S Commands

service

Enable system services.

[no] service {resetinbound | resetoutside}

clear service

show service

Syntax Description

resetinbound	Send a reset to a denied inbound TCP packet.
resetoutside	Send a reset to a denied TCP packet to outside interface.

Command Modes

Configuration mode.

Usage Guidelines

The **service** command works with all inbound TCP connections to statics whose access lists or uauth (user authorization) do not allow inbound. One use is for resetting IDENT connections. If an inbound TCP connection is attempted and denied, you can use the **service resetinbound** command to return an RST (reset flag in the TCP header) to the source. Without the option, the PIX Firewall drops the packet without returning an RST.

For use with IDENT, the PIX Firewall sends a TCP RST to the host connecting inbound and stops the incoming IDENT process so that email outbound can be transmitted without having to wait for IDENT to time out. In this case, the PIX Firewall sends a syslog message stating that the incoming connection was a denied connection. Without **service resetinbound**, the PIX Firewall drops packets that are denied and generates a syslog message stating that the SYN was a denied connection. However, outside hosts keep retransmitting the SYN until the IDENT times out.

When an IDENT connection is timing out, you will notice that connections slow down. Perform a trace to determine that IDENT is causing the delay and then invoke the **service** command.

The **service resetinbound** command provides a safer way to handle an IDENT connection through the PIX Firewall. Ranked in order of security from most secure to less secure are these methods for handling IDENT connections:

1. Use the **service resetinbound** command.
2. Use the **established** command with the **permitto tcp 113** options.
3. Enter **static** and **access-list** command statements to open TCP port 113.

When using the **aaa** command, if the first attempt at authorization fails and a second attempt causes a timeout, use the **service resetinbound** command to reset the client that failed the authorization so that it will not retransmit any connections. An example authorization timeout message in Telnet follows:

```
Unable to connect to remote host: Connection timed out
```

Examples

The following example shows use of the **service resetinbound** command:

```
service resetinbound
show service
service resetinbound
```

If you use the **resetoutside** command, the PIX Firewall actively resets denied TCP packets that terminate at the PIX Firewall unit's least-secure interface. By default, these packets are silently discarded. The **resetoutside** option is highly recommended with dynamic or static interface Port Address Translation (PAT). The static interface PAT is available with PIX Firewall Version 6.0 and higher. This option allows the PIX Firewall to quickly terminate the identity request (IDENT) from an external SMTP or FTP server. Actively resetting these connections avoids the thirty-second time-out delay.

If you wish to remove **service** command statements from the configuration, use the **clear service** command.

session enable

The **session enable** command is a deprecated command.

setup

The **setup** command prompts you to enter the information needed to use the Cisco PIX Device Manager (PDM) with a new PIX Firewall.

setup

Syntax Description

setup	Asks for the information needed to start using a new PIX Firewall unit if no configuration is found in the Flash memory.

Command Modes

Configuration mode.

Usage Guidelines

The PIX Firewall requires some pre-configuration before PDM can connect to it. (The setup dialog automatically appears at boot time if there is no configuration in the Flash memory.) Once you enter the **setup** command, you will be asked for the setup information in Table 8-1.

Table 8-1 PIX Firewall Setup Information

Prompt	Description
`Enable password:`	Specify an enable password for this PIX Firewall. (The password must be at least three characters long.)
`Clock (UTC)`	Set the PIX Firewall clock to Universal Coordinated Time (also known as Greenwich Mean Time).
`Year [system year]:`	Specify current year, or default to the year stored in the host computer.
`Month [system month]:`	Specify current month, or default to the month stored in the host computer.
`Day [system day]:`	Specify current day, or default to the day stored in the host computer.
`Time [system time]`	Specify current time in *hh:mm:ss* format, or default to the time stored in the host computer.
`Inside IP address:`	Network interface IP address of the PIX Firewall.
`Inside network mask:`	A network mask that applies to the inside IP address must be a valid mask such as 255.0.0.0, 255.255.0.0, or 255.255.x.x, etc. Use **0.0.0.0** to specify a default route. The **0.0.0.0** netmask can be abbreviated as **0**.
`Host name:`	The host name you want to display in the PIX Firewall command line prompt.
`Domain name:`	The DNS domain name of the network on which the PIX Firewall runs, for example *example.com*.
`IP address of host running PIX Device Manager:`	IP address on which PDM connects to the PIX Firewall.
`Use this configuration and write to flash?`	Store the new configuration to Flash memory. Same as the **write memory** command. If the answer is **yes**, the inside interface will be enabled and the requested configuration will be written to Flash memory. If the user answers anything else, the setup dialog repeats using the values already entered as the defaults for the questions.

The host and domain names are used to generate the default certificate for the SSL connection. The interface type is determined by the hardware.

Examples

The following example shows how to complete the **setup** command prompts.

```
router (config)# setup
Pre-configure PIX Firewall now through interactive prompts [yes]? y
Enable Password [<use current password>]: ciscopix
Clock (UTC)
  Year [2001]: 2001
  Month [Aug]: Sep
  Day [27]: 12
  Time [22:47:37]: <Enter>
Inside IP address: 192.168.1.1
Inside network mask: 255.255.255.0
Host name: accounting_pix
```

```
Domain name: example.com
IP address of host running PIX Device Manager: 192.168.1.2

The following configuration will be used:
Enable Password: ciscopix
Clock (UTC): 22:47:37 Sep 12 2001
Inside IP address: ...192.168.1.1
Inside network mask: ...255.255.255.0
Host name: ...accounting_pix
Domain name: ...example.com
IP address of host running PIX Device Manager: ...192.168.1.2

Use this configuration and write to flash? y
```

Related Commands

pdm	Configures PIX Device Manager (PDM).

show

View command information.

show *command_keywords* [| {**include** | **exclude** | **begin** | **grep** [**-v**]} *regexp*]

show ?

Syntax Description

command_key words	Any argument or list of arguments that specifies the information to display. Most commands have a **show** command form where the command name is used as **show** argument. For example, the **global** command has an associated **show global** command.
|	The UNIX pipe symbol, "|". This character represents piping output to the filter. When "|" is present, a filtering option and a regular expression must also be present. (Only the first "|" is a pipe character in the syntax.)
include	Includes all output lines that match the specified regular expression.
exclude	Excludes all output lines that match the specified regular expression.
grep	Displays all output lines that match the specified regular expression. **grep** is equivalent to **include** and **grep -v** is equivalent to **exclude**.
begin	Displays all output lines starting from the line that matches the specified regular expression.
regexp	A Cisco IOS software style regular expression. Do not enclose in quotes or double-quotes. Additionally, trailing white spaces (between keywords) are taken as part of the regular expression.

Command Modes

All modes.

Usage Guidelines

The **show** *command_keywords* [| {**include | exclude | begin | grep**} *regexp*] command runs the show command options specified. See individual commands for their show options. (Only the first "|" is a pipe character in this syntax.) The CLI syntax and semantics of the **show** output filtering options are the same as in Cisco IOS software, and are available through console, Telnet, or SSH sessions.

The **show ?** command displays a list of all commands available on the PIX Firewall.

Explanations for the specific **show** commands are documented with the corresponding command. For example, the **show arp** command description is included with the **arp** command.

Examples

The following example illustrates how to use a show command output filter option, where the "|" is the UNIX pipe symbol:

```
pixfirewall(config)# show config | grep access-list
access-list 101 permit tcp any host 10.1.1.3 eq www
access-list 101 permit tcp any host 10.1.1.3 eq smtp
```

The following is sample output from the **show ?** command:

```
pixfirewall(config)# show ?

At the end of show <command>, use the pipe character '|' followed by:
begin|include|exclude|grep [-v] <regular_exp>, to filter show output.

aaa             Enable, disable, or view TACACS+, RADIUS or LOCAL
                user authentication, authorization and accounting

aaa-server      Define AAA Server group
access-group    Bind an access-list to an interface to filter inbound traffic
access-list     Add an access list
activation-key  Modify activation-key.
age             This command is deprecated. See ipsec, isakmp, map, ca commands
alias           Administer overlapping addresses with dual NAT.
apply           Apply outbound lists to source or destination IP addresses
arp             Change or view arp table, set arp timeout value and view statiss
auth-prompt     Customize authentication challenge, reject or acceptance prompt
auto-update     Configure auto update support
banner          Configure login/session banners
blocks          Show system buffer utilization
ca              CEP (Certificate Enrollment Protocol)
                Create and enroll RSA key pairs into a PKI (Public Key Infrastr.
capture         Capture inbound and outbound packets on one or more interfaces
checksum        View configuration information cryptochecksum
chunkstat       Display chunk stats
clock           Show and set the date and time of PIX
conduit         Add conduit access to higher security level network or ICMP
configure       Configure from terminal, floppy, memory, network, or
                factory-default.  The configuration will be merged with the
                active configuration except for factory-default in which case
                the active configuration is cleared first.
conn            Display connection information
console         Set idle timeout for the serial console of the PIX
cpu             Display cpu usage
Crashinfo       Read, write and configure crash write to flash.
crypto          Configure IPsec, IKE, and CA
ctiqbe          Show the current data stored for each CTIQBE session.
curpriv         Display current privilege level
debug           Debug packets or ICMP tracings through the PIX Firewall.
dhcpd           Configure DHCP Server
dhcprelay       Configure DHCP Relay Agent
domain-name     Change domain name
```

```
dynamic-map     Specify a dynamic crypto map template
eeprom          show or reprogram the 525 onboard i82559 devices
enable          Configure enable passwords
established     Allow inbound connections based on established connections
failover        Enable/disable PIX failover feature to a standby PIX
filter          Enable, disable, or view URL, FTP, HTTPS, Java, and ActiveX filg
fips-mode       Enable or disable FIPS mode
fixup           Add or delete PIX service and feature defaults
flashfs         Show, destroy, or preserve filesystem information
fragment        Configure the IP fragment database
global          Specify, delete or view global address pools,
                or designate a PAT(Port Address Translated) address
h225            Show the current h225 data stored for each connection.
h245            List the h245 connections.
h323-ras        Show the current h323 ras data stored for each connection.
history         Display the session command history
http            Configure HTTP server
icmp            Configure access for ICMP traffic that terminates at an interface
interface       Set network interface parameters and configure VLANs
igmp            Clear or display IGMP groups
ip              Set the ip address and mask for an interface
                Define a local address pool
                Configure Unicast RPF on an interface
                Configure the Intrusion Detection System
ipsec           Configure IPSEC policy
isakmp          Configure ISAKMP policy
isakmp log     Clear events in the isakmp log buffer
local-host      Display or clear the local host network information
logging         Enable logging facility
mac-list        Add a list of mac addresses using first match search
map             Configure IPsec crypto map
memory          System memory utilization
mgcp            Configure the Media Gateway Control Protocol fixup
mroute          Configure a multicast route
mtu             Specify MTU(Maximum Transmission Unit) for an interface
multicast       Configure multicast on an interface
name            Associate a name with an IP address
nameif          Assign a name to an interface
names           Enable, disable or display IP address to name conversion
nat             Associate a network with a pool of global IP addresses
ntp             Configure Network Time Protocol
object-group    Create an object group for use in 'access-list', 'conduit', etc
ospf            Show OSPF information or clear ospf items.
outbound        Create an outbound access list
pager           Control page length for pagination
passwd          Change Telnet console access password
pdm             Configure Pix Device Manager
prefix-list     Configure a prefix-list
privilege       Configure/Display privilege levels for commands
processes       Display processes
rip             Broadcast default route or passive RIP
route           Enter a static route for an interface
route-map       Create a route-map.
router          Create/configure OSPF routing process
routing         Configure interface specific unicast routing parameters.
running-config  Display the current running configuration
service         Enable system services
session         Access an internal AccessPro router console
shun            Manages the filtering of packets from undesired hosts
sip             Show the current data stored for each SIP session.
skinny          Show the current data stored for each Skinny session.
snmp-server     Provide SNMP and event information
ssh             Add SSH access to PIX console, set idle timeout, display
                list of active SSH sessions & terminate a SSH session
```

```
startup-config  Display the startup configuration
static          Configure one-to-one address translation rule
sysopt          Set system functional option
tcpstat         Display status of tcp stack and tcp connections
tech-support    Tech support
telnet          Add telnet access to PIX console and set idle timeout
terminal        Set terminal line parameters
tftp-server     Specify default TFTP server address and directory
timeout         Set the maximum idle times
traffic         Counters for traffic statistics
uauth           Display or clear current user authorization information
url-cache       Enable URL caching
url-block       Enable URL pending block buffer and long URL support
url-server      Specify a URL filter server
username        Configure user authentication local database
version         Display PIX system software version
virtual         Set address for authentication virtual servers
vpdn            Configure VPDN (PPTP, L2TP, PPPoE) Policy
vpnclient       Configure Easy VPN Remote
vpngroup        Configure group settings for Cisco VPN Clients and
                Cisco Easy VPN Remote products
who             Show active administration sessions on PIX
xlate           Display current translation and connection slot information
```

show blocks/clear blocks

Show system buffer utilization.

show blocks

clear blocks

Syntax Description

blocks	The blocks in the preallocated system buffer.

Command Modes

Privileged mode.

Usage Guidelines

The **show blocks** command lists preallocated system buffer utilization. In the **show blocks** command listing, the SIZE column displays the block type. The MAX column is the maximum number of allocated blocks. The LOW column is the fewest blocks available since last reboot. The CNT column is the current number of available blocks. A zero in the LOW column indicates a previous event where memory exhausted. A zero in the CNT column means memory is exhausted now. Exhausted memory is not a problem as long as traffic is moving through the PIX Firewall. You can use the **show conn** command to see if traffic is moving. If traffic is not moving and the memory is exhausted, a problem may be indicated.

The **clear blocks** command keeps the maximum count to whatever number is allocated in the system and equates the low count to the current count.

You can also view the information from the **show blocks** command using SNMP.

Examples

The following is sample output from the **show blocks** command:

```
show blocks
  SIZE    MAX    LOW    CNT
     4   1600   1600   1600
    80    100     97     97
   256     80     79     79
  1550    788    402    404
 65536      8      8      8
```

show checksum

Display the configuration checksum.

show checksum

Syntax Description

checksum	The hexadecimal numbers that act as a digital summary of the contents of the configuration.

Command Modes

Unprivileged mode.

Usage Guidelines

The **show checksum** command displays four groups of hexadecimal numbers that act as a digital summary of the contents of the configuration.This checksum is calculated only when you store the configuration in Flash memory. By using the **show config** command and viewing the checksum at the end of the configuration listing and using the **show checksum** command, you can compare the numbers to see if the configuration has changed. The PIX Firewall tests the checksum to determine if a configuration has not been corrupted.

If a dot (".") appears before the checksum in the **show config** or **show checksum** command output, this is a normal configuration load or write mode indicator (when loading from or writing to the firewall Flash memory). This "." is provided to show that the firewall is preoccupied with the operation but not "hung up". It is analogous to a "system processing, please wait" message.

Examples

The following is sample output from the **show checksum** command:

```
show checksum
Cryptochecksum: 1a2833c0 129ac70b 1a88df85 650dbb81
```

show chunkstat

Displays information about management of memory chunks.

show chunkstat

Syntax Description

chunkstat	Displays internal information about management of memory chunks.

Command Modes

Unprivileged mode.

Usage Guidelines

The command **show chunkstat** displays summary information about chunk management, followed by a dump showing the address, content, links, flags and other details.

