



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.

Feature History for Granular Protocol Inspection

Release	Modification
12.3(14)T	This feature was introduced.

Finding Support Information for Platforms and Cisco IOS Software Images

Use Cisco Feature Navigator to find information about platform support and Cisco IOS software image support. Access Cisco Feature Navigator at <http://www.cisco.com/go/fn>. You must have an account on Cisco.com. If you do not have an account or have forgotten your username or password, click **Cancel** at the login dialog box and follow the instructions that appear.

Contents

- [Prerequisites for Granular Inspection Protocol, page 1](#)
- [Restrictions for Granular Inspection Protocol, page 2](#)
- [Information About Granular Protocol Inspection, page 2](#)
- [How to Configure Granular Protocol Inspection, page 5](#)
- [Configuration Examples for Granular Protocol Inspection, page 8](#)
- [Additional References, page 11](#)
- [Command Reference, page 12](#)
- [Glossary, page 33](#)

Prerequisites for Granular Inspection Protocol

- Cisco IOS Firewall software must be installed in your network.



Corporate Headquarters:
Cisco Systems, Inc., 170 West Tasman Drive, San Jose, CA 95134-1706 USA

Copyright © 2005 Cisco Systems, Inc. All rights reserved.

- 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

To use the Granular Protocol Inspection feature, you need to understand the following concepts:

- [Cisco IOS Firewall, page 3](#)
- [Granular Protocol Inspection, page 3](#)
- [Benefits, page 4](#)

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.

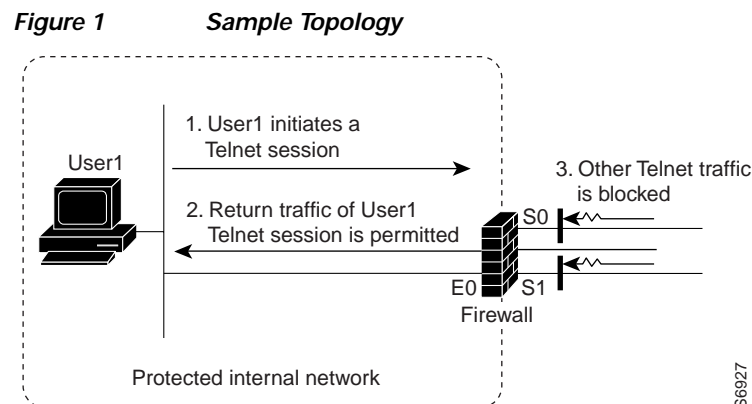
A firewall is a physical software or hardware barrier between one part of an internal network used to control access to and from external networks. This barrier is unique because it allows predefined traffic to pass through the firewall while being monitored for protocol anomalies. The difficult part 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 [Figure 1](#).



Benefits

The Granular Protocol Inspection feature provides the following 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

This section contains the following procedures:

- [Defining Applications, page 5](#) (required)
- [Setting Up Inspection Rules, page 6](#) (required)
- [Verifying the Configuration, page 6](#) (optional)

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.
Step 3	ip port-map <i>appl-name</i> port [tcp udp] [<i>port_num</i> from <i>begin_port_num</i> to <i>end_port_num</i>] [list <i>acl-num</i>] [description <i>description_string</i>] 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: " <i>description_string</i> 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.
Step 3	ip inspect name <i>inspection-name protocol</i> [alert {on off}] [audit-trail {on off}] [timeout seconds] Example: Router(config)# ip inspect name abc user-10	Defines inspection rules. Note Replace the <i>protocol</i> argument with the application (PAM entry) that you just defined in the previous step. In this example, it is <i>user-10</i> .
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	<pre>enable</pre> <p>Example: Router> enable </p>	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	<pre>show ip port-map [appl-name port port-num [detail]]</pre> <p>Example: Router# show ip port-map port 70 detail </p>	Establishes PAM entries.
Step 3	<pre>exit</pre> <p>Example: Router# exit </p>	(Optional) Exits privileged EXEC mode.

Configuration Examples for Granular Protocol Inspection

This section contains the following configuration examples:

- [Defining an Application for the PAM Table: Example, page 8](#)
- [Setting Up an Inspection Rule: Example, page 8](#)
- [Verifying the Configuration: Example, page 9](#)

Defining an Application for the PAM Table: Example

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
```

Setting Up an Inspection Rule: Example

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 SYSMANT
cisco-tna    Cisco TNATIVE
cuseeme     CUSeeMe Protocol
echo        Echo port
esmtplib    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 Procedure 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

```

Verifying the Configuration: Example

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

The following sections provide references related to the Granular Protocol Inspection feature.

Related Documents

Related Topic	Document Title
Security commands: complete command syntax, command mode, defaults, usage guidelines, and examples	Cisco IOS Security Command Reference , Release 12.3 T
Security features including firewalls and authentication	Cisco IOS Security Configuration Guide , Release 12.3

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
No new or modified RFCs are supported by this feature, and support for existing RFCs has not been modified by this feature.	—

Technical Assistance

Description	Link
The Cisco Technical Support website contains thousands of 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/techsupport

Command Reference

This section documents modified commands.

