool and the release notes for your platform and software release. To find information about the features documented in this module, and to see a list of the releases in which each feature is supported, see the feature information table at the end of this module.

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

Prerequisites for Granular Inspection Protocol

- Cisco IOS Firewall software must be installed in your network.
- Access control lists (ACLs) must be applied to specified interfaces to enable the existing firewall software to function properly.

Restrictions for Granular Inspection Protocol

Port ranges cannot be specified directly in the **ip inspect name** command; use the port-to-application mapping (PAM) table.

Information About Granular Protocol Inspection

Cisco IOS Firewall

The Cisco IOS Firewall is a security-specific option that provides inspection firewall functionality and intrusion detection for every network perimeter. By delivering state-of-the-art security features such as stateful, application-based filtering; dynamic per-user authentication and authorization; and URL filtering, the Cisco IOS Firewall adds greater depth and flexibility to existing Cisco IOS security solutions including authentication, encryption, and failover.

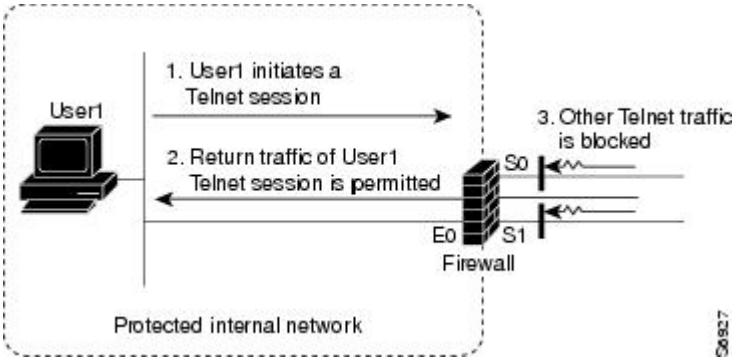
A firewall is a physical software or hardware barrier between one part of an internal network used to control access to and from external networks. This barrier is unique because it allows predefined traffic to pass through the firewall while being monitored for protocol anomalies. The difficult part is determining the criteria by which the packets are granted or denied access through the device.

As mentioned, a firewall blocks traffic and permits other types of traffic to traverse. Firewalls are not just access control lists (ACLs); rather, they are a stateful inspection application.

Granular Protocol Inspection

The Cisco IOS Firewall performs inspections for TCP and UDP traffic. For example, TCP inspections include Telnet traffic (port 23, by default) as well as all other applications on TCP such as Hypertext Transfer Protocol (HTTP), e-mail, instant message (IM) chatter, and so on. Therefore, there is no easy way to inspect Telnet traffic alone and deny all other TCP traffic.

The Granular Protocol Inspection feature allows you to specify TCP or UDP ports using the PAM table. As a result, the Cisco IOS Firewall can restrict traffic inspections to specific applications, thereby permitting a higher degree of granularity in selecting which protocols are to be permitted and denied as shown in the figure below.



Benefits

- Greater flexibility by allowing more granularity in the selection of protocols to be inspected

- Ease of use by providing for group inspection of multiple ports into a single, user-defined application keyword
- Enhanced functionality with the addition of more well-known ports, user-defined applications, and user-defined port ranges
- Improved performance and reduced CPU load resulting from focused inspection selections

How to Configure Granular Protocol Inspection

Defining Applications

Perform the following task to define your applications in the PAM table by using the **ip port-map** command.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **ip port-map** *appl-name* **port** [**tcp** | **udp**] [*port_num* | **from** *begin_port_num* **to** *end_port_num*] [**list** *acl-num*] [**description** *description_string*]
4. **exit**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Router> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Router# configure terminal`	Enters global configuration mode.

	Command or Action	Purpose
Step 3	**ip port-map** *appl-name* **port** [**tcp** | **udp**] [*port_num* | **from** *begin_port_num* **to** *end_port_num*] [**list** *acl-num*] [**description** *description_string*] **Example:** `Router(config)# ip port-map user-10 port udp from 3400 to 3433 list 22 description "test application"`	Establishes PAM entries. **Note** When defining a user application in the PAM table, you must enter the prefix user-; otherwise, the following error message appears: "Unable to add port-map entry. Names for user-defined applications must start with 'user-'." **Note** Write the text string in the following format: "C *description_string* C," where "C" is a delimiting character.
Step 4	**exit** **Example:** `Router(config)# exit`	Exits global configuration mode.