Examples

The following is sample output from the **show chunkstat** command:

```
show chunkstat
Result of firewall command: "show chunkstat"

Chunk statistics: created 1, destroyed: 0,sibs created: 0, sibs trimmed: 0
Dump of chunk at 0100e09c, name "OSPF redist route node chunks", data start @ 0100e504,
end @ 01010904
 flink: 01008f6c, blink: 01008f6c
 next: cccccccc, next_sibling: 00000000, prev_sibling: 00000000
 flags 00000005
 maximum chunk elt's: 256, elt size: 36, index first free 256
 # chunks in use: 0, HWM of total used: 1, alignment: 8
Chunk statistics: created 1, destroyed: 0,sibs created: 0, sibs trimmed: 0
Dump of chunk at 00eb4cbc, name "ulimit chunk", data start @ 00eb4d8c, end @ 00eb4f8c
 flink: 00578910, blink: 00578910
 next: cccccccc, next_sibling: 00000000, prev_sibling: 00000000
 flags 00000001
 maximum chunk elt's: 32, elt size: 16, index first free 32
 # chunks in use: 0, HWM of total used: 0, alignment: 0
Chunk statistics: created 1, destroyed: 0,sibs created: 0, sibs trimmed: 0
Dump of chunk at 00ea0cd4, name "uauth chunk", data start @ 00ea0da4, end @ 00eb4ca4
 flink: 005793a0, blink: 005793a0
 next: cccccccc, next_sibling: 00000000, prev_sibling: 00000000
 flags 00000001
 maximum chunk elt's: 32, elt size: 2552, index first free 32
 # chunks in use: 0, HWM of total used: 1, alignment: 0
Chunk statistics: created 1, destroyed: 0,sibs created: 0, sibs trimmed: 0
Dump of chunk at 00df706c, name "IP subnet NDB entry", data start @ 00df788c, end @
00e8522c
 flink: 00527914, blink: 00527914
 next: cccccccc, next_sibling: 00000000, prev_sibling: 00000000
 flags 00000009
 maximum chunk elt's: 500, elt size: 1156, index first free 498
 # chunks in use: 2, HWM of total used: 2, alignment: 0
Chunk statistics: created 1, destroyed: 0,sibs created: 0, sibs trimmed: 0
Dump of chunk at 00de56c4, name "IP single NDB entry", data start @ 00de5ee4, end @
00df7054
 flink: 005278f4, blink: 005278f4
 next: cccccccc, next_sibling: 00000000, prev_sibling: 00000000
 flags 00000009
 maximum chunk elt's: 500, elt size: 136, index first free 497
 # chunks in use: 3, HWM of total used: 3, alignment: 0
Chunk statistics: created 1, destroyed: 0,sibs created: 0, sibs trimmed: 0
Dump of chunk at 00dd9f34, name "mroute chunk", data start @ 00dd9fa4, end @ 00dda064
 flink: 0056bf10, blink: 0056bf10
 next: cccccccc, next_sibling: 00000000, prev_sibling: 00000000
 flags 00000001
 maximum chunk elt's: 8, elt size: 24, index first free 8
 # chunks in use: 0, HWM of total used: 0, alignment: 0
Chunk statistics: created 1, destroyed: 0,sibs created: 0, sibs trimmed: 0
Dump of chunk at 00dd76cc, name "radix trie", data start @ 00dd7b1c, end @ 00dd9f1c
 flink: 00dd7684, blink: 00dd7684
 next: cccccccc, next_sibling: 00000000, prev_sibling: 00000000
```

show conn

Display all active connections.

show conn [**count**] | [**detail**] | [**protocol tcp** | **udp** | *protocol*] [{**foreign** | **local**} *ip* [*-ip2*]] [**netmask** *mask*]] [{**lport** | **fport**} *port1* [*-port2*]]

show conn state [**up**] [**,conn_inbound**][**,ctiqbe**][**,data_in**][**,data_out**][**,dump**][**,finin**] [**,finout**][**,h225**][**,h323**][**,http_get**][**,mgcp**][**,nojava**][**,rpc**][**,sip**][**,skinny**][**,smtp_data**] [**,smtp_banner**] [**,sqlnet_fixup_data**][**,smtp_incomplete**]

Syntax Description

count	Display only the number of used connections. The precision of the displayed count may vary depending on traffic volume and the type of traffic passing through the PIX Firewall unit.
detail	If specified, displays translation type and interface information.
{**foreign** | **local**} *ip* [**-***ip2*] **netmask** *mask*	Display active connections by the foreign IP address or by local IP address. Qualify foreign or local active connections by network mask.
fixup	Display whether or not RTP traffic is flowing through the PIX Firewall.
{**lport** | **fport**} *port1* [**-***port2*]	Display foreign or local active connections by port. See "Ports" in Chapter 2, "Using PIX Firewall Commands" for a list of valid port literal names.
protocol tcp | **udp** | *protocol*	Display active connections by protocol type. *protocol* is a protocol specified by number. See "Protocols" in Chapter 2, "Using PIX Firewall Commands" for a list of valid protocol literal names.
state	Display active connections by their current state: up (**up**), inbound connection (**conn_inbound**), Computer Telephony Interface Quick Buffer Encoding (CTIQBE) connection (**ctiqbe**), inbound data (**data_in**), outbound data (**data_out**), dump clean up connection (**dump**), FIN inbound (**finin**), FIN outbound (**finout**), H.225 connection (**h225**), H.323 connection (**h323**), HTTP get (**http_get**), Media Gateway Control Protocol (MGCP) connection (**mgcp**), an **outbound** command denying access to Java applets (**nojava**), RPC connection (**rpc**), SIP connection (**sip**), Skinny Client Control Protocol (SCCP) connection (**skinny**), SMTP mail banner (**smtp_banner**), SMTP mail data (**smtp_data**), SQL*Net data fix up (**sqlnet_fixup_data**), and incomplete SMTP mail connection (**smtp_incomplete**).

Command Modes

Privileged mode.

Usage Guidelines

The **show conn** command displays the number of, and information about, active TCP connections. When specifying multiple **show conn state** options, use commas without spaces to the list states to be displayed. For example, the following is correct:

```
pixfirewall(config)# show conn state up,rpc,h323,sip
```

If you insert spaces, the firewall will not recognize the command.

You can also view the connection count information from the **show conn** command using SNMP.

Note For connections using a DNS server, the source port of the connection may be replaced by the *IP address of DNS server* in the **show conn** output.

The **show conn detail** command displays the following information:

{**UDP** | **TCP**} *outside_ifc*:*real_addr*/*real-port* [(*map_addr*/*port*)] *inside_ifc*:*real_addr*/*real_port* [(*map-addr*/*port*)] **flags** *flags*

The connection flags are defined in Table 8-2.

Table 8-2 Connection Flags

Flag	Description
a	awaiting outside ACK to SYN
A	awaiting inside ACK to SYN
B	initial SYN from outside
C	Computer Telephony Interface Quick Buffer Encoding (CTIQBE) media connection
d	dump
D	DNS
E	outside back connection
f	inside FIN
F	outside FIN
g	Media Gateway Control Protocol (MGCP) connection
G	connection is part of a group[1]
h	H.225
H	H.323
i	incomplete TCP or UDP connection
I	inbound data
k	Skinny Client Control Protocol (SCCP) media connection
m	SIP media connection
M	SMTP data
O	outbound data
p	replicated (unused)
P	inside back connection
q	SQL*Net data
r	inside acknowledged FIN
R	outside acknowledged FIN for TCP connection
R	UDP RPC[2]
s	awaiting outside SYN
S	awaiting inside SYN

Table 8-2 Connection Flags (continued)

Flag	Description
t	SIP transient connection[3]
T	SIP connection[4]
U	up

1. The G flag indicates the connection is part of a group. It is set by the GRE and FTP Strict fixups to designate the control connection and all its associated secondary connections. If the control connection terminates, then all associated secondary connections are also terminated.
2. Because each row of **show conn** command output represents one connection (TCP or UDP), there will be only one R flag per row.
3. For UDP connections, the value t indicates that it will timeout after one minute.
4. For UDP connections, the value T indicates that the connection will timeout according to the value specified using the **timeout sip** command.

Examples

The following example shows a TCP session connection from inside host 10.1.1.15 to the outside Telnet server at 192.150.49.10. Because there is no B flag, the connection is initiated from the inside. The “U”, “I”, and “O” flags denote that the connection is active and has received inbound and outbound data.

```
pixfirewall(config)# show conn
2 in use, 2 most used
TCP out 192.150.49.10:23 in 10.1.1.15:1026 idle 0:00:22
Bytes 1774 flags UIO
UDP out 192.150.49.10:31649 in 10.1.1.15:1028 idle 0:00:14
flags D-
```

The following example shows a UDP connection from outside host 192.150.49.10 to inside host 10.1.1.15. The D flag denotes that this is a DNS connection. The number 1028 is the DNS ID over the connection.

```
pixfirewall(config)# show conn detail
2 in use, 2 most used
Flags: A - awaiting inside ACK to SYN, a - awaiting outside ACK to SYN,
       B - initial SYN from outside, C - CTIBQE media, D - DNS, d - dump,
       E - outside back connection, f - inside FIN, F - outside FIN,
       G - group, g - MGCP, H - H.323, h - H.255.0, I - inbound data, i - incomplete,
        k - Skinny media, M - SMTP data, m - SIP media
       O - outbound data, P - inside back connection,
       q - SQL*Net data, R - outside acknowledged FIN,
       R - UDP RPC, r - inside acknowledged FIN, S - awaiting inside SYN,
        s - awaiting outside SYN, T - SIP, t - SIP transient, U - up
TCP outside:192.150.49.10/23 inside:10.1.1.15/1026 flags UIO
UDP outside:192.150.49.10/31649 inside:10.1.1.15/1028 flags dD
```

The following is sample output from the **show conn** command:

```
show conn
6 in use, 6 most used
  TCP out 209.165.201.1:80 in 10.3.3.4:1404 idle 0:00:00 Bytes 11391
  TCP out 209.165.201.1:80 in 10.3.3.4:1405 idle 0:00:00 Bytes 3709
  TCP out 209.165.201.1:80 in 10.3.3.4:1406 idle 0:00:01 Bytes 2685
  TCP out 209.165.201.1:80 in 10.3.3.4:1407 idle 0:00:01 Bytes 2683
  TCP out 209.165.201.1:80 in 10.3.3.4:1403 idle 0:00:00 Bytes 15199
  TCP out 209.165.201.1:80 in 10.3.3.4:1408 idle 0:00:00 Bytes 2688
  UDP out 209.165.201.7:24 in 10.3.3.4:1402 idle 0:01:30
  UDP out 209.165.201.7:23 in 10.3.3.4:1397 idle 0:01:30
  UDP out 209.165.201.7:22 in 10.3.3.4:1395 idle 0:01:30
```

In this example, host 10.3.3.4 on the inside has accessed a website at 209.165.201.1. The global address on the outside interface is 209.165.201.7.

show cpu usage

The **show cpu usage** command displays CPU utilization.

show cpu usage

Syntax Description

cpu usage	The central processing unit (CPU) usage data.

Command Modes

Privileged or configuration mode.

Usage Guidelines

The **show cpu usage** command displays the central processing unit (CPU) usage information.

Examples

The following is sample output from the **show cpu usage** command:

```
pixfirewall# show cpu usage
CPU utilization for 5 seconds: p1%; 1 minute: p2%; 5 minutes: p3%
```

The percentage usage prints as NA (not applicable) if the usage is unavailable for the specified time interval. This can happen if the user asks for CPU usage before the 5-second, 1-minute, or 5-minute time interval has elapsed.

show crashinfo

To display the contents of the crash file stored in Flash memory, enter the **show crashinfo** command in privileged EXEC mode.

show crashinfo [**save**]

Syntax Description

save	(Optional) Displays if the security appliance is configured to save crash information to Flash memory or not.

Defaults

No default behavior or values.

Command Modes

The following table shows the modes in which you can enter the command:

	Firewall Mode		Security Context		
				Multiple	
Command Mode	**Routed**	**Transparent**	**Single**	**Context**	**System**
Privileged EXEC	•	•	•	—	•

Command History

Release	Modification
Preexisting	This command was preexisting.

Usage Guidelines

If the crash file is from a test crash (generated from the **crashinfo test** command), the first string of the crash file is "**: Saved_Test_Crash**" and the last string is "**: End_Test_Crash**". If the crash file is from a real crash, the first string of the crash file is "**: Saved_Crash**" and the last string is "**: End_Crash**". (This includes crashes from use of the **crashinfo force page-fault** or **crashinfo force watchdog** commands).

If there is no crash data saved in flash, or if the crash data has been cleared by entering the **clear crashinfo** command, the **show crashinfo** command displays an error message.

Examples

The following example shows how to display the current crash information configuration:

```
hostname# show crashinfo save
crashinfo save enable
```