- [ip inspect name](#)
- [ip port-map](#)
- [show ip port-map](#)

ip inspect name

To define a set of inspection rules, use the **ip inspect name** command in global configuration mode. To remove the inspection rule for a protocol or to remove the entire set of inspection rules, use the **no** form of this command.

```
ip inspect name inspection-name [parameter max-sessions number] protocol [alert {on | off}]
[audit-trail {on | off}] [timeout seconds]
```

```
no ip inspect name inspection-name [parameter max-sessions number] protocol [alert {on | off}]
[audit-trail {on | off}] [timeout seconds]
```

HTTP Inspection Syntax

```
ip inspect name inspection-name http [urlfilter] [java-list access-list] [alert {on | off}]
[audit-trail {on | off}] [timeout seconds]
```

```
no ip inspect name inspection-name protocol
```

SMTP and ESMTP Inspection Syntax

```
ip inspect name inspection-name {smtp | esmtip} [alert {on | off}] [audit-trail {on | off}]
[max-data number] [timeout seconds]
```

remote-procedure call (RPC) Inspection Syntax

```
ip inspect name inspection-name [parameter max-sessions number] rpc program-number
number [wait-time minutes] [alert {on | off}] [audit-trail {on | off}] [timeout seconds]
```

```
no ip inspect name inspection-name protocol
```

POP3/IMAP Inspection Syntax

```
ip inspect name inspection-name imap [alert {on | off}] [audit-trail {on | off}] [reset]
[secure-login] [timeout number]
```

```
ip inspect name inspection-name pop3 [alert {on | off}] [audit-trail {on | off}] [reset]
[secure-login] [timeout number]
```

Fragment Inspection Syntax

```
ip inspect name inspection-name [parameter max-sessions number] fragment [max number
timeout seconds]
```

```
no ip inspect name inspection-name [parameter max-sessions number] fragment [max number
timeout seconds]
```

Application Firewall Provisioning Syntax

```
ip inspect name inspection-name [parameter max-sessions number] appfw policy-name
```

```
no ip inspect name inspection-name [parameter max-sessions number] appfw policy-name
```

User-Defined Application Syntax

ip inspect name *inspection-name user-10* [**alert** {**on** | **off**}] [**audit-trail** {**on** | **off**}] [**timeout** *seconds*]

no ip inspect name *inspection-name user-10* [**alert** {**on** | **off**}] [**audit-trail** {**on** | **off**}] [**timeout** *seconds*]

Session Limiting Syntax

no ip inspect name *inspection-name* [**parameter max-sessions** *number*]

Syntax Description

<i>inspection-name</i>	Names the set of inspection rules. If you want to add a protocol to an existing set of rules, use the same <i>inspection-name</i> as the existing set of rules. Note The <i>inspection-name</i> cannot exceed 16 characters; otherwise, the name will be truncated to the 16-character limit.
parameter max-sessions <i>number</i>	(Optional) Limits the number of established firewall sessions that a firewall rule creates. The default is that there is no limit to the number of firewall sessions.
<i>protocol</i>	A protocol keyword listed in Table 1 or Table 2 .
alert { on off }	(Optional) For each inspected protocol, the generation of alert messages can be set be on or off . If no option is selected, alerts are generated on the basis of the setting of the ip inspect alert-off command.
audit-trail { on off }	(Optional) For each inspected protocol, audit trail can be set on or off . If no option is selected, an audit trail message are generated on the basis of the setting of the ip inspect audit-trail command.
timeout <i>seconds</i>	(Optional) To override the global TCP or User Datagram Protocol (UDP), or Internet Control Message Protocol (ICMP) idle timeouts for the specified protocol, specify the number of seconds for a different idle timeout. This timeout overrides the global TCP, UDP, or ICMP timeouts but will not override the global Domain Name System (DNS) timeout.
http	Specifies the HTTP protocol for Java applet blocking.
urlfilter	(Optional) Associates URL filtering with HTTP inspection.
java-list <i>access-list</i>	(Optional) Specifies the numbered standard access list to use to determine “friendly” sites. This keyword is available only for the HTTP protocol, for Java applet blocking. Java blocking only works with numbered standard access lists.
smtp esmtplib	Specifies the protocol being used to inspect the traffic.
max-data <i>number</i>	(Optional) Specifies the maximum number of bytes (data) that can be transferred in a single Simple Mail Transport Protocol (SMTP) session. After the maximum value is exceeded, the firewall logs an alert message and closes the session. Default value: 20 MB
rpc program-number <i>number</i>	Specifies the program number to permit. This keyword is available only for the remote-procedure call protocol.

wait-time <i>minutes</i>	(Optional) Specifies the number of minutes to keep a small hole in the firewall to allow subsequent connections from the same source address and to the same destination address and port. The default wait-time is zero minutes. This keyword is available only for the remote-procedure call (RPC) protocol.
reset	(Optional) Resets the TCP connection if the client enters a non-protocol command before authentication is complete.
secure-login	(Optional) Causes a user at a non-secure location to use encryption for authentication.
imap	Specifies that the Internet Message Access Protocol (IMAP) is being used.
pop3	Specifies that the Post Office Protocol, Version 3 (POP3) is being used.
fragment	Specifies fragment inspection for the named rule.
max <i>number</i>	(Optional) Specifies the maximum number of unassembled packets for which state information (structures) is allocated by Cisco IOS software. Unassembled packets are packets that arrive at the router interface before the initial packet for a session. The acceptable range is 50 through 10000. The default is 256 state entries. Memory is allocated for the state structures, and setting this value to a larger number may cause memory resources to be exhausted.
timeout <i>seconds</i> (fragmentation)	(Optional) Configures the number of seconds that a packet state structure remains active. When the timeout value expires, the router drops the unassembled packet, freeing that structure for use by another packet. The default timeout value is 1 second. If this number is set to a value greater than 1 second, it is automatically adjusted by the Cisco IOS software when the number of free state structures goes below certain thresholds: when the number of free states is fewer than 32, the timeout is divided by 2. When the number of free states is fewer than 16, the timeout is set to 1 second.
appfw	Specifies application firewall provisioning.
<i>policy-name</i>	Application firewall policy name. Note This name must match the name specified via the appfw policy-name command.
<i>appname</i>	Specifies a user- or a system-defined application; for example, user-payroll-sap and user-sametime . Application names can contain hyphens and underscores; however, a user-defined application must have the prefix user- in its title.
port	Specifies the port range for an application.
tcp udp	Specifies the protocol being used to inspect the traffic.
from <i>begin_port_num to end_port_num</i> <i>port_num1 ...</i>	Specifies the starting and ending port numbers or a range of ports from 1 to 5. You must use the from and to keywords together.
list <i>acl_list_num</i>	(Optional) Specifies an access control list number. Only standard ACLs are supported.
description <i>description_string</i>	(Optional) Specifies a description of up to 40 characters.