Setting Up Inspection Rules

Perform the following task to set up your inspection rules by using the **ip inspect name** command.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **ip inspect name** *inspection-name protocol* [**alert** {**on** | **off**}] [**audit-trail** {**on** | **off**}] [**timeout** *seconds*]
4. **exit**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Router> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Router# configure terminal`	Enters global configuration mode.

	Command or Action	Purpose
Step 3	**ip inspect name** *inspection-name protocol* [**alert** {**on** | **off**}] [**audit-trail** {**on** | **off**}] [**timeout** *seconds*] **Example:** `Router(config)# ip inspect name abc user-10`	Defines inspection rules. **Note** Replace the *protocol* argument with the application (PAM entry) that you just defined in the previous step. In this example, it is *user-10*.
Step 4	**exit** **Example:** `Router(config)# exit`	Exits global configuration mode.

Verifying the Configuration

Perform the following task to verify your applications and inspection rules.

SUMMARY STEPS

1. **enable**
2. **show ip port-map** [*appl-name* | **port** *port-num* [**detail**]]
3. **exit**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Router> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**show ip port-map** [*appl-name* | **port** *port-num* [**detail**]] **Example:** `Router# show ip port-map port 70 detail`	Establishes PAM entries.
Step 3	**exit** **Example:** `Router# exit`	(Optional) Exits privileged EXEC mode.

Configuration Examples for Granular Protocol Inspection

Example Defining an Application for the PAM Table

In the following example from the **ip port-map** command, a user-defined application named user-10 is defined in the PAM table for five ports using the TCP protocol. Standard access list 77 is applied to define host-specific port mapping and “TEST STRING” is the description.

```
Router# configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
Router(config)# ip port-map user-10 port tcp 4000 5000 6000 7000 8000 list 77 description
"TEST STRING"
Router(config)# end
```

Example Setting Up an Inspection Rule

The following example from the **ip inspect name** command, lists user-10 as an application with the description “TEST STRING.”

```
Router# configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
Router(config)# ip inspect name abc ?
  bootpc       Bootstrap Protocol Client
  bootps       Bootstrap Protocol Server
  cisco-fna    Cisco FNATIVE
  cisco-sys    Cisco SYSMAINT
  cisco-tna    Cisco TNATIVE
  cuseeme      CUSeeMe Protocol
  echo         Echo port
  esmtp        Extended SMTP
  finger       Finger
  fragment     IP fragment inspection
  ftp          File Transfer Protocol
  gopher       Gopher
  gtpv0        GPRS Tunneling Protocol Version 0
  gtpv1        GPRS Tunneling Protocol Version 1
  h323         H.323 Protocol (e.g, MS NetMeeting, Intel Video Phone)
  http         HTTP Protocol
  icmp         ICMP Protocol
  imap         IMAP Protocol
  imap3        Interactive Mail Access Protocol 3
  kerberos     Kerberos
  ldap         Lightweight Directory Access Protocol
  netbios-dgm  NETBIOS Datagram Service
  netshow      Microsoft NetShow Protocol
  nntp         Network News Transport Protocol
  parameter    Specify inspection parameters
  pop3         POP3 Protocol
  pwdgen       Password  Generator Protocol
  rcmd         R commands (r-exec, r-login, r-sh)
  realaudio    Real Audio Protocol
  rpc          Remote Prodedure Call Protocol
  rtsp         Real Time Streaming Protocol
  secure-http  Secure Hypertext Transfer Protocol
  sip          SIP Protocol
  skinny       Skinny Client Control Protocol
  smtp         Simple Mail Transfer Protocol
  snmp         Simple Network Management Protocol
```

```
  snmptrap     SNMP Trap
  sqlnet       SQL Net Protocol
  sqlsrv       SQL Service
  streamworks  StreamWorks Protocol
  tacacs       Login Host Protocol (TACACS)
  tacacs-ds    TACACS-Database Service
  tcp          Transmission Control Protocol
  telnet       Telnet
  tftp         TFTP Protocol
  udp          User Datagram Protocol
  vdolive      VDOLive Protocol
  user-10      TEST STRING        <----- !user-defined application!
```

In the following example from the **ip inspect name** command, an inspection rule is established for user-10:

```
Router# configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
Router(config)# ip inspect name abc user-10
Router(config)# end
```

Example Verifying the Configuration

The following example verifies your port-map configuration:

```
Router# show running-config

|
include port-map
ip port-map user-10 port tcp 4000 5000 6000 7000 8000 list 77 description "TEST STRING"
```