The following example shows the output for a crash file test. (However, this test does not actually crash the security appliance. It provides a simulated example file.)

```
hostname(config)# crashinfo test
hostname(config)# exit
hostname# show crashinfo
: Saved_Test_Crash

Thread Name: ci/console (Old pc 0x001a6ff5 ebp 0x00e88920)
```

```
Traceback:
0: 00323143
1: 0032321b
2: 0010885c
3: 0010763c
4: 001078db
5: 00103585
6: 00000000
    vector 0x000000ff (user defined)
       edi 0x004f20c4
       esi 0x00000000
       ebp 0x00e88c20
       esp 0x00e88bd8
       ebx 0x00000001
       edx 0x00000074
       ecx 0x00322f8b
       eax 0x00322f8b
error code n/a
       eip 0x0010318c
        cs 0x00000008
    eflags 0x00000000
       CR2 0x00000000
Stack dump: base:0x00e8511c size:16384, active:1476
0x00e89118: 0x004f1bb4
0x00e89114: 0x001078b4
0x00e89110-0x00e8910c: 0x00000000
0x00e89108-0x00e890ec: 0x12345678
0x00e890e8: 0x004f1bb4
0x00e890e4: 0x00103585
0x00e890e0: 0x00e8910c
0x00e890dc-0x00e890cc: 0x12345678
0x00e890c8: 0x00000000
0x00e890c4-0x00e890bc: 0x12345678
0x00e890b8: 0x004f1bb4
0x00e890b4: 0x001078db
0x00e890b0: 0x00e890e0
0x00e890ac-0x00e890a8: 0x12345678
0x00e890a4: 0x001179b3
0x00e890a0: 0x00e890b0
0x00e8909c-0x00e89064: 0x12345678
0x00e89060: 0x12345600
0x00e8905c: 0x20232970
0x00e89058: 0x616d2d65
0x00e89054: 0x74002023
0x00e89050: 0x29676966
0x00e8904c: 0x6e6f6328
0x00e89048: 0x31636573
0x00e89044: 0x7069636f
0x00e89040: 0x64786970
0x00e8903c-0x00e88e50: 0x00000000
0x00e88e4c: 0x000a7473
0x00e88e48: 0x6574206f
0x00e88e44: 0x666e6968
0x00e88e40: 0x73617263
0x00e88e3c-0x00e88e38: 0x00000000
0x00e88e34: 0x12345600
0x00e88e30-0x00e88dfc: 0x00000000
0x00e88df8: 0x00316761
0x00e88df4: 0x74706100
0x00e88df0: 0x12345600
0x00e88dec-0x00e88ddc: 0x00000000
0x00e88dd8: 0x00000070
0x00e88dd4: 0x616d2d65
```

```
0x00e88dd0: 0x74756f00
0x00e88dcc: 0x00000000
0x00e88dc8: 0x00e88e40
0x00e88dc4: 0x004f20c4
0x00e88dc0: 0x12345600
0x00e88dbc: 0x00000000
0x00e88db8: 0x00000035
0x00e88db4: 0x315f656c
0x00e88db0: 0x62616e65
0x00e88dac: 0x0030fcf0
0x00e88da8: 0x3011111f
0x00e88da4: 0x004df43c
0x00e88da0: 0x0053fef0
0x00e88d9c: 0x004f1bb4
0x00e88d98: 0x12345600
0x00e88d94: 0x00000000
0x00e88d90: 0x00000035
0x00e88d8c: 0x315f656c
0x00e88d88: 0x62616e65
0x00e88d84: 0x00000000
0x00e88d80: 0x004f20c4
0x00e88d7c: 0x00000001
0x00e88d78: 0x01345678
0x00e88d74: 0x00f53854
0x00e88d70: 0x00f7f754
0x00e88d6c: 0x00e88db0
0x00e88d68: 0x00e88d7b
0x00e88d64: 0x00f53874
0x00e88d60: 0x00e89040
0x00e88d5c-0x00e88d54: 0x12345678
0x00e88d50-0x00e88d4c: 0x00000000
0x00e88d48: 0x004f1bb4
0x00e88d44: 0x00e88d7c
0x00e88d40: 0x00e88e40
0x00e88d3c: 0x00f53874
0x00e88d38: 0x004f1bb4
0x00e88d34: 0x0010763c
0x00e88d30: 0x00e890b0
0x00e88d2c: 0x00e88db0
0x00e88d28: 0x00e88d88
0x00e88d24: 0x0010761a
0x00e88d20: 0x00e890b0
0x00e88d1c: 0x00e88e40
0x00e88d18: 0x00f53874
0x00e88d14: 0x0010166d
0x00e88d10: 0x0000000e
0x00e88d0c: 0x00f53874
0x00e88d08: 0x00f53854
0x00e88d04: 0x0048b301
0x00e88d00: 0x00e88d30
0x00e88cfc: 0x0000000e
0x00e88cf8: 0x00f53854
0x00e88cf4: 0x0048a401
0x00e88cf0: 0x00f53854
0x00e88cec: 0x00f53874
0x00e88ce8: 0x0000000e
0x00e88ce4: 0x0048a64b
0x00e88ce0: 0x0000000e
0x00e88cdc: 0x00f53874
0x00e88cd8: 0x00f7f96c
0x00e88cd4: 0x0048b4f8
0x00e88cd0: 0x00e88d00
0x00e88ccc: 0x0000000f
0x00e88cc8: 0x00f7f96c
```

```
0x00e88cc4-0x00e88cc0: 0x0000000e
0x00e88cbc: 0x00e89040
0x00e88cb8: 0x00000000
0x00e88cb4: 0x00f5387e
0x00e88cb0: 0x00f53874
0x00e88cac: 0x00000002
0x00e88ca8: 0x00000001
0x00e88ca4: 0x00000009
0x00e88ca0-0x00e88c9c: 0x00000001
0x00e88c98: 0x00e88cb0
0x00e88c94: 0x004f20c4
0x00e88c90: 0x0000003a
0x00e88c8c: 0x00000000
0x00e88c88: 0x0000000a
0x00e88c84: 0x00489f3a
0x00e88c80: 0x00e88d88
0x00e88c7c: 0x00e88e40
0x00e88c78: 0x00e88d7c
0x00e88c74: 0x001087ed
0x00e88c70: 0x00000001
0x00e88c6c: 0x00e88cb0
0x00e88c68: 0x00000002
0x00e88c64: 0x0010885c
0x00e88c60: 0x00e88d30
0x00e88c5c: 0x00727334
0x00e88c58: 0xa0ffffff
0x00e88c54: 0x00e88cb0
0x00e88c50: 0x00000001
0x00e88c4c: 0x00e88cb0
0x00e88c48: 0x00000002
0x00e88c44: 0x0032321b
0x00e88c40: 0x00e88c60
0x00e88c3c: 0x00e88c7f
0x00e88c38: 0x00e88c5c
0x00e88c34: 0x004b1ad5
0x00e88c30: 0x00e88c60
0x00e88c2c: 0x00e88e40
0x00e88c28: 0xa0ffffff
0x00e88c24: 0x00323143
0x00e88c20: 0x00e88c40
0x00e88c1c: 0x00000000
0x00e88c18: 0x00000008
0x00e88c14: 0x0010318c
0x00e88c10-0x00e88c0c: 0x00322f8b
0x00e88c08: 0x00000074
0x00e88c04: 0x00000001
0x00e88c00: 0x00e88bd8
0x00e88bfc: 0x00e88c20
0x00e88bf8: 0x00000000
0x00e88bf4: 0x004f20c4
0x00e88bf0: 0x000000ff
0x00e88bec: 0x00322f87
0x00e88be8: 0x00f5387e
0x00e88be4: 0x00323021
0x00e88be0: 0x00e88c10
0x00e88bdc: 0x004f20c4
0x00e88bd8: 0x00000000 *
0x00e88bd4: 0x004eabb0
0x00e88bd0: 0x00000001
0x00e88bcc: 0x00f5387e
0x00e88bc8-0x00e88bc4: 0x00000000
0x00e88bc0: 0x00000008
0x00e88bbc: 0x0010318c
0x00e88bb8-0x00e88bb4: 0x00322f8b
```

```
0x00e88bb0: 0x00000074
0x00e88bac: 0x00000001
0x00e88ba8: 0x00e88bd8
0x00e88ba4: 0x00e88c20
0x00e88ba0: 0x00000000
0x00e88b9c: 0x004f20c4
0x00e88b98: 0x000000ff
0x00e88b94: 0x001031f2
0x00e88b90: 0x00e88c20
0x00e88b8c: 0xffffffff
0x00e88b88: 0x00e88cb0
0x00e88b84: 0x00320032
0x00e88b80: 0x37303133
0x00e88b7c: 0x312f6574
0x00e88b78: 0x6972772f
0x00e88b74: 0x342f7665
0x00e88b70: 0x64736666
0x00e88b6c: 0x00020000
0x00e88b68: 0x00000010
0x00e88b64: 0x00000001
0x00e88b60: 0x123456cd
0x00e88b5c: 0x00000000
0x00e88b58: 0x00000008

Cisco XXX Firewall Version X.X
Cisco XXX Device Manager Version X.X

Compiled on Fri 15-Nov-04 14:35 by root

hostname up 10 days 0 hours

Hardware:   XXX-XXX, 64 MB RAM, CPU Pentium 200 MHz
Flash i28F640J5 @ 0x300, 16MB
BIOS Flash AT29C257 @ 0xfffd8000, 32KB

0: ethernet0: address is 0003.e300.73fd, irq 10
1: ethernet1: address is 0003.e300.73fe, irq 7
2: ethernet2: address is 00d0.b7c8.139e, irq 9
Licensed Features:
Failover:           Disabled
VPN-DES:            Enabled
VPN-3DES-AES:       Disabled
Maximum Interfaces: 3
Cut-through Proxy:  Enabled
Guards:             Enabled
URL-filtering:      Enabled
Inside Hosts:       Unlimited
Throughput:         Unlimited
IKE peers:          Unlimited

This XXX has a Restricted (R) license.

Serial Number: 480430455 (0x1ca2c977)
Running Activation Key: 0xc2e94182 0xc21d8206 0x15353200 0x633f6734
Configuration last modified by enable_15 at 13:49:42.148 UTC Wed Nov 20 2004

------------------ show clock ------------------

15:34:28.129 UTC Sun Nov 24 2004

------------------ show memory ------------------

Free memory:        50444824 bytes
Used memory:        16664040 bytes
```

```
-------------     ----------------
Total memory:       67108864 bytes

------------------ show conn count ------------------

0 in use, 0 most used

------------------ show xlate count ------------------

0 in use, 0 most used

------------------ show blocks ------------------

  SIZE    MAX    LOW    CNT
     4   1600   1600   1600
    80    400    400    400
   256    500    499    500
  1550   1188    795    927

------------------ show interface ------------------

interface ethernet0 "outside" is up, line protocol is up
  Hardware is i82559 ethernet, address is 0003.e300.73fd
  IP address 172.23.59.232, subnet mask 255.255.0.0
  MTU 1500 bytes, BW 10000 Kbit half duplex
        6139 packets input, 830375 bytes, 0 no buffer
        Received 5990 broadcasts, 0 runts, 0 giants
        0 input errors, 0 CRC, 0 frame, 0 overrun, 0 ignored, 0 abort
        90 packets output, 6160 bytes, 0 underruns
        0 output errors, 13 collisions, 0 interface resets
        0 babbles, 0 late collisions, 47 deferred
        0 lost carrier, 0 no carrier
        input queue (curr/max blocks): hardware (5/128) software (0/2)
        output queue (curr/max blocks): hardware (0/1) software (0/1)
interface ethernet1 "inside" is up, line protocol is down
  Hardware is i82559 ethernet, address is 0003.e300.73fe
  IP address 10.1.1.1, subnet mask 255.255.255.0
  MTU 1500 bytes, BW 10000 Kbit half duplex
        0 packets input, 0 bytes, 0 no buffer
        Received 0 broadcasts, 0 runts, 0 giants
        0 input errors, 0 CRC, 0 frame, 0 overrun, 0 ignored, 0 abort
        1 packets output, 60 bytes, 0 underruns
        0 output errors, 0 collisions, 0 interface resets
        0 babbles, 0 late collisions, 0 deferred
        1 lost carrier, 0 no carrier
        input queue (curr/max blocks): hardware (128/128) software (0/0)
        output queue (curr/max blocks): hardware (0/1) software (0/1)
interface ethernet2 "intf2" is administratively down, line protocol is down
  Hardware is i82559 ethernet, address is 00d0.b7c8.139e
  IP address 127.0.0.1, subnet mask 255.255.255.255
  MTU 1500 bytes, BW 10000 Kbit half duplex
        0 packets input, 0 bytes, 0 no buffer
        Received 0 broadcasts, 0 runts, 0 giants
        0 input errors, 0 CRC, 0 frame, 0 overrun, 0 ignored, 0 abort
        0 packets output, 0 bytes, 0 underruns
        0 output errors, 0 collisions, 0 interface resets
        0 babbles, 0 late collisions, 0 deferred
        0 lost carrier, 0 no carrier
        input queue (curr/max blocks): hardware (128/128) software (0/0)
        output queue (curr/max blocks): hardware (0/0) software (0/0)

------------------ show cpu usage ------------------

CPU utilization for 5 seconds = 0%; 1 minute: 0%; 5 minutes: 0%
```

```
------------------ show process ------------------

     PC       SP       STATE       Runtime    SBASE     Stack Process
Hsi 001e3329 00763e7c 0053e5c8           0 00762ef4 3784/4096 arp_timer
Lsi 001e80e9 00807074 0053e5c8           0 008060fc 3792/4096 FragDBGC
Lwe 00117e3a 009dc2e4 00541d18           0 009db46c 3704/4096 dbgtrace
Lwe 003cee95 009de464 00537718           0 009dc51c 8008/8192 Logger
Hwe 003d2d18 009e155c 005379c8           0 009df5e4 8008/8192 tcp_fast
Hwe 003d2c91 009e360c 005379c8           0 009e1694 8008/8192 tcp_slow
Lsi 002ec97d 00b1a464 0053e5c8           0 00b194dc 3928/4096 xlate clean
Lsi 002ec88b 00b1b504 0053e5c8           0 00b1a58c 3888/4096 uxlate clean
Mrd 002e3a17 00c8f8d4 0053e600           0 00c8d93c 7908/8192 tcp_intercept_times
Lsi 00423dd5 00d3a22c 0053e5c8           0 00d392a4 3900/4096 route_process
Hsi 002d59fc 00d3b2bc 0053e5c8           0 00d3a354 3780/4096 PIX Garbage Collecr
Hwe 0020e301 00d5957c 0053e5c8           0 00d55614 16048/16384 isakmp_time_keepr
Lsi 002d377c 00d7292c 0053e5c8           0 00d719a4 3928/4096 perfmon
Hwe 0020bd07 00d9c12c 0050bb90           0 00d9b1c4 3944/4096 IPSec
Mwe 00205e25 00d9e1ec 0053e5c8           0 00d9c274 7860/8192 IPsec timer handler
Hwe 003864e3 00db26bc 00557920           0 00db0764 6904/8192 qos_metric_daemon
Mwe 00255a65 00dc9244 0053e5c8           0 00dc8adc 1436/2048 IP Background
Lwe 002e450e 00e7bb94 00552c30           0 00e7ad1c 3704/4096 pix/trace
Lwe 002e471e 00e7cc44 00553368           0 00e7bdcc 3704/4096 pix/tconsole
Hwe 001e5368 00e7ed44 00730674           0 00e7ce9c 7228/8192 pix/intf0
Hwe 001e5368 00e80e14 007305d4           0 00e7ef6c 7228/8192 pix/intf1
Hwe 001e5368 00e82ee4 00730534        2470 00e8103c 4892/8192 pix/intf2
H*  001a6ff5 0009ff2c 0053e5b0        4820 00e8511c 12860/16384 ci/console
Csi 002dd8ab 00e8a124 0053e5c8           0 00e891cc 3396/4096 update_cpu_usage
Hwe 002cb4d1 00f2bfbc 0051e360           0 00f2a134 7692/8192 uauth_in
Hwe 003d17d1 00f2e0bc 00828cf0           0 00f2c1e4 7896/8192 uauth_thread
Hwe 003e71d4 00f2f20c 00537d20           0 00f2e294 3960/4096 udp_timer
Hsi 001db3ca 00f30fc4 0053e5c8           0 00f3004c 3784/4096 557mcfix
Crd 001db37f 00f32084 0053ea40   508286220 00f310fc 3688/4096 557poll
Lsi 001db435 00f33124 0053e5c8           0 00f321ac 3700/4096 557timer
Hwe 001e5398 00f441dc 008121e0           0 00f43294 3912/4096 fover_ip0
Cwe 001dcdad 00f4523c 00872b48         120 00f44344 3528/4096 ip/0:0
Hwe 001e5398 00f4633c 008121bc          10 00f453f4 3532/4096 icmp0
Hwe 001e5398 00f47404 00812198           0 00f464cc 3896/4096 udp_thread/0
Hwe 001e5398 00f4849c 00812174           0 00f475a4 3456/4096 tcp_thread/0
Hwe 001e5398 00f495bc 00812150           0 00f48674 3912/4096 fover_ip1
Cwe 001dcdad 00f4a61c 008ea850           0 00f49724 3832/4096 ip/1:1
Hwe 001e5398 00f4b71c 0081212c           0 00f4a7d4 3912/4096 icmp1
Hwe 001e5398 00f4c7e4 00812108           0 00f4b8ac 3896/4096 udp_thread/1
Hwe 001e5398 00f4d87c 008120e4           0 00f4c984 3832/4096 tcp_thread/1
Hwe 001e5398 00f4e99c 008120c0           0 00f4da54 3912/4096 fover_ip2
Cwe 001e542d 00f4fa6c 00730534           0 00f4eb04 3944/4096 ip/2:2
Hwe 001e5398 00f50afc 0081209c           0 00f4fbb4 3912/4096 icmp2
Hwe 001e5398 00f51bc4 00812078           0 00f50c8c 3896/4096 udp_thread/2
Hwe 001e5398 00f52c5c 00812054           0 00f51d64 3832/4096 tcp_thread/2
Hwe 003d1a65 00f78284 008140f8           0 00f77fdc  300/1024 listen/http1
Mwe 0035cafa 00f7a63c 0053e5c8           0 00f786c4 7640/8192 Crypto CA

------------------ show failover ------------------

No license for Failover

------------------ show traffic ------------------

outside:
        received (in 865565.090 secs):
                6139 packets    830375 bytes
                0 pkts/sec      0 bytes/sec
        transmitted (in 865565.090 secs):
```

```
                90 packets      6160 bytes
                0 pkts/sec      0 bytes/sec
inside:
        received (in 865565.090 secs):
                0 packets       0 bytes
                0 pkts/sec      0 bytes/sec
        transmitted (in 865565.090 secs):
                1 packets       60 bytes
                0 pkts/sec      0 bytes/sec
intf2:
        received (in 865565.090 secs):
                0 packets       0 bytes
                0 pkts/sec      0 bytes/sec
        transmitted (in 865565.090 secs):
                0 packets       0 bytes
                0 pkts/sec      0 bytes/sec

------------------ show perfmon ------------------


PERFMON STATS:    Current      Average
Xlates               0/s          0/s
Connections          0/s          0/s
TCP Conns            0/s          0/s
UDP Conns            0/s          0/s
URL Access           0/s          0/s
URL Server Req       0/s          0/s
TCP Fixup            0/s          0/s
TCPIntercept         0/s          0/s
HTTP Fixup           0/s          0/s
FTP Fixup            0/s          0/s
AAA Authen           0/s          0/s
AAA Author           0/s          0/s
AAA Account          0/s          0/s
: End_Test_Crash
```

Related Commands

Command	Description
clear crashinfo	Deletes the contents of the crash file.
crashinfo force	Forces a crash of the security appliance.
crashinfo save disable	Disables crash information from writing to Flash memory.
crashinfo test	Tests the ability of the security appliance to save crash information to a file in Flash memory.

show crypto engine [verify]

Shows cryptography engine statistics or runs the Known Answer Test (KAT).

show crypto engine [**verify**]

Syntax Description

crypto engine	Displays usage statistics for the firewall cryptography engine.
verify	Runs the Known Answer Test (KAT).