<i>user-10</i>	Represents a user-defined application in the port-to-application mapping (PAM) table of the ip port-map command.
router-traffic	(Optional) Enables inspection of traffic destined to or originated from a router. Applicable only for H.323, TCP, and UDP protocols. For the command format, see the Note after Table 1 .

Defaults

No inspection rules are defined until you define them using this command.

no ip inspect-name protocol removes the inspection rule for the specified protocol.

no ip inspect name removes the entire set of inspection rules.

Command Modes

Global configuration

Command History

Release	Modification
11.2 P	This command was introduced.
12.0(5)T	Introduced configurable alert and audit trail, IP fragmentation checking, and NetShow protocol support.
12.2(11)YU	Support was added for ICMP and SIP protocols and the urlfilter keyword was added to the HTTP inspection syntax.
12.2(15)T	Support was added for ICMP, SIP protocols, and the urlfilter keyword was integrated into Cisco IOS Release 12.2(15)T.
12.3(1)	Skinny protocol support was added.
12.3(7)T	Extended Simple Mail Transfer Protocol (ESMTP) protocol support was added.
12.3(14)T	The appfw keyword and the <i>policy-name</i> argument were added to support application firewall provisioning. The parameter max-sessions , secure-login , reset , and router-traffic keywords were added. Support for a larger list of protocols including user-defined applications was added.

Usage Guidelines

To define a set of inspection rules, enter this command for each protocol that you want the Cisco IOS firewall to inspect, using the same *inspection-name*. Give each set of inspection rules a unique *inspection-name*, which should not exceed the 16-character limit. Define either one or two sets of rules per interface—you can define one set to examine both inbound and outbound traffic, or you can define two sets: one for outbound traffic and one for inbound traffic.

To define a single set of inspection rules, configure inspection for all the desired application-layer protocols, and for ICMP, TCP, and UDP, or as desired. This combination of TCP, UDP, and application-layer protocols join together to form a single set of inspection rules with a unique name. (There are no application-layer protocols associated with ICMP.)

To remove the inspection rule for a protocol, use the **no** form of this command with the specified inspection name and protocol; to remove the entire set of inspection rules, use the **no** form of this command only; that is, do not list any inspection names or protocols.

In general, when inspection is configured for a protocol, return traffic entering the internal network will be permitted only if the packets are part of a valid, existing session for which state information is being maintained.

Table 1 Protocol Keywords—Transport-Layer and Network-Layer Protocols

Protocol	Keyword
ICMP	icmp
TCP	tcp
UDP	udp

Note The TCP, UDP, and H.323 protocols support the **router-traffic** keyword, which enables inspection of traffic destined to or originated from a router. The command format is as follows:

```
ip inspect name inspection-name { TCP | UDP | H323 } [alert { on | off }] [audit-trail { on | off }][router-traffic][timeout seconds]
```

TCP and UDP Inspection

You can configure TCP and UDP inspection to permit TCP and UDP packets to enter the internal network through the firewall, even if the application-layer protocol is not configured to be inspected. However, TCP and UDP inspection do not recognize application-specific commands, and therefore might not permit all return packets for an application, particularly if the return packets have a different port number from the previous exiting packet.

Any application-layer protocol that is inspected will take precedence over the TCP or UDP packet inspection. For example, if inspection is configured for FTP, all control channel information will be recorded in the state table, and all FTP traffic will be permitted back through the firewall if the control channel information is valid for the state of the FTP session. The fact that TCP inspection is configured is irrelevant.

With TCP and UDP inspection, packets entering the network must exactly match an existing session: the entering packets must have the same source or destination addresses and source or destination port numbers as the exiting packet (but reversed). Otherwise, the entering packets will be blocked at the interface.

Granular protocol inspection allows you to specify TCP or UDP ports by using the PAM table. This eliminates having to inspect all applications running under TCP or UDP and the need for multiple access control lists (ACLs) to filter the traffic.

Using the PAM table, you simply pick an existing application or define a new one for inspection thereby simplifying ACL configuration.

ICMP Inspection

An ICMP inspection session is on the basis of the source address of the inside host that originates the ICMP packet. Dynamic access control lists (ACLs) are created for return ICMP packets of the allowed types (echo-reply, time-exceeded, destination unreachable, and timestamp reply) for each session. There are no port numbers associated with an ICMP session, and the permitted IP address of the return packet is wild-carded in the ACL. The wildcard address is because the IP address of the return packet cannot be known in advance for time-exceeded and destination-unreachable replies. These replies can come from intermediate devices rather than the intended destination.