The following example verifies your inspection rule configuration:

```
Router# show running-config

|
include inspect
ip inspect name abc user-10
```

The following example displays information about the user-defined application called user-10.

```
Router# show ip port-map user-10
Host specific:    user-10              tcp port 4000...8000    in list 77    user defined
```

The following example displays detailed information about the user-defined application called user-10.

```
Router# show ip port-map user-10 detail
IP port-map entry for application 'user-10':
     tcp 4000...8000                   list 77 "TEST STRING"                  user defined
```

Additional References

Related Documents

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

Related Topic	Document Title
Security commands: complete command syntax, command mode, defaults, usage guidelines, and examples	*Cisco IOS Security Command Reference*

Standards

Standards	Title
None	--

MIBs

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

RFCs

RFCs	Title
None	--

Technical Assistance

Description	Link
The Cisco Support and Documentation website provides online resources to download documentation, software, and tools. Use these resources to install and configure the software and to troubleshoot and resolve technical issues with Cisco products and technologies. Access to most tools on the Cisco Support and Documentation website requires a Cisco.com user ID and password.	http://www.cisco.com/cisco/web/support/index.html

Feature Information for Granular Protocol Inspection

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

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

Table 16 Feature Information for Granular Protocol Inspection

Feature Name	Releases	Feature Information
Granular Protocol Inspection	12.3(14)T	The Granular Protocol Inspection feature adds flexibility to the Cisco IOS Firewall by allowing it to perform a higher degree of inspection of TCP and User Data Protocol (UDP) traffic for most RFC 1700 application types. The following commands were introduced or modified: **ip inspect name**, **ip port-map**, **show ip port-map**.

Glossary

firewall --A 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.

granular--Degree of componentization. Small, fine-grained components provide greater flexibility in assembling the right combination of functionality, but can be difficult to manage.

inspection rule --A rule that specifies what IP traffic (which application-layer protocols) will be inspected by CBAC at an interface.

PAM --port-to-application mapping. A flexible, per-application port mapping capability that allows the Cisco IOS Firewall to support applications running on nonstandard ports. This feature allows network administrators to customize access control for specific applications and services, in order to meet their distinct network needs.

traffic inspection --A way that CBAC inspects traffic that travels through the firewall to discover and manage state information for TCP and UDP sessions. This state information is used to create temporary openings in the firewall's access lists to allow return traffic and additional data connections for permissible sessions (sessions that originated from within the protected internal network).

UDP --User Data Protocol. A connectionless service--there are no actual sessions, so the software approximates sessions by examining the information in the packet and determining if the packet is similar to other UDP packets (for example, similar source/destination addresses and port numbers) and if the packet was detected soon after another similar UDP packet. "Soon" means within the configurable UDP idle timeout period.

and other figures included in the document are shown for illustrative purposes only. Any use of actual IP addresses or phone numbers in illustrative content is unintentional and coincidental.

TCP Out-of-Order Packet Support for Cisco IOS Firewall and Cisco IOS IPS

First Published: November 17, 2006

Last Updated: November 17, 2006

This feature allows out-of-order packets in TCP streams to be cached and reassembled before they are inspected by Cisco IOS Intrusion Prevention System (IPS) or Cisco IOS Firewall.

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest caveats and feature information, see Bug Search Tool and the release notes for your platform and software release. To find information about the features documented in this module, and to see a list of the releases in which each feature is supported, see the feature information table at the end of this module.

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

Prerequisites for TCP Out-of-Order Packet Support for Cisco IOS Firewall and Cisco IOS IPS

Cisco IOS IPS or Cisco IOS Firewall must be configured on your router.

Restrictions for TCP Out-of-Order Packet Support for Cisco IOS Firewall and Cisco IOS IPS

- The feature is enabled by default. The user must explicitly disable it. To disable TCP out-of-order packet buffering and reassembly, issue the **ip inspect tcp reassembly queue length 0** command.
- Zone-based policy firewall is not supported. Only Cisco IOS IPS and Cisco IOS Firewall application inspection can support out-of-order TCP packets.

Information About TCP Out-of-Order Packet Support for Cisco IOS Firewall and Cisco IOS IPS

How TCP Out-of-Order Packet Support Works

Cisco IOS Firewall and IPS track packets in TCP connections. If configured to look into the application data of the packets, Cisco IOS Firewall and IPS expect the TCP packets to arrive in the correct order because some data items are split across segments. When packets arrive out of order, they are dropped by the firewall or IPS. Dropping out-of-order packets can cause significant delays in end applications because packets are dropped only after the retransmission timer expires (on behalf of the sender).