Command Modes

Privileged or configuration mode.

Usage Guidelines

The **show crypto engine** command displays usage statistics for the cryptography engine used by the firewall.

The **show crypto engine verify** command runs the Known Answer Test (KAT) from the firewall CLI. Additionally, when booted for the first time or after a reload, the firewall performs the Know Answer Test (KAT) before any configuration information is read from the Flash memory. If the KAT fails, then the firewall issues an error message and reloads. The KAT is performed to check the integrity of the cryptography engine used by the firewall.

Examples

The following example shows sample output for the **show crypto engine** command:

```
pixfirewall# show crypto engine
Crypto Engine Connection Map:
    size = 8, free = 6, used = 1, active = 1
```

In this command output, *size* is total number of undirectional IPSec tunnels, *free* is the number of unused undirectional IPSec tunnels, *used* is the number of allocated undirectional IPSec tunnels, and *active* is the number of active undirectional IPSec tunnels. Because tunnel 0 is reserved for system use, *size* is equal to *free* plus *used* plus one.

The following example shows sample output for the **show crypto engine** command when output is specified for a VAC or a VAC+:

```
VAC+:

pixfirewall# show crypto interface
Encryption hardware device : VAC+ (Crypto5823 revision 0x1)

VAC:

pixfirewall# show crypto interface
Encryption hardware device : VAC (IRE2141 with 2048KB, HW:1.0, CGXROM:1.9, FW:6.5)
```

The following example shows the **show crypto engine verify** command output for a successful KAT:

```
pixfirewall# show crypto engine verify
FIPS: Known Answer Test begin

FIPS: software DES      success
FIPS: software SHA      success
FIPS: software RSA      success

FIPS:    software to software       DES/SHA1 tunnel check success.

FIPS: Known Answer Test finish
```

The following is sample output from a KAT that failed during start up of the firewall:

```
Cisco PIX Firewall Version 6.3(1)
Licensed Features:
Failover: Enabled
VPN-DES: Enabled
VPN-3DES-AES: Enabled
Maximum Interfaces: 6
Cut-through Proxy: Enabled
Guards: Enabled
URL-filtering: Enabled
```

```
Inside Hosts: Unlimited
Throughput: Unlimited
IKE peers: Unlimited

This PIX has an Unrestricted (UR) license.

FIPS: software AES fail

An internal error occurred. Specifically, a programming assertion was
violated. Copy the error message exactly as it appears, and get the
output of the show version command and the contents of the configuration
file. Then call your technical support representative.

assertion "result != FALSE" failed: file "crypto_nist_tests.c", line 529
No thread name

Traceback:
0: 0040d84d
1: 00260608
...
```

show crypto interface [counters]

Displays the VPN accelerator cards (VACs) installed in the firewall chassis and, for the VAC+, the packet, payload byte, queue length, and moving average counters for traffic moving through the card.

show crypto interface [**counters**]

clear crypto interface counters

Syntax Description

counters	Displays packet count, byte queue, and moving averages for traffic through a VAC+.
crypto interface	Displays the VPN accelerator cards (VACs) installed in the firewall chassis.

Command Modes

Privileged or configuration mode.

Usage Guidelines

The **show crypto interface** command lists VPN accelerator cards (VACs) installed in the firewall chassis. (This same information is also displayed in **show version** output.)

The **show crypto interface counters** command displays information, as described in Table 8-3, for the PIX Firewall VAC+ card only.

Table 8-3 show crypto interface counters

Counter	Description
`interfaces`	The number and type of crypto interface cards installed.
`packet count`	The number of packets sent to the installed crypto interface card(s).

Table 8-3 show crypto interface counters

Counter	Description
payload bytes	The number of bytes of payload either after decapsulation or before encapsulation.
input queue (curr/max)	The total number of packets that are awaiting service from the crypto interface card(s).
interface queue (curr/max)	The total number of packets that have been queued at the crypto interface card(s) for service.
output queue (curr/max)	The total number of packets that have been released by the crypto interface card(s) and are awaiting dispatch to the packet path.
moving averages 5second 1minute 5minute	5 second, 1 minute, and 5 minute moving averages of the packet count and payload bytes through all crypto interface cards.

The **clear crypto interface counters** command clears only the packet, payload byte, queue length, and moving average counters. It does not affect any actual packets queued.

Examples

The following is sample output from the **show crypto interface** and **show crypto interface counters** commands when a VAC+ card is installed:

```
pixfirewall# show crypto interface
Encryption hardware device : Crypto5823 (revision 0x1)
pixfirewall(config)# show crypto interface counters

interfaces: 1
  Crypto5823 (revision 0x1), maximum queue size 64

packet count:               318657093
payload bytes:              89861300946
input     queue (curr/max): 1336/1584
interface queue (curr/max): 64/64
output    queue (curr/max): 0/64

moving averages
  5second   128273 pkts/sec   289 Mbits/sec
  1minute   128326 pkts/sec   290 Mbits/sec
  5minute   128279 pkts/sec   289 Mbits/sec
```

The following is the same sample output after the **clear crypto interface counters** command has been used:

```
pixfirewall# clear crypto interface counters
pixfirewall# show crypto interface counters
interfaces: 1
  Crypto5823 (revision 0x1), maximum queue size 64

packet count:               355968
payload bytes:              100382976
input     queue (curr/max): 1317/1537
interface queue (curr/max): 64/64
output    queue (curr/max): 0/64

moving averages
  5second       NA pkts/sec    NA Mbits/sec
```

```
  1minute       NA pkts/sec    NA Mbits/sec
  5minute       NA pkts/sec    NA Mbits/sec
```

The following is sample output from the **show crypto interface** and **show crypto interface counters** commands when a VAC card is installed:

```
pixfirewall# show crypto interface
Encryption hardware device : IRE2141 with 2048KB, HW:1.0, CGXROM:1.9, FW:6.5
pixfirewall# show crypto interface counters
no crypto interface counters available
```

The following is sample output from the **show crypto interface** and **show crypto interface counters** commands when no crypto interface card is installed (neither a VAC nor a VAC+):

```
pixfirewall# show crypto interface
pixfirewall# show crypto interface counters
no crypto interface counters available
```

show ip local pool

The **show ip local pool** command displays:

- any included netmask if it is configured.
- fixes an alignment problem if present with possible varied length pool names.

Syntax Description

ip local pool	List of configured local pool IP addresses

Command Modes

Configuration mode.

Usage Guidelines

The **show ip local pool** command can now output the netmask if it is configured. The **show ip local pool** command displays previously configured local pool addresses.

The following is sample output from the **show ip local pool** command:

```
argus-520(config)# sh ip local pool
Pool            Begin           End             Mask            Free    In use
VPNClient       10.1.0.1        10.1.0.25       Not configured    25         0
...
Pool            Begin           End             Mask            Free    In use
ReallyReallyReallyLongPoolName
                192.168.0.1     192.168.64.0    255.255.0.0     16384         0
```

show history

Display previously entered commands.

show history

Syntax Description		
	history	The list of previous entries.

Command Modes Available in unprivileged mode, privileged mode, and configuration mode.

Usage Guidelines The **show history** command displays previously entered commands. You can examine commands individually with the up and down arrows or by entering **^p** to view previously entered lines or **^n** to view the next line.

Examples The following is sample output from the **show history** command when run in unprivileged mode:

```
pixfirewall> show history
    show history
    help
    show history
```

The following is sample output from the **show history** command when run in privileged mode:

```
pixfirewall# show history
    show history
    help
    show history
    enable
    show history
```

The following is sample output from the **show history** command when run in configuration mode:

```
pixfirewall(config)# show history
    show history
    help
    show history
    enable
    show history
    config t
    show history
```

show local-host/clear local host

View local host network states.

show local-host [*ip_address*]

clear local-host [*ip_address*]

Syntax Description		
	ip_address	Local host IP address.

Command Modes Privileged mode for the **show** commands and configuration mode for the **clear** commands.

Usage Guidelines

The **show local-host** command displays the translation and connection slots for all local hosts. This command also provides information for hosts configured with the **nat 0** command when normal translation and connection states may not apply. The **show local-host detail** command displays more information about active xlates and connections. Use the *ip_address* option to limit the display to a single host.

The **clear local-host** command stops traffic on all local hosts. The **clear local-host** *ip_address* command stops traffic on the local host specified by its IP address.

On a PIX 501, cleared hosts are released from the license limit. You can view the number of hosts that are counted toward the license limit with the **show local-host** command.

Note Clearing the network state of a local host stops all connections and xlates associated with the local hosts.

Examples

The following is sample output from the **show local-host** command:

```
show local-host 10.1.1.15
local host: <10.1.1.15>, conn(s)/limit = 2/0, embryonic(s)/limit = 0/0
   Xlate(s):
       PAT Global 172.16.3.200(1024) Local 10.1.1.15(55812)
       PAT Global 172.16.3.200(1025) Local 10.1.1.15(56836)
       PAT Global 172.16.3.200(1026) Local 10.1.1.15(57092)
       PAT Global 172.16.3.200(1027) Local 10.1.1.15(56324)
       PAT Global 172.16.3.200(1028) Local 10.1.1.15(7104)
   Conn(s):
       TCP out 192.150.49.10:23 in 10.1.1.15:1246 idle 0:00:20 Bytes 449 flags UIO
       TCP out 192.150.49.10:21 in 10.1.1.15:1247 idle 0:00:10 Bytes 359 flags UIO
```

The xlate describes the translation slot information and the Conn is the connection state information.

The following is sample command output from the **show local-host** command:

```
pixfirewall(config)# show local-host
local host: <10.1.1.15>, conn(s)/limit = 2/0, embryonic(s)/limit = 0/0
    Xlate(s):
        PAT Global 192.150.49.1(1024) Local 10.1.1.15(516)
        PAT Global 192.150.49.1(0) Local 10.1.1.15 ICMP id 340
        PAT Global 192.150.49.1(1024) Local 10.1.1.15(1028)
    Conn(s):
        TCP out 192.150.49.10:23 in 10.1.1.15:1026 idle 0:00:25
                  Bytes 1774 flags UIO
        UDP out 192.150.49.10:31649 in 10.1.1.15:1028 idle 0:00:17
                  flags D-
```

For comparison, the following is sample command output from the **show local-host detail** command:

```
pixfirewall(config)# show local-host detail
     local host: <10.1.1.15>,
         TCP connection count/limit = 0/unlimited
         TCP embryonic count = 0
         TCP intercept watermark = unlimited
         UDP connection count/limit = 0/unlimited
         Xlate(s):
             TCP PAT from inside:10.1.1.15/1026 to outside:192.150.49.1/1024
                       flags ri
             ICMP PAT from inside:10.1.1.15/21505 to outside:192.150.49.1/0
                       flags ri
             UDP PAT from inside:10.1.1.15/1028 to outside:192.150.49.1/1024
                       flags ri
         Conn(s):
```

```
TCP outside:192.150.49.10/23 inside:10.1.1.15/1026 flags UIO
UDP outside:192.150.49.10/31649 inside:10.1.1.15/1028 flags dD
```

The next example shows how the **clear local-host** command clears the local host information:

```
clear local-host 10.1.1.15
show local-host 10.1.1.15
```

Once the information is cleared, nothing more displays until the hosts reestablish their connections, which were stopped by the **clear local-host** command, and more data is produced.

show memory

Show system memory utilization.

show memory[*detail*]

Syntax Description

memory	The system memory data.
detail	Additional detail on system memory data.

Command Modes

Privileged mode.

Usage Guidelines

The **show memory** command displays a summary of the maximum physical memory and current free memory available to the PIX Firewall operating system. Memory in the PIX Firewall is allocated as needed.

You can also view the information from the **show memory** command using SNMP.

Examples

The following is sample output from the **show memory** command:

```
Result of firewall command: "show memory"

Free memory:        17149656 bytes
Used memory:        16404776 bytes
-------------     ----------------
Total memory:       33554432 bytes
```

The following is sample output from the **show memory detail** command:

```
Result of firewall command: "show memory detail"

Free memory:                     49734280 bytes
Used memory:
     Allocated memory in use:    11175212 bytes
     Reserved memory:             6199372 bytes
----------------------------   ----------------
Total memory:                    67108864 bytes
----- fragmented memory statistics -----
 fragment size       count          total
    (bytes)                        (bytes)
---------------  ----------  --------------
```

```
          16             9            144
          24             2             48
          80             2            160
         128             1            128
         216             1            216
         224             3            672
         232             1            232
         240             1            240
         296             2            592
         312             1            312
         320             1            320
         816             1            816
        1136             1           1136
        1336             1           1336
        5504             3          16528
        5784             4          23280
        9024             1           9024
       13744             1          13744*
       14768             4          61400
       21872             1          21872
    49582080             1       49582080**
*  - top most releasable chunk.
** - contiguous memory on top of heap.
----- allocated memory statistics -----
 fragment size       count          total
    (bytes)                        (bytes)
----------------  ----------  --------------
          24            20            480
          32          1426          45632
          40          1439          57560
          48           197           9456
          56          1013          56728
          64           110           7040
          72            69           4968
          80            35           2800
          88            56           4928
          96            10            960
         104            12           1248
         112            15           1680
         120            19           2280
         128             7            896
         136            35           4760
         144             3            432
         152            61           9272
         160             7           1120
         168             1            168
         176             1            176
         184             1            184
         200             6           1200
         216             1            216
         224             4            896
         256             1            256
         264           706         186384
         272             2            544
         280             4           1120
         288             2            576
         296             1            296
         304             1            304
         328             1            328
         344             2            688
         352            62          21824
         360             3           1080
         392             1            392
         424             4           1696
```

```
  464             1            464
  512            21          10752
  576             4           2304
  640             1            640
  704             2           1408
  768             2           1536
  832             1            832
  896             1            896
 1024             7           7168
 1088             1           1088
 1152             2           2304
 1408             6           8448
 1536             1           1536
 1600             3           4800
 1664             1           1664
 1856             1           1856
 1920             1           1920
 2048             1           2048
 2112             1           2112
 2240             1           2240
 2368             1           2368
 3072            16          49152
 4096            51         208896
 4608             1           4608
 8192            14         114688
 9728             1           9728
10240             1          10240
10752             1          10752
14848             3          44544
18944            28         530432
23040             1          23040
27136             2          54272
31232             4         124928
39424             3         118272
76288            15        1144320
141824            7         992768
174592            4         698368
436736           10        4367360
698880            3        2096640
```

show ospf

Displays general information about OSPF routing processes.

show ospf [*pid*]

Syntax Description

pid	The ID of the OSPF process.