Application-Layer Protocol Inspection

In general, if you configure inspection for an application-layer protocol, packets for that protocol should be permitted to exit the firewall (by configuring the correct access control list), and packets for that protocol will only be allowed back in through the firewall if they belong to a valid existing session. Each protocol packet is inspected to maintain information about the session state.

Java, H.323, RPC, SIP, and SMTP inspection have additional information, described in the next five sections. [Table 2](#) lists the supported application-layer protocols.

Table 2 Protocol Keywords—Application-Layer Protocols

Protocol	Keyword
Application Firewall	appfw
CU-SeeMe	cuseeme
ESMTP	smtp
FTP	ftp
IMAP	imap
Java	http
H.323	h323
Microsoft NetShow	netshow
POP3	pop3
RealAudio	realaudio
RPC	rpc
SIP	sip
Simple Mail Transfer Protocol (SMTP)	smtp
Skinny Client Control Protocol (SCCP)	skinny
StreamWorks	streamworks
Structured Query Language*Net (SQL*Net)	sqlnet
TFTP	tftp
UNIX R commands (rlogin, rexec, rsh)	rcmd
VDOLive	vdolive
WORD	user-defined application name; use prefix -user
	Note All applications that appear under the show ip port-map command are supported.

Java Inspection

Java inspection enables Java applet filtering at the firewall. Java applet filtering distinguishes between trusted and untrusted applets by relying on a list of external sites that you designate as “friendly.” If an applet is from a friendly site, the firewall allows the applet through. If the applet is not from a friendly site, the applet will be blocked. Alternately, you could permit applets from all sites except sites specifically designated as “hostile.”

**Note**

Before you configure Java inspection, you must configure a numbered standard access list that defines “friendly” and “hostile” external sites. You configure this numbered standard access list to permit traffic from friendly sites, and to deny traffic from hostile sites. If you do not configure a numbered standard access list, but use a “placeholder” access list in the **ip inspect name** *inspection-name* **http** command, all Java applets will be blocked.

**Note**

Java blocking forces a strict order on TCP packets. To properly verify that Java applets are not in the response, a firewall will drop any TCP packet that is out of order. Because the network—not the firewall—determines how packets are routed, the firewall cannot control the order of the packets; the firewall can only drop and retransmit all TCP packets that are not in order.

**Caution**

Context-Based Access Control (CBAC) does not detect or block encapsulated Java applets. Therefore, Java applets that are wrapped or encapsulated, such as applets in .zip or .jar format, are *not* blocked at the firewall. CBAC also does not detect or block applets loaded via FTP, gopher, or HTTP on a nonstandard port.

H.323 Inspection

If you want CBAC inspection to work with NetMeeting 2.0 traffic (an H.323 application-layer protocol), you must also configure inspection for TCP, as described in the chapter “Configuring Context-Based Access Control” in the *Cisco IOS Security Configuration Guide*. This requirement exists because NetMeeting 2.0 uses an additional TCP channel not defined in the H.323 specification.

RPC Inspection

RPC inspection allows the specification of various program numbers. You can define multiple program numbers by creating multiple entries for RPC inspection, each with a different program number. If a program number is specified, all traffic for that program number will be permitted. If a program number is not specified, all traffic for that program number will be blocked. For example, if you created an RPC entry with the NFS program number, all NFS traffic will be allowed through the firewall.

SIP Inspection

You can configure SIP inspection to permit media sessions associated with SIP-signaled calls to traverse the firewall. Because SIP is frequently used to signal both incoming and outgoing calls, it is often necessary to configure SIP inspection in both directions on a firewall (both from the protected internal network and from the external network). Because inspection of traffic from the external network is not done with most protocols, it may be necessary to create an additional inspection rule to cause only SIP inspection to be performed on traffic coming from the external network.

SMTP Inspection

SMTP inspection causes SMTP commands to be inspected for illegal commands. Packets with illegal commands are modified to a “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
- VRFY

ESMTP Inspection

Like SMTP, ESMTP inspection also causes the commands to be inspected for illegal commands. Packets with illegal commands are modified to a “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 ESMTP command is any command except the following:

- AUTH
- DATA
- EHLO
- ETRN
- HELO
- HELP
- MAIL
- NOOP
- QUIT
- RCPT
- RSET
- SAML
- SEND
- SOML
- VRFY

In addition to inspecting commands, the ESMTP firewall also inspects the following extensions via 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

SMTP and ESMTP cannot exist simultaneously. An attempt to configure both protocols will result in an error message.

Use of the `urlfilter` Keyword

If you specify the **urlfilter** keyword, the Cisco IOS Firewall will interact with a URL filtering software to control web traffic for a given host or user on the basis of a specified security policy.



Note

Enabling HTTP inspection with or without any option 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.

Use of the `timeout` Keyword

If you specify a timeout for any of the transport-layer or application-layer protocols, the timeout will override the global idle timeout for the interface to which the set of inspection rules is applied.

If the protocol is TCP or a TCP application-layer protocol, the timeout will override the global TCP idle timeout. If the protocol is UDP or a UDP application-layer protocol, the timeout will override the global UDP idle timeout.

If you do not specify a timeout for a protocol, the timeout value applied to a new session of that protocol will be taken from the corresponding TCP or UDP global timeout value valid at the time of session creation.

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 one 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.

IP Fragmentation Inspection

CBAC inspection rules can help protect hosts against certain denial-of-service attacks involving fragmented IP packets. Even though the firewall keeps an attacker from making actual connections to a given host, the attacker may still be able to disrupt services provided by that host. This is done by sending many noninitial IP fragments or by sending complete fragmented packets through a router with an ACL that filters the first fragment of a fragmented packet. These fragments can tie up resources on the target host as it tries to reassemble the incomplete packets.