Out-of-order TCP packet support enables Cisco IOS Firewall and IPS to hold a copy of the out-of-order packet in a buffer (whose size is configurable with a maximum of 1024 packets per session). The original packet passes through the router and reaches its destination, but the firewall or IPS do not execute on the packet. When the next packet arrives, the firewall or IPS look for that packet to “fill the hole,” providing a consecutive sequence of segments. If this packet does not fulfill that requirement, it is processed as an out-of-order packet; when another packet arrives and provides a consecutive sequence of segments, it is processed by the firewall or IPS.

How to Configure Cisco IOS Firewall or IPS to Handle TCP Out-of-Order Packets

Changing Default TCP Out-of-Order Packet Parameters

Use this task to change any of the predefined parameters that instruct Cisco IOS Firewall application inspection or Cisco IOS IPS how to handle out-of-order TCP packets.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. ip inspect tcp reassembly {[queue length packet-number] [timeout seconds] [memory limit size-in-kb] [alarm {on | off}]}

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Router> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Router# configure terminal`	Enters global configuration mode.
Step 3	ip inspect tcp reassembly {[queue length packet-number] [timeout seconds] [memory limit size-in-kb] [alarm {on | off}]} **Example:** `Router(config)# ip inspect tcp reassembly queue length 10 timeout 8`	Sets parameters that define how a Cisco IOS IPS handles out-of-order TCP packets. • **queue length** *packet-number*-- Maximum number of out-of-order packets that can be held per queue (buffer). Note that there are 2 queues per session. Available value range: 0 to 1024. Default value: 16. If the queue length is set to 0, all out-of-order packets are dropped. • **timeout** *seconds*-- Number of seconds the TCP reassembly module will hold out-of-order segments waiting for the first segment missing in the sequence. After the timeout timer has expired, a retry timer is started. The value for the retry timer is four times the configured timeout value. • **memory limit** *size-in-kb* --Maximum allowed memory use by the TCP reassembly module. • **alarm** {**on** | **off**}--If enabled, a syslog message is generated when an out-of-order packet is dropped. Default value: **on**

Configuration Examples for TCP Out-of-Order Packet Parameters

• Example Verifying TCP Out-of-Order Packets, page 100

Example Verifying TCP Out-of-Order Packets

The following example shows how to instruct Cisco IOS IPS how to handle out of order packets for TCP connections:

```
Router(config)#
ip inspect tcp reassembly queue length 18
Router(config)#
ip inspect tcp reassembly memory limit 200
```

The following sample output displays the configured out-of-order packet parameters:

```
Router# show ip ips statistics
Signature Statistics [process switch:fast switch]
Signature 1000: 324 packets checked: [124:200]
Signature 1024: 100 packets checked: [0:100]
Interfaces configured for ips 0
Session creations since subsystem startup or last reset 0
Current session counts (estab/half-open/terminating) [0:0:0]
Maxever session counts (estab/half-open/terminating) [0:0:0]
Last session created never
Last statistic reset never
TCP reassembly statistics
received 200 packets out-of-order; dropped 25
peak memory usage; 200 KB; current usage: 154 KB
peak queue length 18
```

Additional References

Related Documents

Related Topic	Document Title
Cisco IOS commands	Cisco IOS Master Commands List, All Releases
IPS configuration	"IPS 5.x Signature Format Support and Usability Enhancements"
Firewall IPS commands	*Cisco IOS Security Command Reference*

Standards

Standard	Title
None	--

MIBs

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

RFCs

RFC	Title
None	--

Technical Assistance

Description	Link
The Cisco Support and Documentation website provides online resources to download documentation, software, and tools. Use these resources to install and configure the software and to troubleshoot and resolve technical issues with Cisco products and technologies. Access to most tools on the Cisco Support and Documentation website requires a Cisco.com user ID and password.	http://www.cisco.com/cisco/web/support/index.html

Feature Information for TCP Out-of-Order Packet Support for Cisco IOS Firewall and IPS

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

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

Table 17 Feature Information for TCP Out-of-Order Support

Feature Name	Releases	Feature Information
TCP Out-of-Order Packet Support for Cisco IOS Firewall and Cisco IOS IPS	12.4(11)T	This feature allows out-of-order packets in TCP streams to be cached and reassembled before they are inspected by Cisco IOS Intrusion Prevention System (IPS) or Cisco IOS Firewall. The following command was introduced by this feature: **ip inspect tcp reassembly**.