Defaults

The default is to list all OSPF processes if no *pid* is specified.

Command Modes

The **show ospf** command is available in privileged mode.

Usage Guidelines

The OSPF routing-related **show** commands are available in privileged command mode on the firewall. You do not need to be in an OSPF configuration subcommand mode to use the OSPF-related **show** commands.

If the *pid* is included, only information for the specified routing process is included.

Examples

The following examples are sample output from the **show ospf** [*pidpid*] (with a *pid* of 5)and **show ospf** commands:

```
pixfirewall# show ospf 5

 Routing Process "ospf 5" with ID 127.0.0.1 and Domain ID 0.0.0.5
 Supports only single TOS(TOS0) routes
 Supports opaque LSA
 SPF schedule delay 5 secs, Hold time between two SPFs 10 secs
 Minimum LSA interval 5 secs. Minimum LSA arrival 1 secs
 Number of external LSA 0. Checksum Sum 0x     0
 Number of opaque AS LSA 0. Checksum Sum 0x     0
 Number of DCbitless external and opaque AS LSA 0
 Number of DoNotAge external and opaque AS LSA 0
 Number of areas in this router is 0. 0 normal 0 stub 0 nssa
 External flood list length 0

pixfirewall# show ospf

 Routing Process "ospf 5" with ID 127.0.0.1 and Domain ID 0.0.0.5
 Supports only single TOS(TOS0) routes
 Supports opaque LSA
 SPF schedule delay 5 secs, Hold time between two SPFs 10 secs
 Minimum LSA interval 5 secs. Minimum LSA arrival 1 secs
 Number of external LSA 0. Checksum Sum 0x     0
 Number of opaque AS LSA 0. Checksum Sum 0x     0
 Number of DCbitless external and opaque AS LSA 0
 Number of DoNotAge external and opaque AS LSA 0
 Number of areas in this router is 0. 0 normal 0 stub 0 nssa
 External flood list length 0

 Routing Process "ospf 12" with ID 172.23.59.232 and Domain ID 0.0.0.12
 Supports only single TOS(TOS0) routes
 Supports opaque LSA
 SPF schedule delay 5 secs, Hold time between two SPFs 10 secs
 Minimum LSA interval 5 secs. Minimum LSA arrival 1 secs
 Number of external LSA 0. Checksum Sum 0x     0
 Number of opaque AS LSA 0. Checksum Sum 0x     0
 Number of DCbitless external and opaque AS LSA 0
 Number of DoNotAge external and opaque AS LSA 0
 Number of areas in this router is 0. 0 normal 0 stub 0 nssa
 External flood list length 0
```

Related Commands

prefix-list	Configures a prefix list to be used for OSPF routing.
route-map	Creates a route map for redistributing routes from one routing protocol to another. Used in configuring OSPF routing on the firewall.
router ospf	Configures global parameters for the OSPF routing processes on the firewall, and enables or disables OSPF routing through the firewall.
routing interface	Configures interface-specific OSPF routing parameters.

show ospf border-routers

Displays the internal OSPF routing table entries to an Area Border Router (ABR) and Autonomous System Boundary Router (ASBR).

show ospf border-routers

Syntax Description

border-routers	Area Border Routers (ABRs) and Autonomous System Boundary Routers (ASBRs).

Defaults

None.

Command Modes

The **show ospf border-routers** command is available in privileged mode.

Usage Guidelines

The OSPF routing-related **show** commands are available in privileged command mode on the firewall. You do not need to be in an OSPF configuration subcommand mode to use the OSPF-related **show** commands.

Examples

The following is sample output from the **show ospf border-routers** command:

```
pixfirewall# show ospf border-routers
OSPF Process 109 internal Routing Table
Destination Next Hop Cost Type Rte Type Area SPF No
192.168.97.53 172.16.1.53 10 ABR INTRA 0.0.0.3 3
192.168.103.51 192.168.96.51 10 ABR INTRA 0.0.0.3 3
192.168.103.52 192.168.96.51 20 ASBR INTER 0.0.0.3 3
192.168.103.52 172.16.1.53 22 ASBR INTER 0.0.0.3 3
```

Related Commands

prefix-list	Configures a prefix list to be used for OSPF routing.
route-map	Creates a route map for redistributing routes from one routing protocol to another. Used in configuring OSPF routing on the firewall.
router ospf	Configures global parameters for the OSPF routing processes on the firewall, and enables or disables OSPF routing through the firewall.
routing interface	Configures interface-specific OSPF routing parameters.

show ospf database

Displays LSA information in the OSPF database for a specific network area or router.

show ospf [*pid*] **database** [**internal**] [**adv-router** [*ip_address*]]

show ospf [*pid* [*area_id*]] **database** [**internal**] [**self-originate**] [*lsid*]

show ospf [*pid* [*area_id*]] **database** {**router** | **network** | **summary** | **asbr-summary** | **external** | **nssa-external** | **database-summary**}]

Syntax Description

adv-router [*ip_address*]	Displays all the link-state advertisements (LSAs) of the specified router. If no IP address is included, the information is about the local router itself (in that case, the output is the same as with the **self-originate** keyword).
area_id	The ID of the area that is associated with the OSPF address range. If you intend to associate areas with IP subnets, you can specify a subnet address as the *area_id*. When used in the context of authentication, *area_id* is the identifier of the area on which authentication is to be enabled. When using a cost context, *area_id* is the identifier for the stub or NSSA. When used in the context of a prefix list, *area_id* is the identifier of the area on which filtering is configured. When used in a stub area or not-so-stubby area (NSSA) context, *area_id* is the identifier for the stub or NSSA area. When used in the context of an area range, *area_id* is the identifier of the area at whose boundary to summarize routes.
asbr-summary	Displays information only about the Autonomous System Boundary Router (ASBR) summary LSAs.
database-summary	Displays how many of each type of LSA for each area there are in the database, and the total.
external	Routes external to a specified autonomous system.
internal	Routes that are internal to a specified autonomous system.
ip_address	The IP address of the OSPF router.
lsid	The link state ID, specified as an IP address. The *lsid* describes the portion of the Internet environment that is being described by the link-state advertisement (LSA). The value entered depends on the type of the LSA, but the value must be entered in the form of an IP address, as follows: • When the LSA is describing a network, set *lsid* to the network IP address (for Type 3 summary link advertisements and for autonomous system external link advertisements) or a derived IP address with the network subnet mask (from which the OSPF process interprets the network IP address). • When the LSA is describing a router, set *lsid* to the OSPF router ID of the router. • When an autonomous system external advertisement (Type 5) is describing a default route, set *lsid* to the default destination (0.0.0.0).
network	Displays information only about the network LSAs.
nssa-external	Displays information only about the not-so-stubby area (NSSA) external LSAs.
pid	The ID of the OSPF process.
router	Displays information only about the router LSAs.

self-originate	Displays only self-originated LSAs (from the local router).
summary	Displays information only about the summary LSAs.

Defaults

None.

Command Modes

The **show ospf database** command is available in privileged mode.

Usage Guidelines

The OSPF routing-related **show** commands are available in privileged command mode on the firewall. You do not need to be in an OSPF configuration subcommand mode to use the OSPF-related **show** commands.

The various forms of this command deliver information about different OSPF link-state advertisements (LSAs).

Examples

The following is sample output from the **show ospf database** command:

```
pixfirewall# show ospf database
OSPF Router with ID(192.168.1.11) (Process ID 1)
                Router Link States(Area 0)
Link ID   ADV Router   Age   Seq# Checksum Link count
192.168.1.8 192.168.1.8 1381 0x8000010D    0xEF60 2
192.168.1.11 192.168.1.11 1460 0x800002FE    0xEB3D 4
192.168.1.12 192.168.1.12 2027 0x80000090    0x875D 3
192.168.1.27 192.168.1.27 1323 0x800001D6    0x12CC 3
                Net Link States(Area 0)
Link ID ADV Router   Age   Seq# Checksum
172.16.1.27 192.168.1.27 1323 0x8000005B    0xA8EE
172.17.1.11 192.168.1.11 1461 0x8000005B    0x7AC
                Type-10 Opaque Link Area Link States (Area 0)
Link ID ADV Router   Age Seq# Checksum Opaque ID
10.0.0.0 192.168.1.11 1461 0x800002C8    0x8483   0
10.0.0.0 192.168.1.12 2027 0x80000080    0xF858   0
10.0.0.0 192.168.1.27 1323 0x800001BC    0x919B   0
10.0.0.1 192.168.1.11 1461 0x8000005E    0x5B43   1
```

The following is sample output from the **show ospf database asbr-summary** command:

```
pixfirewall# show ospf database asbr-summary
OSPF Router with ID(192.168.239.66) (Process ID 300)
Summary ASB Link States(Area 0.0.0.0)
Routing Bit Set on this LSA
LS age: 1463
Options: (No TOS-capability)
LS Type: Summary Links(AS Boundary Router)
Link State ID: 172.16.245.1 (AS Boundary Router address)
Advertising Router: 172.16.241.5
LS Seq Number: 80000072
Checksum: 0x3548
Length: 28
Network Mask: 0.0.0.0
TOS: 0 Metric: 1
```

The following is sample output from the **show ospf database router** command:

```
pixfirewall# show ospf database router
OSPF Router with id(192.168.239.66) (Process ID 300)
Router Link States(Area 0.0.0.0)
Routing Bit Set on this LSA
LS age: 1176
Options: (No TOS-capability)
LS Type: Router Links
Link State ID: 10.187.21.6
Advertising Router: 10.187.21.6
LS Seq Number: 80002CF6
Checksum: 0x73B7
Length: 120
AS Boundary Router
Number of Links: 8
Link connected to: another Router (point-to-point)
(link ID) Neighboring Router ID: 10.187.21.5
(Link Data) Router Interface address: 10.187.21.6
Number of TOS metrics: 0
TOS 0 Metrics: 2
```

The following is sample output from the **show ospf database network** command:

```
pixfirewall# show ospf database network
OSPF Router with id(192.168.239.66) (Process ID 300)
Displaying Net Link States(Area 0.0.0.0)
LS age: 1367
Options: (No TOS-capability)
LS Type: Network Links
Link State ID: 10.187.1.3 (address of Designated Router)
Advertising Router: 192.168.239.66
LS Seq Number: 800000E7
Checksum: 0x1229
Length: 52
Network Mask: 255.255.255.0
Attached Router: 192.168.239.66
Attached Router: 10.187.241.5
Attached Router: 10.187.1.1
Attached Router: 10.187.54.5
Attached Router: 10.187.1.5
```

The following is sample output from the **show ospf database summary** command:

```
pixfirewall# show ospf database summary
OSPF Router with id(192.168.239.66) (Process ID 300)
Displaying Summary Net Link States(Area 0.0.0.0)
LS age: 1401
Options: (No TOS-capability)
LS Type: Summary Links(Network)
Link State ID: 10.187.240.0 (summary Network Number)
Advertising Router: 10.187.241.5
LS Seq Number: 80000072
Checksum: 0x84FF
Length: 28
Network Mask: 255.255.255.0 TOS: 0 Metric: 1
```

The following is sample output from the **show ospf database external** command:

```
pixfirewall# show ospf database external
OSPF Router with id(192.168.239.66) (Autonomous system 300)
                  Displaying AS External Link States
LS age: 280
Options: (No TOS-capability)
LS Type: AS External Link
Link State ID: 143.10.0.0 (External Network Number)
Advertising Router: 10.187.70.6
```

```
LS Seq Number: 80000AFD
Checksum: 0xC3A
Length: 36
Network Mask: 255.255.0.0
      Metric Type: 2 (Larger than any link state path)
      TOS: 0
      Metric: 1
      Forward Address: 0.0.0.0
      External Route Tag: 0
```

Related Commands

prefix-list	Configures a prefix list to be used for OSPF routing.
route-map	Creates a route map for redistributing routes from one routing protocol to another. Used in configuring OSPF routing on the firewall.
router ospf	Configures global parameters for the OSPF routing processes on the firewall, and enables or disables OSPF routing through the firewall.
routing interface	Configures interface-specific OSPF routing parameters.

show ospf flood-list

Displays a list of OSPF link-state advertisements (LSAs) waiting to be flooded over an interface.

show ospf flood-list *if_name*

Syntax Description

flood-list	The list of link-state advertisements (LSAs) waiting to be flooded over an interface.
if_name	The name of the interface for which to display neighbor information.

Defaults

None.

Command Modes

The **show ospf flood-list** command is available in privileged mode.

Usage Guidelines

The OSPF routing-related **show** commands are available in privileged command mode on the firewall. You do not need to be in an OSPF configuration subcommand mode to use the OSPF-related **show** commands.

Examples

The following is sample output from the **show ospf flood-list** command, where the *if_name* is **outside**:

```
pixfirewall# show ospf flood-list outside
Interface outside, Queue length 20
Link state flooding due in 12 msec
Type LS ID  ADV RTR   Seq NO Age Checksum
5 10.2.195.0 192.168.0.163 0x80000009 0 0xFB61
5 10.1.192.0 192.168.0.163 0x80000009 0 0x2938
5 10.2.194.0 192.168.0.163 0x80000009 0 0x757
5 10.1.193.0 192.168.0.163 0x80000009 0 0x1E42
```

```
5 10.2.193.0 192.168.0.163 0x80000009 0 0x124D
5 10.1.194.0 192.168.0.163 0x80000009 0 0x134C
```

Related Commands

prefix-list	Configures a prefix list to be used for OSPF routing.
route-map	Creates a route map for redistributing routes from one routing protocol to another. Used in configuring OSPF routing on the firewall.
router ospf	Configures global parameters for the OSPF routing processes on the firewall, and enables or disables OSPF routing through the firewall.
routing interface	Configures interface-specific OSPF routing parameters.

show ospf interface

Displays OSPF-related interface information.

show ospf interface *if_name*

Syntax Description

if_name	The name of the interface for which to display OSPF-related information.

Defaults

None.

Command Modes

The **show ospf interface** *if_name* command is available in privileged mode.

Usage Guidelines

The OSPF routing-related **show** commands are available in privileged command mode on the firewall. You do not need to be in an OSPF configuration subcommand mode to use the OSPF-related **show** commands.

Examples

The following is sample output from the **show ospf interface** *if_name* command, where the *if_name* is **inside**:

```
pixfirewall# show ospf interface inside
inside is up, line protocol is up
Internet Address 192.168.254.202, Mask 255.255.255.0, Area 0.0.0.0
AS 201, Router ID 192.77.99.1, Network Type BROADCAST, Cost: 10
Transmit Delay is 1 sec, State OTHER, Priority 1
Designated Router id 192.168.254.10, Interface address 192.168.254.10
Backup Designated router id 192.168.254.28, Interface addr 192.168.254.28
Timer intervals configured, Hello 10, Dead 60, Wait 40, Retransmit 5
Hello due in 0:00:05
Neighbor Count is 8, Adjacent neighbor count is 2
  Adjacent with neighbor 192.168.254.28 (Backup Designated Router)
  Adjacent with neighbor 192.168.254.10 (Designated Router)
```

Related Commands

prefix-list	Configures a prefix list to be used for OSPF routing.
route-map	Creates a route map for redistributing routes from one routing protocol to another. Used in configuring OSPF routing on the firewall.
router ospf	Configures global parameters for the OSPF routing processes on the firewall, and enables or disables OSPF routing through the firewall.
routing interface	Configures interface-specific OSPF routing parameters.

show ospf neighbor

Displays OSPF-neighbor information on a per-interface basis.

show ospf neighbor [*if_name*] [*nbr_router_id*] [**detail**]

Syntax Description

detail	List all neighbors.
if_name	The name of the interface for which to display neighbor information.
nbr_router_id	The IP address of the neighbor router.