Using fragmentation inspection, the firewall maintains an *interfragment state* (structure) for IP traffic. Noninitial fragments are discarded unless the corresponding initial fragment was permitted to pass through the firewall. Noninitial fragments received before the corresponding initial fragments are discarded.

**Note**

Fragmentation inspection can have undesirable effects in certain cases, because it can result in the firewall discarding any packet whose fragments arrive out of order. There are many circumstances that can cause out-of-order delivery of legitimate fragments. Apply fragmentation inspection in situations where legitimate fragments, which are likely to arrive out of order, might have a severe performance impact.

Because routers running Cisco IOS software are used in a very large variety of networks, and because the CBAC feature is often used to isolate parts of internal networks from one another, the fragmentation inspection feature is not enabled by default. Fragmentation detection must be explicitly enabled for an inspection rule using the **ip inspect name** command. Unfragmented traffic is never discarded because it lacks a fragment state. Even when the system is under heavy attack with fragmented packets, legitimate fragmented traffic, if any, will still get some fraction of the firewall's fragment state resources, and legitimate, unfragmented traffic can flow through the firewall unimpeded.

Application Firewall Provisioning

Application firewall provisioning allows you to configure your Cisco IOS Firewall to detect and prohibit a specific protocol type of traffic.

Most firewalls provide only packet filtering capabilities that simply permit or deny traffic without inspecting the data stream; the Cisco IOS application firewall can detect whether or not a packet is in compliance with given HTTP protocol. If the packet is determined to be unauthorized, it will be dropped, the connection will be reset, and a syslog message will be generated, as appropriate.

User-Defined Applications

You can define your own applications and enter them into the port-to-application mapping (PAM) table using the **ip port-map** command. Then you set up your inspection rules by inserting your user-defined application as a value for the *protocol* argument in the **ip inspect name** command.

Session Limiting

Users can limit the number of established firewall sessions that a firewall rule creates by setting the "max-sessions" threshold. A session counter is maintained for each firewall interface. When a session count exceeds the specified threshold, an alert FW-4-SESSION_THRESHOLD_EXCEEDED message is logged to the syslog server and no new sessions can be created.

Examples

The following example causes the software to inspect TCP sessions and UDP sessions, and to specifically allow CU-SeeMe, FTP, and RPC traffic back through the firewall for existing sessions only. For UDP traffic, audit-trail is on. For FTP traffic, the idle timeout is set to override the global TCP idle timeout. For RPC traffic, program numbers 100003, 100005, and 100021 are permitted.

```
ip inspect name myrules tcp
ip inspect name myrules udp audit-trail on
ip inspect name myrules cuseeme
ip inspect name myrules ftp timeout 120
ip inspect name myrules rpc program-number 100003
ip inspect name myrules rpc program-number 100005
ip inspect name myrules rpc program-number 100021
```

The following example adds fragment checking to software inspection of TCP and UDP sessions for the rule named “myrules.” In this example, the firewall software will allocate 100 state structures, and the timeout value for dropping unassembled packets is set to 4 seconds. If 100 initial fragments for 100 different packets are sent through the router, all of the state structures will be used up. The initial fragment for packet 101 will be dropped. Additionally, if the number of free state structures (structures available for use by unassembled packets) drops below the threshold values, 32 or 16, the timeout value is automatically reduced to 2 or 1, respectively. Changing the timeout value frees up packet state structures more quickly.

```
ip inspect name myrules tcp
ip inspect name myrules udp audit-trail on
ip inspect name myrules cuseeme
ip inspect name myrules ftp timeout 120
ip inspect name myrules rpc program-number 100003
ip inspect name myrules rpc program-number 100005
ip inspect name myrules rpc program-number 100021
ip inspect name myrules fragment max 100 timeout 4
```

The following firewall and SIP 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
```

The following example shows two configured inspections named fw_only and fw_urlf; URL filtering will work only on the traffic that is inspected by fw_urlf. Note that the **java-list access-list** option has been enabled, which disables java scanning.

```
ip inspect name fw_only http java-list 51 timeout 30
interface e0
  ip inspect fw_only in
!
ip inspect name fw_urlf http urlfilter java-list 51 timeout 30
interface e1
  ip inspect fw_urlf in
```

The following example shows how to define the HTTP application firewall policy mypolicy. This policy includes all supported HTTP policy rules. This example also includes sample output from the **show appfw configuration** and **show ip inspect config** commands, which allow you to verify the configured setting for the application policy.

```
! Define the HTTP policy.
appfw policy-name mypolicy
  application http
  strict-http action allow alarm
  content-length maximum 1 action allow alarm
  content-type-verification match-req-rsp action allow alarm
  max-header-length request 1 response 1 action allow alarm
  max-uri-length 1 action allow alarm
  port-misuse default action allow alarm
```

```

request-method rfc default action allow alarm
request-method extension default action allow alarm
transfer-encoding type default action allow alarm
!
!
! Apply the policy to an inspection rule.
ip inspect name firewall appfw mypolicy
ip inspect name firewall http
!
!
! Apply the inspection rule to all HTTP traffic entering the FastEthernet0/0 interface.
interface FastEthernet0/0
 ip inspect firewall in
!
!
! Issue the show appfw configuration command and the show ip inspect config command after
the inspection rule "mypolicy" is applied to all incoming HTTP traffic on the
FastEthernet0/0 interface.
!
Router# show appfw configuration