Defaults

None.

Command Modes

The **show ospf neighbor** command is available in privileged mode.

Usage Guidelines

The OSPF routing-related **show** commands are available in privileged command mode on the firewall. You do not need to be in an OSPF configuration subcommand mode to use the OSPF-related **show** commands.

Examples

The following is sample output from the **show ospf neighbor** *if_name nbr_router_id* command, where the *if_name* is **inside** and the *nbr_router_id* is 10.199.199.137:

```
pixfirewall# show ospf neighbor inside 10.199.199.137
Neighbor 10.199.199.137, interface address 192.168.80.37
In the area 0.0.0.0 via interface inside
Neighbor priority is 1, State is FULL
Options 2
Dead timer due in 0:00:37
Link State retransmission due in 0:00:04
```

The following is sample output from the **show ospf neighbor detail** command, where the *if_name* is **outside**:

```
pixfirewall# show ospf neighbor outside detail
Neighbor 192.168.5.2, interface address 10.225.200.28
In the area 0 via interface outside
Neighbor priority is 1, State is FULL, 6 state changes
DR is 10.225.200.28 BDR is 10.225.200.30
Options is 0x42
Dead timer due in 00:00:36
```

```
Neighbor is up for 00:09:46
Index 1/1, retransmission queue length 0, number of retransmission 1
First 0x0(0)/0x0(0) Next 0x0(0)/0x0(0)
Last retransmission scan length is 1, maximum is 1
Last retransmission scan time is 0 msec, maximum is 0 msec
```

Related Commands

prefix-list	Configures a prefix list to be used for OSPF routing.
route-map	Creates a route map for redistributing routes from one routing protocol to another. Used in configuring OSPF routing on the firewall.
router ospf	Configures global parameters for the OSPF routing processes on the firewall, and enables or disables OSPF routing through the firewall.
routing interface	Configures interface-specific OSPF routing parameters.

show ospf request-list

Displays a list of all link-state advertisements (LSAs) requested by a router.

show ospf request-list *nbr_router_id if_name*

Syntax Description

if_name	The name of the interface for which to display neighbor information. Displays the list of all LSAs requested by the router from this interface.
nbr_router_id	The ID of the neighbor router, specified by IP address. Displays the list of all LSAs requested by the router from this neighbor.

Defaults

None.

Command Modes

The **show ospf request-list** *nbr_router_id if_name* command is available in privileged mode.

Usage Guidelines

The OSPF routing-related **show** commands are available in privileged command mode on the firewall. You do not need to be in an OSPF configuration subcommand mode to use the OSPF-related **show** commands.

Examples

The following is sample output from the **show ospf request-list** command, where the *nbr_router_id* is 192.168.1.12 and the *if_name* is **inside**:

```
pixfirewall# show ospf request-list 192.168.1.12 inside
OSPF Router with ID (192.168.1.11) (Process ID 1)
Neighbor 192.168.1.12, interface inside address 172.16.1.12
Type LS ID   ADV RTR   Seq NO Age Checksum
1 192.168.1.12 192.168.1.12 0x8000020D 8 0x6572
```

Related Commands

prefix-list	Configures a prefix list to be used for OSPF routing.
route-map	Creates a route map for redistributing routes from one routing protocol to another. Used in configuring OSPF routing on the firewall.
router ospf	Configures global parameters for the OSPF routing processes on the firewall, and enables or disables OSPF routing through the firewall.
routing interface	Configures interface-specific OSPF routing parameters.

show ospf retransmission-list

Displays a list of all link-state advertisements (LSAs) waiting to be resent.

show retransmission-list *nbr_router_id if_name*

Syntax Description

if_name	The name of the interface for which to display neighbor information. Displays the list of all LSAs waiting to be resent for this neighbor.
nbr_router_id	The ID of the neighbor router, specified by IP address. Displays the list of all LSAs waiting to be resent for this interface.

Defaults

None.

Command Modes

The **show retransmission-list** *nbr_router_id if_name* command is available in privileged mode.

Usage Guidelines

The OSPF routing-related **show** commands are available in privileged command mode on the firewall. You do not need to be in an OSPF configuration subcommand mode to use the OSPF-related **show** commands.

Examples

The following is sample output from the **show ospf retransmission-list** command, where the *nbr_router_id* is 192.168.1.11 and the *if_name* is **outside**:

```
pixfirewall# show ospf retransmission-list 192.168.1.11 outside
OSPF Router with ID (192.168.1.12) (Process ID 1)
Neighbor 192.168.1.11, interface outside address 172.16.1.11
Link state retransmission due in 3764 msec, Queue length 2
Type LS ID   ADV RTR   Seq NO Age Checksum
1 192.168.1.12 192.168.1.12 0x80000210 0 0xB196
```

Related Commands

prefix-list	Configures a prefix list to be used for OSPF routing.
route-map	Creates a route map for redistributing routes from one routing protocol to another. Used in configuring OSPF routing on the firewall.

router ospf	Configures global parameters for the OSPF routing processes on the firewall, and enables or disables OSPF routing through the firewall.
routing interface	Configures interface-specific OSPF routing parameters.

show ospf summary-address

Displays a list of all summary address redistribution information configured under an OSPF process.

show ospf summary-address

Syntax Description

summary-address	An address representing multiple (aggregated) addresses.

Defaults

None.

Command Modes

The **show** command is available in privileged mode.

Usage Guidelines

The OSPF routing-related **show** commands are available in privileged command mode on the firewall. You do not need to be in an OSPF configuration subcommand mode to use the OSPF-related **show** commands.

Examples

The following is sample output from the **show ospf summary-address** command for an OSPF process with the *pid* of 5:

```
pixfirewall# show ospf summary-address
OSPF Process 5, Summary-address
10.2.0.0/255.255.0.0 Metric -1, Type 0, Tag 0
10.2.0.0/255.255.0.0 Metric -1, Type 0, Tag 10
```

Related Commands

prefix-list	Configures a prefix list to be used for OSPF routing.
route-map	Creates a route map for redistributing routes from one routing protocol to another. Used in configuring OSPF routing on the firewall.
router ospf	Configures global parameters for the OSPF routing processes on the firewall, and enables or disables OSPF routing through the firewall.
routing interface	Configures interface-specific OSPF routing parameters.

show ospf virtual links

Displays parameters and the current state of OSPF virtual links.

show ospf virtual-links

Syntax Description

virtual-links	OSPF virtual links.

Defaults

None.

Command Modes

The **show ospf virtual-links** command is available in privileged mode.

Usage Guidelines

The OSPF routing-related **show** commands are available in privileged command mode on the firewall. You do not need to be in an OSPF configuration subcommand mode to use the OSPF-related **show** commands.

Examples

The following is sample output from the **show ospf virtual-links** command:

```
pixfirewall# show ospf virtual-links
Virtual Link to router 192.168.101.2 is up
Transit area 0.0.0.1, via interface Ethernet0, Cost of using 10
Transmit Delay is 1 sec, State POINT_TO_POINT
Timer intervals configured, Hello 10, Dead 40, Wait 40, Retransmit 5
Hello due in 0:00:08
Adjacency State FULL
```

Related Commands

prefix-list	Configures a prefix list to be used for OSPF routing.
route-map	Creates a route map for redistributing routes from one routing protocol to another. Used in configuring OSPF routing on the firewall.
router ospf	Configures global parameters for the OSPF routing processes on the firewall, and enables or disables OSPF routing through the firewall.
routing interface	Configures interface-specific OSPF routing parameters.

show processes

Display processes.

show processes

Syntax Description

processes	The processes running on the PIX Firewall.

Command Modes

Privileged mode.

Usage Guidelines

The **show processes** command displays a list of the running processes. Processes are lightweight threads requiring only a few instructions. In the listing, PC is the program counter, SP is the stack pointer, STATE is the address of a thread queue, Runtime is the number of milliseconds that the thread has been running, SBASE is the stack base address, Stack is the current number of bytes used and the total size of the stack, and Process lists the thread's function.

Examples

The following is sample output from the **show processes** command:

```
pixfirewall(config)# show processes

    PC       SP       STATE       Runtime    SBASE     Stack Process
Hsi 001e7de9 0074e3ac 0054c8e0          0 0074d424 3884/4096 arp_timer
Lsi 001ecf55 007f15a4 0054c8e0         10 007f062c 3800/4096 FragDBGC
Lwe 00119af7 009bd7ec 00550040          0 009bc984 3688/4096 dbgtrace
Lwe 003da59d 009bf97c 00545218          0 009bda34 8008/8192 Logger
Hwe 003de658 009c2a74 005454c8          0 009c0afc 8024/8192 tcp_fast
Hwe 003de5d1 009c4b24 005454c8          0 009c2bac 8024/8192 tcp_slow
Lsi 002f8611 00af8e94 0054c8e0          0 00af7f0c 3944/4096 xlate clean
Lsi 002f851f 00af9f34 0054c8e0          0 00af8fbc 3884/4096 uxlate clean
Mwe 002ef7ff 00c6e304 0054c8e0          0 00c6c36c 7908/8192 tcp_intercept_times
Lsi 0042fb65 00d18b5c 0054c8e0          0 00d17bd4 3768/4096 route_process
Hsi 002e0b9c 00d19bec 0054c8e0         10 00d18c84 3780/4096 PIX Garbage Collecr
Hwe 00213ad9 00d2391c 0054c8e0          0 00d1f9b4 16048/16384 isakmp_time_keepr
Lsi 002de91c 00d3cc84 0054c8e0          0 00d3bcfc 3944/4096 perfmon
Mwe 0020b339 00d670b4 0054c8e0          0 00d6513c 7860/8192 IPsec timer handler
Hwe 00391143 00d7b9fc 005668f0          0 00d79ab4 6904/8192 qos_metric_daemon
Mwe 0025d205 00d92594 0054c8e0          0 00d91e2c 1436/2048 IP Background
Lwe 002f0302 00e44ee4 00561c08          0 00e4406c 3704/4096 pix/trace
Lwe 002f051e 00e45f94 00562338          0 00e4511c 3704/4096 pix/tconsole
H*  0011f4ef 0009fefc 0054c8c8       1580 00e4e484 13548/16384 ci/console
Csi 002e923b 00e5348c 0054c8e0          0 00e52534 3432/4096 update_cpu_usage
Hwe 002d63d1 00ef7324 0052bc98          0 00ef349c 15884/16384 uauth_in
Hwe 003dd0e5 00ef9424 00811bf8          0 00ef754c 7896/8192 uauth_thread
Hwe 003f2c62 00efa574 00545818          0 00ef95fc 3960/4096 udp_timer
Hsi 001dfcf2 00efc22c 0054c8e0          0 00efb2b4 3928/4096 557mcfix
Crd 001dfca7 00efd2ec 0054cd58  764174020 00efc364 3688/4096 557poll
Lsi 001dfd5d 00efe38c 0054c8e0          0 00efd414 3700/4096 557timer
Cwe 001e1785 00f1440c 0085b790        770 00f12514 7344/8192 pix/intf0
Mwe 003f29d2 00f154fc 0085a420          0 00f145c4 3896/4096 riprx/0
Msi 0039a3a1 00f1660c 0054c8e0          0 00f15694 3888/4096 riptx/0
Cwe 001e1785 00f1c744 008d0d00          0 00f1a84c 7928/8192 pix/intf1
Mwe 003f29d2 00f1d854 0085a3d8          0 00f1c91c 3896/4096 riprx/1
Msi 0039a3a1 00f1e964 0054c8e0          0 00f1d9ec 3888/4096 riptx/1
Cwe 001ea085 00f24b0c 0071aa6c          0 00f22ba4 8040/8192 pix/intf2
Mwe 003f29d2 00f25bac 0085a390          0 00f24c74 3896/4096 riprx/2
Msi 0039a3a1 00f26cbc 0054c8e0          0 00f25d44 3888/4096 riptx/2
Hwe 003dd379 00f4c3b4 007fd000          0 00f4c10c  300/1024 listen/http1
Mwe 00367556 00f4e60c 0054c8e0          0 00f4c694 7640/8192 Crypto CA
Mrd 002650c9 00f7bf3c 0054c918       4780 00f79fc4 7744/8192 OSPF Router
Mrd 00265869 00f7960c 0054c918       4760 00f78ed4 1608/2048 OSPF Hello
```

show routing

Displays the (non-default) interface-specific routing configuration.

show routing [**interface** *if_name*]

Syntax Description

if_name	The name of the interface for which to display the configuration.

Defaults

None.

Command Modes

The **show routing** command is available in privileged mode.

Usage Guidelines

The OSPF routing-related **show** commands are available in privileged command mode on the firewall. You do not need to be in an OSPF configuration subcommand mode to use the OSPF-related **show** commands.

Examples

The following is sample output from the **show routing** command:

```
pixfirewall# show routing
routing interface outside
  ospf retransmit-interval 15
routing interface inside
  ospf cost 206
```

The following is sample output from the **show routing** [**interface** *if_name*] command:

```
pixfirewall# show routing interface outside
routing interface outside
  ospf retransmit-interval 15
```

Related Commands

prefix-list	Configures a prefix list to be used for OSPF routing.
route-map	Creates a route map for redistributing routes from one routing protocol to another. Used in configuring OSPF routing on the firewall.
router ospf	Configures global parameters for the OSPF routing processes on the firewall, and enables or disables OSPF routing through the firewall.
routing interface	Configures interface-specific OSPF routing parameters.

show running-config

Display the PIX Firewall running configuration.

show running-config

Syntax Description

running-config	The configuration running on the PIX Firewall.

Command Modes

Privileged mode.

Usage Guidelines

The **show running-config** command displays the current running configuration. The keyword **running-config** is used to match the Cisco IOS software command. The **show running-config** command output is the same as the pre-existing PIX Firewall **write terminal** command.

The **running-config** keyword can be used only in the **show running-config** command. It cannot be used with **no** or **clear**, or as a standalone command. If it is, the CLI treats it as a non-supported command. Also, for this reason, when **?**, **no ?**, or **clear ?** are entered, a **running-config** option is not listed in the command list.

Note PIX Device Manager (PDM) commands will appear in your configuration after you use PDM to connect to or configure your PIX Firewall.