Application Firewall Rule configuration
Application Policy name mypolicy
  Application http
    strict-http action allow alarm
    content-length minimum 0 maximum 1 action allow alarm
    content-type-verification match-req-rsp action allow alarm
    max-header-length request length 1 response length 1 action allow alarm
    max-uri-length 1 action allow alarm
    port-misuse default action allow alarm
    request-method rfc default action allow alarm
    request-method extension default action allow alarm
    transfer-encoding default action allow alarm

Router# show ip inspect config

Session audit trail is disabled
Session alert is enabled
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 firewall
http alert is on audit-trail is off timeout 3600

```

Related Commands

Command	Description
ip inspect	Applies a set of inspection rules to an interface.
ip inspect alert-off	Disables CBAC alert messages.
ip inspect audit trail	Turns on CBAC audit trail messages, which will be displayed on the console after each CBAC session close.

ip port-map

To establish port-to-application mapping (PAM), use the **ip port-map** command in global configuration mode. To delete user-defined PAM entries, use the **no** form of this command.

```
ip port-map appl-name port [tcp / udp] [port_num / from begin_port_num to end_port_num] [list
acl-num] [description description_string]
```

```
no ip port-map appl-name port [tcp / udp] [port_num / from begin_port_num to end_port_num] [list
acl-num] [description description_string]
```

Syntax Description		
appl-name		Specifies the name of the application with which to apply the port mapping. An application name can contain an underscore or a hyphen. An application can also be system or user-defined. However, a user-defined application must have the prefix user- in it; for example, user-payroll , user-sales , or user-10 . Otherwise, the following error message appears: “Unable to add port-map entry. Names for user-defined applications must start with 'user-'.”
port		Indicates that a port number maps to the application. You can specify up to five port numbers for each port.
tcp / udp		(Optional) Specifies the protocol for the application. For well-known applications (and those existing already under PAM), you can omit these keywords and the system assumes the standard protocol for that application. However, for user-defined applications, you must specify either tcp or udp .
port_num		(Optional) Identifies a port number in the range 1 to 65535.
from <i>begin_port_num</i> to <i>end_port_num</i>		(Optional) Specifies a range of port numbers. You must use the from and to keywords together.
list <i>acl-num</i>		(Optional) Indicates that the port mapping information applies to a specific host or subnet by associating it to an access control list (ACL) number used with PAM.
description <i>description_string</i>		(Optional) Specifies a description of up to 40 characters.
	Note	Write the text string in the following format: “ <i>C description_string C</i> ,” where “ <i>C</i> ” is a delimiting character.

Defaults No default behavior or values

Command Modes Global configuration

Command History	Release	Modification
	12.0(5)T	This command was introduced.

Release	Modification
12.3(1)	Skippy Client Control Protocol (SCCP) support was added.
12.3(14)T	Support was added for the following: <ul style="list-style-type: none"> • User-defined application names • User-specified descriptions • Port ranges • tcp and udp keywords • from <i>begin_port_num</i> to <i>end_port_num</i> keyword-argument combination • description <i>description_string</i> keyword-argument combination

Usage Guidelines

The **ip port-map** command associates TCP or User Datagram Protocol (UDP) port numbers with applications or services, establishing a table of default port mapping information at the firewall. This information is used to support network environments that run services using ports that are different from the registered or well-known ports associated with a service or application.

When you issue the **no** form of the command, include all the parameters needed to remove the entry matching that specific set of parameters. For example, if you issued **no ip port-map appl-name**, then all entries for that application are removed.

The port mapping information in the PAM table is of one of three types:

- System-defined
- User-defined
- Host-specific

System-Defined Port Mapping

Initially, PAM creates a set of system-defined entries in the mapping table using well-known or registered port mapping information set up during the system start-up. The Cisco IOS Firewall Context-Based Access Control (CBAC) feature requires the system-defined port mapping information to function properly.

You can delete or modify system-defined port mapping information. Use the **no** form of the command for deletion and the regular form of the command to remap information to another application.

You can also add new port numbers to system-defined applications. However, for some system-defined applications like HTTP and Simple Mail Transfer Protocol (SMTP), in which the firewall inspects deeper into packets, their protocol (UDP or TCP) cannot be changed from that defined in the system. In those instances, error messages display.

[Table 3](#) lists some default system-defined services and applications in the PAM table. (Use the **show ip port-map** command for the complete list.)

Table 3 System-Defined Port Mapping

Application Name	Well-Known or Registered Port Number	Protocol Description
cuseeme	7648	CU-SeeMe Protocol
exec	512	Remote Process Execution

Table 3 System-Defined Port Mapping (continued)

Application Name	Well-Known or Registered Port Number	Protocol Description
ftp	21	File Transfer Protocol (control port)
h323	1720	H.323 Protocol (for example, MS NetMeeting, Intel Video Phone)
http	80	Hypertext Transfer Protocol
login	513	Remote login
msrpc	135	Microsoft Remote Procedure Call
netshow	1755	Microsoft NetShow
real-audio-video	7070	RealAudio and RealVideo
sccp	2000	Skinny Client Control Protocol (SCCP)
smtp	25	Simple Mail Transfer Protocol (SMTP)
sql-net	1521	SQL-NET
streamworks	1558	StreamWorks Protocol
sunrpc	111	SUN Remote Procedure Call
tftp	69	Trivial File Transfer Protocol
vdolive	7000	VDOLive Protocol

**Note**

You can override system-defined entries for a specific host or subnet using the **list** *acl-num* option in the **ip port-map** command.

User-Defined Port Mapping

Network applications that use nonstandard ports require user-defined entries in the mapping table. Use the **ip port-map** command to create default user-defined entries in the PAM table. These entries automatically appear as an option for the **ip inspect name** command to facilitate the creation of inspection rules.

You can specify up to five separate port numbers for each port-map in a single entry. You can also specify a port range in a single entry. However, you may not specify both single port numbers and port ranges in the same entry.

**Note**

If you try to map an application to a system-defined port, a message appears warning you of a mapping conflict. Delete the system-defined entry before mapping it to another application. Deleted system defined mappings appear in the running-configuration in their **no ip port-map** form.

Use the **no** form of the **ip port-map** command to delete user-defined entries from the PAM table. To remove a single mapping, use the **no** form of the command with all its parameters.

To overwrite an existing user-defined port mapping, use the **ip port-map** command to associate another service or application with the specific port.

Multiple commands for the same application name are cumulative.

If you assign the same port number to a new application, the new entry replaces the existing entry and it no longer appears in the running configuration. You receive a message about the remapping.

You cannot specify a port number that is in a range assigned to another application; however, you can specify a range that takes over one singly allocated port, or fully overlaps another range.

You cannot specify overlapping port ranges.

Host-Specific Port Mapping

User-defined entries in the mapping table can include host-specific mapping information, which establishes port mapping information for specific hosts or subnets. In some environments, it might be necessary to override the default port mapping information for a specific host or subnet, including a system-defined default port mapping information. Use the **list acl-num** option for the **ip port-map** command to specify an ACL for a host or subnet that uses PAM.



Note

If the host-specific port mapping information is the same as existing system-defined or user-defined default entries, host-specific port changes have no effect.

Examples

The following example provides examples for adding and removing user-defined PAM configuration entries at the firewall.

In the following example, nonstandard port 8000 is established as the user-defined default port for HTTP services:

```
ip port-map http port 8000
```

The following example shows PAM entries that establish a range of nonstandard ports for HTTP services:

```
ip port-map http 8001
ip port-map http 8002
ip port-map http 8003
ip port-map http 8004
```

In the following example the command fails because it tries to map port 21, which is the system-defined default port for FTP, with HTTP:

```
ip port-map http port 21
```

In the following example, a specific host uses port 8000 for FTP services. ACL 10 identifies the server address (192.168.32.43), while port 8000 is mapped with FTP services:

```
access-list 10 permit 192.168.32.43
ip port-map ftp port 8000 list 10
```

In the following example, port 21, which is normally reserved for FTP services, is mapped to the RealAudio application for the hosts in list 10. In this configuration, hosts in list 10 do not recognize FTP activity on port 21.

```
ip port-map realaudio port 21 list 10
```

In the following example, the **ip port-map** command fails and generates an error message:

```
ip port-map netshow port 21
```

```
Command fail: the port 21 has already been defined for ftp by the system.
              No change can be made to the system defined port mappings.
```

In the following example, the **no** form of this command deletes user-defined entries from the PAM table. It has no effect on the system-defined port mappings. This command deletes the host-specific port mapping of FTP.

```
no ip port-map ftp port 1022 list 10
```



Note

All **no** forms of the **ip port-map** command appear before other entries in the running configuration.

In the following example, the command fails because it tries to delete the system-defined default port for HTTP:

```
no ip port-map http port 80
```

In the following example, a specific host uses port 8000 for FTP services. ACL 10 identifies the server address (192.168.32.43), while port 8000 is mapped with FTP services.

```
access-list 10 permit 192.168.32.43
ip port-map ftp port 8000 list 10
```

In the following example, a specific subnet runs HTTP services on port 8080. ACL 50 identifies the subnet, while the PAM entry maps port 8080 with HTTP services.

```
access-list 50 permit 192.168.92.0
ip port-map http 8080 list 50
```

In the following example, a specific host runs HTTP services on port 25, which is the system-defined port number for SMTP services. This requires a host-specific PAM entry that overrides the system-defined default port mapping for HTTP, which is port 80. ACL 15 identifies the host address (192.168.33.43), while port 25 is mapped with HTTP services.

```
access-list 15 permit 192.168.33.43
ip port-map http port 25 list 15
```

In the following example, the same port number is required by different services running on different hosts. Port 8000 is required for HTTP services by host 192.168.3.4, while port 8000 is required for FTP services by host 192.168.5.6. ACL 10 and ACL 20 identify the specific hosts, while PAM maps the ports with the services for each ACL.

```
access-list 10 permit 192.168.3.4
access-list 20 permit 192.168.5.6
ip port-map http port 8000 list 10
ip port-map http ftp 8000 list 20
```

In the following example, five separate port numbers are specified:

```
ip port-map user-my-app port tcp 8085 8087 8092 8093 8094
```

In the following example, multiple commands for the same application name are cumulative and both ports map to the myapp application:

```
ip port-map user-myapp port tcp 3400
ip port-map user-myapp port tcp 3500
```

In the following example, the same port number is assigned to a new application. The new entry replaces the existing entry, meaning that port 5670 gets mapped to user-my-new-app and its mapping to myapp is removed. As a result, the first command no longer appears in the running configuration and you receive a message about the remapping.

```
ip port-map user-myapp port tcp 5670
ip port-map user-my-new-app port tcp 5670
```

In the following example, the second command assigns port 8085 to user-my-new-app because you cannot specify a port number that is in a range assigned to another application. As a result, the first command no longer appears in the running configuration, and you receive a message about the port being moved from one application to another.

```
ip port-map user-my-app port tcp 8085
ip port-map user-my-new-app port tcp from 8080 to 8090
```

Similarly, in the following example the second command assigns port range 8080 to 8085 to user-my-new-app and the first command no longer appears in the running configuration. You receive a message about the remapping.

```
ip port-map user-my-app port tcp from 8080 to 8085
ip port-map user-my-new-app port tcp from 8080 to 8090
```

Related Commands

Command	Description
show ip port-map	Displays the PAM information.

show ip port-map

To display the port-to-application mapping (PAM) information, use the **show ip port-map** command in privileged EXEC mode.

```
show ip port-map [appl-name | port port-num [detail]]
```

Syntax Description		
<i>appl-name</i>	(Optional)	Specifies the name of the application to which to apply the port mapping.
port <i>port-num</i>	(Optional)	Specifies the alternative port number that maps to the application.
detail	(Optional)	Shows the port or application details.

Command Modes	Privileged EXEC
---------------	-----------------

Command History	Release	Modification
	12.0(5)T	This command was introduced.
	12.3(14)T	The detail keyword was added and command output was modified to display user-defined applications.

Usage Guidelines	Use this command to display the port mapping information at the firewall, including the system-defined and user-defined information. Include the application name to display the list of entries by application. Include the port number to display the entries by port.
------------------	--

Examples	The following is sample output from the show ip port-map command, including system- and user-defined mapping information. Notice that multiple port numbers display in a series such as 554, 8554, or 1512...1525, or a range such as 55000 to 62000. When there are multiple ports, they all display if they can fit into the fixed-field width. If they cannot fit into the fixed-field width, they display with an ellipse, such as 1512...1525 shown below.
----------	--

```
Router# show ip port-map

Default mapping: snmp      udp port 161                system defined
Host specific:   snmp      udp port 577                in list 55 user defined
Host specific:   snmp      udp port 55000-62000 in list 57 user defined
Default mapping: echo      tcp port 7                  system defined
Default mapping: echo      udp port 7                  system defined
Default mapping: telnet    tcp port 23                 system defined
Default mapping: wins      tcp port 1512...1525       system defined
Default mapping: n2h2server tcp port 9285              system defined
Default mapping: n2h2server udp port 9285              system defined
Default mapping: nntp      tcp port 119               system defined
Default mapping: pptp      tcp port 1725              system defined
Default mapping: rtsp      tcp port 554,8554          system defined
Default mapping: bootpc    udp port 68                 system defined
Default mapping: gdoi      udp port 848               system defined
Default mapping: tacacs    udp port 49                 system defined
```

```
Default mapping:  gopher      tcp port 70                system defined
Default mapping:  icabrowser  udp port 1604             system defined
```

The following sample output from the **show ip port-map snmp** command displays information about the SNMP application:

```
Router# show ip port-map snmp

Default mapping:  snmp      udp port 161                system defined
Host specific:    snmp      udp port 577                in list 55  user defined
Host specific:    snmp      udp port 55000-62000      in list 57  user defined
```

The following sample output from the **show ip port-map snmp detail** command displays detailed information about the SNMP application:

```
Router# show ip port-map snmp detail

IP port-map entry for application 'snmp':
  udp 161                Simple Network Management Protoco system defined
  udp 577                list 55 User's SNMP Port          user defined
  udp 55000-62000       list 57 User's Another SNMP Port   user defined
```

The following sample output from the **show ip port-map port 577** command displays information about port 577:

```
Router# show ip port-map port 577

Host specific:  snmp  udp port 577  in list 55  user defined
```

The following sample output from the **show ip port-map port 55800** command displays information about port 55800:

```
Router# show ip port-map port 55800

Host specific:  snmp  udp port 55800  in list 57  user defined
```

The following sample output from the **show ip-port-map port 577 detail** command displays detailed information about port 577:

```
Router# show ip port-map port 577 detail

IP Port-map entry for port 577:
  snmp                udp list 55                user defined
```

Related Commands

Command	Description
ip port-map	Establishes PAM entries.

Glossary

CBAC—Context-Based Access Control. A protocol that provides internal users with secure access control for each application and for all traffic across network perimeters. CBAC enhances security by scrutinizing both source and destination addresses and by tracking each application's connection status.

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.



Note

Refer to [Internetworking Terms and Acronyms](#) for terms not included in this glossary.

CCVP, the Cisco logo, and Welcome to the Human Network are trademarks of Cisco Systems, Inc.; Changing the Way We Work, Live, Play, and Learn is a service mark of Cisco Systems, Inc.; and Access Registrar, Aironet, Catalyst, CCDA, CCDP, CCIE, CCIP, CCNA, CCNP, CCSP, Cisco, the Cisco Certified Internetwork Expert logo, Cisco IOS, Cisco Press, Cisco Systems, Cisco Systems Capital, the Cisco Systems logo, Cisco Unity, Enterprise/Solver, EtherChannel, EtherFast, EtherSwitch, Fast Step, Follow Me Browsing, FormShare, GigaDrive, HomeLink, Internet Quotient, IOS, iPhone, IP/TV, iQ Expertise, the iQ logo, iQ Net Readiness Scorecard, iQuick Study, LightStream, Linksys, MeetingPlace, MGX, Networkers, Networking Academy, Network Registrar, PIX, ProConnect, ScriptShare, SMARTnet, StackWise, The Fastest Way to Increase Your Internet Quotient, and TransPath are registered trademarks of Cisco Systems, Inc. and/or its affiliates in the United States and certain other countries.

All other trademarks mentioned in this document or Website are the property of their respective owners. The use of the word partner does not imply a partnership relationship between Cisco and any other company. (0711R)

Copyright © 2005 Cisco Systems, Inc. All rights reserved.