Examples

The following is sample output from the **show running-config** command:

```
pixfirewall# show running-config
: Saved
:
PIX Version 6.2(1)
nameif ethernet0 outside security0
nameif ethernet1 inside security100
nameif ethernet2 intf2 security10
enable password 8Ry2YjIyt7RRXU24 encrypted
passwd 2KFQnbNIdI.2KYOU encrypted
hostname pixdoc515
domain-name cisco.com
fixup protocol ftp 21
fixup protocol http 80
fixup protocol h323 h225 1720
fixup protocol h323 ras 1718-1719
fixup protocol ils 389
fixup protocol rsh 514
fixup protocol rtsp 554
fixup protocol smtp 25
fixup protocol snmp 161-162
fixup protocol sqlnet 1521
fixup protocol sip 5060
fixup protocol skinny 2000
names
access-list inside_outbound_nat0_acl permit ip 10.1.3.0 255.255.255.0 10.1.2.0
access-list inside_outbound_nat0_acl permit ip any any
access-list outside_cryptomap_20 permit ip 10.1.3.0 255.255.255.0 10.1.2.0 255.
access-list outside_cryptomap_40 permit ip any any
access-list 101 permit ip any any
pager lines 24
logging on
interface ethernet0 10baset
interface ethernet1 100full
interface ethernet2 100full shutdown
icmp permit any outside
icmp permit any inside
mtu outside 1500
mtu inside 1500
mtu intf2 1500
ip address outside 172.23.59.230 255.255.0.0 pppoe
ip address inside 10.1.3.1 255.255.255.0
ip address intf2 127.0.0.1 255.255.255.0
multicast interface inside
ip audit info action alarm
ip audit attack action alarm
```

```
no failover
failover timeout 0:00:00
failover poll 15
failover ip address outside 0.0.0.0
failover ip address inside 0.0.0.0
failover ip address intf2 0.0.0.0
pdm location 10.1.2.1 255.255.255.255 outside
pdm location 10.1.2.0 255.255.255.0 outside
pdm logging alerts 100
pdm history enable
arp timeout 14400
global (inside) 6 192.168.1.2-192.168.1.3
global (inside) 3 192.168.4.1
nat (inside) 0 access-list inside_outbound_nat0_acl
access-group 101 in interface outside
route outside 0.0.0.0 0.0.0.0 172.23.59.225 1
timeout xlate 3:00:00
timeout conn 1:00:00 half-closed 0:10:00 udp 0:02:00 rpc 0:10:00 h323 0:05:00 s0
timeout uauth 0:05:00 absolute
aaa-server TACACS+ protocol tacacs+
aaa-server RADIUS protocol radius
aaa-server LOCAL protocol local
http server enable
http 0.0.0.0 0.0.0.0 outside
no snmp-server location
no snmp-server contact
snmp-server community public
no snmp-server enable traps
floodguard enable
sysopt connection permit-ipsec
crypto ipsec transform-set ESP-DES-SHA esp-des esp-sha-hmac
crypto map outside_map 20 ipsec-isakmp
crypto map outside_map 20 match address outside_cryptomap_20
crypto map outside_map 20 set peer 172.23.59.231
crypto map outside_map 20 set transform-set ESP-DES-SHA
crypto map outside_map 40 ipsec-isakmp
crypto map outside_map 40 match address outside_cryptomap_40
crypto map outside_map 40 set peer 123.5.5.5
isakmp key ******** address 172.23.59.231 netmask 255.255.255.255 no-xauth no-c
isakmp peer fqdn no-xauth no-config-mode
isakmp policy 20 authentication pre-share
isakmp policy 20 encryption des
isakmp policy 20 hash sha
isakmp policy 20 group 2
isakmp policy 20 lifetime 86400
isakmp policy 40 authentication rsa-sig
isakmp policy 40 encryption 3des
isakmp policy 40 hash sha
isakmp policy 40 group 2
isakmp policy 40 lifetime 86400
telnet timeout 5
ssh timeout 5
console timeout 10
dhcprelay timeout 60
terminal width 80
Cryptochecksum:4d600490f46b5d335c0fbf2eda0015a2
: end
```

Note A configuration error at bootup will cause the cryptochecksum to display all zeros. Perform the **write memory** command, then the **show running-config** command again to diplay the proper checksum.

show startup-config

Display the PIX Firewall startup configuration.

show startup-config

Syntax Description

startup-config	The configuration present at startup on the PIX Firewall.

Command Modes

Privileged mode.

Usage Guidelines

The **show startup-config** command displays the startup configuration of the PIX Firewall. The keyword **startup-config** is used to match the Cisco IOS software command. The **show startup-config** command output is the same as the pre-existing PIX Firewall **show configure** command. The **show startup-config** command is not needed for PDM but is provided for compatibility with Cisco IOS software.

The **startup-config** keyword can be used only in the **show startup-config** command. It cannot be used with **no** or **clear**, or as a standalone command. If it is, the CLI treats it as a non-supported command. Also, for this reason, when **?**, **no ?**, or **clear ?** are entered, a **startup-config** option is not listed in the command list.

Examples

The following is sample output from the **show startup-config** command:

```
pixfirewall# show startup-config
: Saved
: Written by enable_15 at 17:14:09.092 UTC Tue Apr 9 2002
PIX Version 6.2(0)227
nameif ethernet0 outside security0
nameif ethernet1 inside security100
nameif ethernet2 intf2 security10
enable password 8Ry2YjIyt7RRXU24 encrypted
passwd 2KFQnbNIdI.2KYOU encrypted
hostname pixdoc515
domain-name cisco.com
fixup protocol ftp 21
fixup protocol http 80
fixup protocol h323 h225 1720
fixup protocol h323 ras 1718-1719
fixup protocol ils 389
fixup protocol rsh 514
fixup protocol rtsp 554
fixup protocol smtp 25
fixup protocol sqlnet 1521
fixup protocol sip 5060
fixup protocol skinny 2000
names
access-list inside_outbound_nat0_acl permit ip 10.1.3.0 255.255.255.0 10.1.2.0
access-list inside_outbound_nat0_acl permit ip any any
access-list outside_cryptomap_20 permit ip 10.1.3.0 255.255.255.0 10.1.2.0 255.
access-list outside_cryptomap_40 permit ip any any
access-list 101 permit ip any any
pager lines 24
logging on
interface ethernet0 10baset
```

```
interface ethernet1 100full
interface ethernet2 100full shutdown
icmp permit any outside
icmp permit any inside
mtu outside 1500
mtu inside 1500
mtu intf2 1500
ip address outside 172.23.59.230 255.255.0.0 pppoe
ip address inside 10.1.3.1 255.255.255.0
ip address intf2 127.0.0.1 255.255.255.0
multicast interface inside
ip audit info action alarm
ip audit attack action alarm
no failover
failover timeout 0:00:00
failover poll 15
failover ip address outside 0.0.0.0
failover ip address inside 0.0.0.0
failover ip address intf2 0.0.0.0
pdm location 10.1.2.1 255.255.255.255 outside
pdm location 10.1.2.0 255.255.255.0 outside
pdm logging alerts 100
pdm history enable
arp timeout 14400
global (inside) 6 192.168.1.2-192.168.1.3
global (inside) 3 192.168.4.1
nat (inside) 0 access-list inside_outbound_nat0_acl
access-group 101 in interface outside
route outside 0.0.0.0 0.0.0.0 172.23.59.225 1
timeout xlate 3:00:00
timeout conn 1:00:00 half-closed 0:10:00 udp 0:02:00 rpc 0:10:00 h323 0:05:00 s0
timeout uauth 0:05:00 absolute
aaa-server TACACS+ protocol tacacs+
aaa-server RADIUS protocol radius
aaa-server LOCAL protocol local
http server enable
http 0.0.0.0 0.0.0.0 outside
no snmp-server location
no snmp-server contact
snmp-server community public
no snmp-server enable traps
floodguard enable
sysopt connection permit-ipsec
crypto ipsec transform-set ESP-DES-SHA esp-des esp-sha-hmac
crypto map outside_map 20 ipsec-isakmp
crypto map outside_map 20 match address outside_cryptomap_20
crypto map outside_map 20 set peer 172.23.59.231
crypto map outside_map 20 set transform-set ESP-DES-SHA
crypto map outside_map 40 ipsec-isakmp
crypto map outside_map 40 match address outside_cryptomap_40
crypto map outside_map 40 set peer 123.5.5.5
isakmp key ******** address 172.23.59.231 netmask 255.255.255.255 no-xauth no-c
isakmp peer fqdn no-xauth no-config-mode
isakmp policy 20 authentication pre-share
isakmp policy 20 encryption des
isakmp policy 20 hash sha
isakmp policy 20 group 2
isakmp policy 20 lifetime 86400
isakmp policy 40 authentication rsa-sig
isakmp policy 40 encryption 3des
isakmp policy 40 hash sha
isakmp policy 40 group 2
isakmp policy 40 lifetime 86400
telnet timeout 5
```

```
ssh timeout 5
```

show tech-support

View information to help a support analyst.

show tech-support [**no-config**]

Syntax Description

no-config	Excludes the output of the running configuration.
tech-support	The data used for diagnosis by technical support analysts.

Command Modes

Privileged mode.

Usage Guidelines

The **show tech-support** command lists information that technical support analysts need to help you diagnose PIX Firewall problems. This command combines the output from the **show** commands that provide the most information to a technical support analyst.

Examples

The following is sample output from the **show tech-support no-config** command, which excludes the running configuration:

```
pixfirewall(config)# show tech-support no-config

Cisco PIX Firewall Version 6.3(1)
Cisco PIX Device Manager Version 2.1(1)

Compiled on Fri 15-Nov-02 14:35 by root

pixfirewall up 2 days 8 hours

Hardware:   PIX-515, 64 MB RAM, CPU Pentium 200 MHz
Flash i28F640J5 @ 0x300, 16MB
BIOS Flash AT29C257 @ 0xfffd8000, 32KB

0: ethernet0: address is 0003.e300.73fd, irq 10
1: ethernet1: address is 0003.e300.73fe, irq 7
2: ethernet2: address is 00d0.b7c8.139e, irq 9
Licensed Features:
Failover:           Disabled
VPN-DES:            Enabled
VPN-3DES-AES:       Disabled
Maximum Interfaces: 3
Cut-through Proxy:  Enabled
Guards:             Enabled
URL-filtering:      Enabled
Inside Hosts:       Unlimited
Throughput:         Unlimited
IKE peers:          Unlimited

This PIX has a Restricted (R) license.

Serial Number: 480430455 (0x1ca2c977)
Running Activation Key: 0xc2e94182 0xc21d8206 0x15353200 0x633f6734
Configuration last modified by enable_15 at 23:05:24.264 UTC Sat Nov 16 2002
```

```
------------------ show clock ------------------

00:08:14.911 UTC Sun Nov 17 2002

------------------ show memory ------------------

Free memory:        50708168 bytes
Used memory:        16400696 bytes
-------------     ----------------
Total memory:       67108864 bytes

------------------ show conn count ------------------

0 in use, 0 most used

------------------ show xlate count ------------------

0 in use, 0 most used

------------------ show blocks ------------------

  SIZE    MAX    LOW    CNT
     4   1600   1600   1600
    80    400    400    400
   256    500    499    500
  1550   1188    795    919

------------------ show interface ------------------

interface ethernet0 "outside" is up, line protocol is up
  Hardware is i82559 ethernet, address is 0003.e300.73fd
  IP address 172.23.59.232, subnet mask 255.255.0.0
  MTU 1500 bytes, BW 10000 Kbit half duplex
        1267 packets input, 185042 bytes, 0 no buffer
        Received 1248 broadcasts, 0 runts, 0 giants
        0 input errors, 0 CRC, 0 frame, 0 overrun, 0 ignored, 0 abort
        20 packets output, 1352 bytes, 0 underruns
        0 output errors, 0 collisions, 0 interface resets
        0 babbles, 0 late collisions, 9 deferred
        0 lost carrier, 0 no carrier
        input queue (curr/max blocks): hardware (13/128) software (0/2)
        output queue (curr/max blocks): hardware (0/1) software (0/1)
interface ethernet1 "inside" is up, line protocol is down
  Hardware is i82559 ethernet, address is 0003.e300.73fe
  IP address 10.1.1.1, subnet mask 255.255.255.0
  MTU 1500 bytes, BW 10000 Kbit half duplex
        0 packets input, 0 bytes, 0 no buffer
        Received 0 broadcasts, 0 runts, 0 giants
        0 input errors, 0 CRC, 0 frame, 0 overrun, 0 ignored, 0 abort
        1 packets output, 60 bytes, 0 underruns
        0 output errors, 0 collisions, 0 interface resets
        0 babbles, 0 late collisions, 0 deferred
        1 lost carrier, 0 no carrier
        input queue (curr/max blocks): hardware (128/128) software (0/0)
        output queue (curr/max blocks): hardware (0/1) software (0/1)
interface ethernet2 "intf2" is administratively down, line protocol is down
  Hardware is i82559 ethernet, address is 00d0.b7c8.139e
  IP address 127.0.0.1, subnet mask 255.255.255.255
  MTU 1500 bytes, BW 10000 Kbit half duplex
        0 packets input, 0 bytes, 0 no buffer
        Received 0 broadcasts, 0 runts, 0 giants
        0 input errors, 0 CRC, 0 frame, 0 overrun, 0 ignored, 0 abort
        0 packets output, 0 bytes, 0 underruns
        0 output errors, 0 collisions, 0 interface resets
```

```
        0 babbles, 0 late collisions, 0 deferred
        0 lost carrier, 0 no carrier
        input queue (curr/max blocks): hardware (128/128) software (0/0)
        output queue (curr/max blocks): hardware (0/0) software (0/0)

------------------ show cpu usage ------------------

CPU utilization for 5 seconds = 0%; 1 minute: 0%; 5 minutes: 0%

------------------ show process ------------------


    PC       SP       STATE       Runtime    SBASE     Stack Process
Hsi 001e3329 00763e7c 0053e5c8           0 00762ef4 3784/4096 arp_timer
Lsi 001e80e9 00807074 0053e5c8           0 008060fc 3832/4096 FragDBGC
Lwe 00117e3a 009dc2e4 00541d18           0 009db46c 3704/4096 dbgtrace
Lwe 003cee95 009de464 00537718           0 009dc51c 8008/8192 Logger
Hwe 003d2d18 009e155c 005379c8           0 009df5e4 8008/8192 tcp_fast
Hwe 003d2c91 009e360c 005379c8           0 009e1694 8008/8192 tcp_slow
Lsi 002ec97d 00b1a464 0053e5c8           0 00b194dc 3928/4096 xlate clean
Lsi 002ec88b 00b1b504 0053e5c8           0 00b1a58c 3888/4096 uxlate clean
Mwe 002e3a17 00c8f8d4 0053e5c8           0 00c8d93c 7908/8192 tcp_intercept_times
Lsi 00423dd5 00d3a22c 0053e5c8           0 00d392a4 3900/4096 route_process
Hsi 002d59fc 00d3b2bc 0053e5c8           0 00d3a354 3780/4096 PIX Garbage Collecr
Hwe 0020e301 00d5957c 0053e5c8           0 00d55614 16048/16384 isakmp_time_keepr
Lsi 002d377c 00d7292c 0053e5c8           0 00d719a4 3928/4096 perfmon
Hwe 0020bd07 00d9c12c 0050bb90           0 00d9b1c4 3944/4096 IPSec
Mwe 00205e25 00d9e1ec 0053e5c8           0 00d9c274 7860/8192 IPsec timer handler
Hwe 003864e3 00db26bc 00557920           0 00db0764 6952/8192 qos_metric_daemon
Mwe 00255a65 00dc9244 0053e5c8           0 00dc8adc 1436/2048 IP Background
Lwe 002e450e 00e7bb94 00552c30           0 00e7ad1c 3704/4096 pix/trace
Lwe 002e471e 00e7cc44 00553368           0 00e7bdcc 3704/4096 pix/tconsole
Hwe 001e5368 00e7ed44 00730674           0 00e7ce9c 7228/8192 pix/intf0
Hwe 001e5368 00e80e14 007305d4           0 00e7ef6c 7228/8192 pix/intf1
Hwe 001e5368 00e82ee4 00730534        2470 00e8103c 4892/8192 pix/intf2
H*  0011d7f7 0009ff2c 0053e5b0         780 00e8511c 13004/16384 ci/console
Csi 002dd8ab 00e8a124 0053e5c8           0 00e891cc 3396/4096 update_cpu_usage
Hwe 002cb4d1 00f2bfbc 0051e360           0 00f2a134 7692/8192 uauth_in
Hwe 003d17d1 00f2e0bc 00828cf0           0 00f2c1e4 7896/8192 uauth_thread
Hwe 003e71d4 00f2f20c 00537d20           0 00f2e294 3960/4096 udp_timer
Hsi 001db3ca 00f30fc4 0053e5c8           0 00f3004c 3784/4096 557mcfix
Crd 001db37f 00f32084 0053ea40   121094970 00f310fc 3744/4096 557poll
Lsi 001db435 00f33124 0053e5c8           0 00f321ac 3700/4096 557timer
Hwe 001e5398 00f441dc 008121e0           0 00f43294 3912/4096 fover_ip0
Cwe 001dcdad 00f4523c 00872b48          20 00f44344 3528/4096 ip/0:0
Hwe 001e5398 00f4633c 008121bc           0 00f453f4 3532/4096 icmp0
Hwe 001e5398 00f47404 00812198           0 00f464cc 3896/4096 udp_thread/0
Hwe 001e5398 00f4849c 00812174           0 00f475a4 3832/4096 tcp_thread/0
Hwe 001e5398 00f495bc 00812150           0 00f48674 3912/4096 fover_ip1
Cwe 001dcdad 00f4a61c 008ea850           0 00f49724 3832/4096 ip/1:1
Hwe 001e5398 00f4b71c 0081212c           0 00f4a7d4 3912/4096 icmp1
Hwe 001e5398 00f4c7e4 00812108           0 00f4b8ac 3896/4096 udp_thread/1
Hwe 001e5398 00f4d87c 008120e4           0 00f4c984 3832/4096 tcp_thread/1
Hwe 001e5398 00f4e99c 008120c0           0 00f4da54 3912/4096 fover_ip2
Cwe 001e542d 00f4fa6c 00730534           0 00f4eb04 3944/4096 ip/2:2
Hwe 001e5398 00f50afc 0081209c           0 00f4fbb4 3912/4096 icmp2
Hwe 001e5398 00f51bc4 00812078           0 00f50c8c 3896/4096 udp_thread/2
Hwe 001e5398 00f52c5c 00812054           0 00f51d64 3832/4096 tcp_thread/2
Hwe 003d1a65 00f78284 008140f8           0 00f77fdc  300/1024 listen/http1
Mwe 0035cafa 00f7a63c 0053e5c8           0 00f786c4 7640/8192 Crypto CA

------------------ show failover ------------------

No license for Failover
```

```
------------------ show traffic ------------------

outside:
        received (in 205213.390 secs):
                1267 packets    185042 bytes
                0 pkts/sec      0 bytes/sec
        transmitted (in 205213.390 secs):
                20 packets      1352 bytes
                0 pkts/sec      0 bytes/sec
inside:
        received (in 205215.800 secs):
                0 packets       0 bytes
                0 pkts/sec      0 bytes/sec
        transmitted (in 205215.800 secs):
                1 packets       60 bytes
                0 pkts/sec      0 bytes/sec
intf2:
        received (in 205215.810 secs):
                0 packets       0 bytes
                0 pkts/sec      0 bytes/sec
        transmitted (in 205215.810 secs):
                0 packets       0 bytes
                0 pkts/sec      0 bytes/sec

------------------ show perfmon ------------------


PERFMON STATS:    Current      Average
Xlates               0/s          0/s
Connections          0/s          0/s
TCP Conns            0/s          0/s
UDP Conns            0/s          0/s
URL Access           0/s          0/s
URL Server Req       0/s          0/s
TCP Fixup            0/s          0/s
TCPIntercept         0/s          0/s
HTTP Fixup           0/s          0/s
FTP Fixup            0/s          0/s
AAA Authen           0/s          0/s
AAA Author           0/s          0/s
AAA Account          0/s          0/s
```

The following is sample output from the **show tech-support** command, which includes the running configuration:

```
pixfirewall(config)# show tech-support

Cisco PIX Firewall Version 6.3(1)
Cisco PIX Device Manager Version 2.1(1)

Compiled on Fri 15-Nov-02 14:35 by root

pixfirewall up 2 days 9 hours

Hardware:   PIX-515, 64 MB RAM, CPU Pentium 200 MHz
Flash i28F640J5 @ 0x300, 16MB
BIOS Flash AT29C257 @ 0xfffd8000, 32KB

0: ethernet0: address is 0003.e300.73fd, irq 10
1: ethernet1: address is 0003.e300.73fe, irq 7
2: ethernet2: address is 00d0.b7c8.139e, irq 9
Licensed Features:
Failover:           Disabled
```

```
VPN-DES:            Enabled
VPN-3DES-AES:       Disabled
Maximum Interfaces: 3
Cut-through Proxy:  Enabled
Guards:             Enabled
URL-filtering:      Enabled
Inside Hosts:       Unlimited
Throughput:         Unlimited
IKE peers:          Unlimited

This PIX has a Restricted (R) license.

Serial Number: 480430455 (0x1ca2c977)
Running Activation Key: 0xc2e94182 0xc21d8206 0x15353200 0x633f6734
Configuration last modified by enable_15 at 23:05:24.264 UTC Sat Nov 16 2002

------------------ show clock ------------------

00:08:39.591 UTC Sun Nov 17 2002

------------------ show memory ------------------

Free memory:        50708168 bytes
Used memory:        16400696 bytes
-------------     ----------------
Total memory:       67108864 bytes

------------------ show conn count ------------------

0 in use, 0 most used

------------------ show xlate count ------------------

0 in use, 0 most used

------------------ show blocks ------------------

  SIZE    MAX    LOW    CNT
     4   1600   1600   1600
    80    400    400    400
   256    500    499    500
  1550   1188    795    919

------------------ show interface ------------------

interface ethernet0 "outside" is up, line protocol is up
  Hardware is i82559 ethernet, address is 0003.e300.73fd
  IP address 172.23.59.232, subnet mask 255.255.0.0
  MTU 1500 bytes, BW 10000 Kbit half duplex
        1267 packets input, 185042 bytes, 0 no buffer
        Received 1248 broadcasts, 0 runts, 0 giants
        0 input errors, 0 CRC, 0 frame, 0 overrun, 0 ignored, 0 abort
        20 packets output, 1352 bytes, 0 underruns
        0 output errors, 0 collisions, 0 interface resets
        0 babbles, 0 late collisions, 9 deferred
        0 lost carrier, 0 no carrier
        input queue (curr/max blocks): hardware (13/128) software (0/2)
        output queue (curr/max blocks): hardware (0/1) software (0/1)
interface ethernet1 "inside" is up, line protocol is down
  Hardware is i82559 ethernet, address is 0003.e300.73fe
  IP address 10.1.1.1, subnet mask 255.255.255.0
  MTU 1500 bytes, BW 10000 Kbit half duplex
        0 packets input, 0 bytes, 0 no buffer
        Received 0 broadcasts, 0 runts, 0 giants
```

```
        0 input errors, 0 CRC, 0 frame, 0 overrun, 0 ignored, 0 abort
        1 packets output, 60 bytes, 0 underruns
        0 output errors, 0 collisions, 0 interface resets
        0 babbles, 0 late collisions, 0 deferred
        1 lost carrier, 0 no carrier
        input queue (curr/max blocks): hardware (128/128) software (0/0)
        output queue (curr/max blocks): hardware (0/1) software (0/1)
interface ethernet2 "intf2" is administratively down, line protocol is down
  Hardware is i82559 ethernet, address is 00d0.b7c8.139e
  IP address 127.0.0.1, subnet mask 255.255.255.255
  MTU 1500 bytes, BW 10000 Kbit half duplex
        0 packets input, 0 bytes, 0 no buffer
        Received 0 broadcasts, 0 runts, 0 giants
        0 input errors, 0 CRC, 0 frame, 0 overrun, 0 ignored, 0 abort
        0 packets output, 0 bytes, 0 underruns
        0 output errors, 0 collisions, 0 interface resets
        0 babbles, 0 late collisions, 0 deferred
        0 lost carrier, 0 no carrier
        input queue (curr/max blocks): hardware (128/128) software (0/0)
        output queue (curr/max blocks): hardware (0/0) software (0/0)

------------------ show cpu usage ------------------

CPU utilization for 5 seconds = 0%; 1 minute: 0%; 5 minutes: 0%

------------------ show process ------------------


    PC       SP       STATE       Runtime    SBASE     Stack Process
Hsi 001e3329 00763e7c 0053e5c8          0 00762ef4 3784/4096 arp_timer
Lsi 001e80e9 00807074 0053e5c8          0 008060fc 3832/4096 FragDBGC
Lwe 00117e3a 009dc2e4 00541d18          0 009db46c 3704/4096 dbgtrace
Lwe 003cee95 009de464 00537718          0 009dc51c 8008/8192 Logger
Hwe 003d2d18 009e155c 005379c8          0 009df5e4 8008/8192 tcp_fast
Hwe 003d2c91 009e360c 005379c8          0 009e1694 8008/8192 tcp_slow
Lsi 002ec97d 00b1a464 0053e5c8          0 00b194dc 3928/4096 xlate clean
Lsi 002ec88b 00b1b504 0053e5c8          0 00b1a58c 3888/4096 uxlate clean
Mwe 002e3a17 00c8f8d4 0053e5c8          0 00c8d93c 7908/8192 tcp_intercept_times
Lsi 00423dd5 00d3a22c 0053e5c8          0 00d392a4 3900/4096 route_process
Hsi 002d59fc 00d3b2bc 0053e5c8          0 00d3a354 3780/4096 PIX Garbage Collecr
Hwe 0020e301 00d5957c 0053e5c8          0 00d55614 16048/16384 isakmp_time_keepr
Lsi 002d377c 00d7292c 0053e5c8          0 00d719a4 3928/4096 perfmon
Hwe 0020bd07 00d9c12c 0050bb90          0 00d9b1c4 3944/4096 IPSec
Mwe 00205e25 00d9e1ec 0053e5c8          0 00d9c274 7860/8192 IPsec timer handler
Hwe 003864e3 00db26bc 00557920          0 00db0764 6952/8192 qos_metric_daemon
Mwe 00255a65 00dc9244 0053e5c8          0 00dc8adc 1436/2048 IP Background
Lwe 002e450e 00e7bb94 00552c30          0 00e7ad1c 3704/4096 pix/trace
Lwe 002e471e 00e7cc44 00553368          0 00e7bdcc 3704/4096 pix/tconsole
Hwe 001e5368 00e7ed44 00730674          0 00e7ce9c 7228/8192 pix/intf0
Hwe 001e5368 00e80e14 007305d4          0 00e7ef6c 7228/8192 pix/intf1
Hwe 001e5368 00e82ee4 00730534       2470 00e8103c 4892/8192 pix/intf2
H*  0011d7f7 0009ff2c 0053e5b0        950 00e8511c 13004/16384 ci/console
Csi 002dd8ab 00e8a124 0053e5c8          0 00e891cc 3396/4096 update_cpu_usage
Hwe 002cb4d1 00f2bfbc 0051e360          0 00f2a134 7692/8192 uauth_in
Hwe 003d17d1 00f2e0bc 00828cf0          0 00f2c1e4 7896/8192 uauth_thread
Hwe 003e71d4 00f2f20c 00537d20          0 00f2e294 3960/4096 udp_timer
Hsi 001db3ca 00f30fc4 0053e5c8          0 00f3004c 3784/4096 557mcfix
Crd 001db37f 00f32084 0053ea40  121109610 00f310fc 3744/4096 557poll
Lsi 001db435 00f33124 0053e5c8          0 00f321ac 3700/4096 557timer
Hwe 001e5398 00f441dc 008121e0          0 00f43294 3912/4096 fover_ip0
Cwe 001dcdad 00f4523c 00872b48         20 00f44344 3528/4096 ip/0:0
Hwe 001e5398 00f4633c 008121bc          0 00f453f4 3532/4096 icmp0
Hwe 001e5398 00f47404 00812198          0 00f464cc 3896/4096 udp_thread/0
Hwe 001e5398 00f4849c 00812174          0 00f475a4 3832/4096 tcp_thread/0
```

```
Hwe 001e5398 00f495bc 00812150          0 00f48674 3912/4096 fover_ip1
Cwe 001dcdad 00f4a61c 008ea850          0 00f49724 3832/4096 ip/1:1
Hwe 001e5398 00f4b71c 0081212c          0 00f4a7d4 3912/4096 icmp1
Hwe 001e5398 00f4c7e4 00812108          0 00f4b8ac 3896/4096 udp_thread/1
Hwe 001e5398 00f4d87c 008120e4          0 00f4c984 3832/4096 tcp_thread/1
Hwe 001e5398 00f4e99c 008120c0          0 00f4da54 3912/4096 fover_ip2
Cwe 001e542d 00f4fa6c 00730534          0 00f4eb04 3944/4096 ip/2:2
Hwe 001e5398 00f50afc 0081209c          0 00f4fbb4 3912/4096 icmp2
Hwe 001e5398 00f51bc4 00812078          0 00f50c8c 3896/4096 udp_thread/2
Hwe 001e5398 00f52c5c 00812054          0 00f51d64 3832/4096 tcp_thread/2
Hwe 003d1a65 00f78284 008140f8          0 00f77fdc  300/1024 listen/http1
Mwe 0035cafa 00f7a63c 0053e5c8          0 00f786c4 7640/8192 Crypto CA

------------------ show failover ------------------

No license for Failover

------------------ show traffic ------------------

outside:
        received (in 205238.740 secs):
                1267 packets    185042 bytes
                0 pkts/sec      0 bytes/sec
        transmitted (in 205238.740 secs):
                20 packets      1352 bytes
                0 pkts/sec      0 bytes/sec
inside:
        received (in 205242.200 secs):
                0 packets       0 bytes
                0 pkts/sec      0 bytes/sec
        transmitted (in 205242.200 secs):
                1 packets       60 bytes
                0 pkts/sec      0 bytes/sec
intf2:
        received (in 205242.200 secs):
                0 packets       0 bytes
                0 pkts/sec      0 bytes/sec
        transmitted (in 205242.200 secs):
                0 packets       0 bytes
                0 pkts/sec      0 bytes/sec

------------------ show perfmon ------------------


PERFMON STATS:    Current      Average
Xlates               0/s          0/s
Connections          0/s          0/s
TCP Conns            0/s          0/s
UDP Conns            0/s          0/s
URL Access           0/s          0/s
URL Server Req       0/s          0/s
TCP Fixup            0/s          0/s
TCPIntercept         0/s          0/s
HTTP Fixup           0/s          0/s
FTP Fixup            0/s          0/s
AAA Authen           0/s          0/s
AAA Author           0/s          0/s
AAA Account          0/s          0/s

------------------ show running-config ------------------

: Saved
:
PIX Version 6.3(1)
```

```
interface ethernet0 auto
interface ethernet1 auto
interface ethernet2 auto shutdown
nameif ethernet0 outside security0
nameif ethernet1 inside security100
nameif ethernet2 intf2 security10
enable password 8Ry2YjIyt7RRXU24 encrypted
passwd 2KFQnbNIdI.2KYOU encrypted
hostname pixfirewall
domain-name cisco.com
fixup protocol ftp 21
fixup protocol http 80
fixup protocol h323 h225 1720
fixup protocol h323 ras 1718-1719
fixup protocol ils 389
fixup protocol rsh 514
fixup protocol rtsp 554
fixup protocol smtp 25
fixup protocol sqlnet 1521
fixup protocol sip 5060
fixup protocol skinny 2000
fixup protocol sip udp 5060
names
access-list 101 permit tcp any host 10.1.1.3 eq www
access-list 101 permit tcp any host 10.1.1.3 eq smtp
pager lines 24
mtu outside 1500
mtu inside 1500
mtu intf2 1500
ip address outside 172.23.59.232 255.255.0.0
ip address inside 10.1.1.1 255.255.255.0
ip address intf2 127.0.0.1 255.255.255.255
ip audit info action alarm
ip audit attack action alarm
pdm history enable
arp timeout 14400
global (outside) 1 interface
nat (inside) 1 0.0.0.0 0.0.0.0 0 0
route-map maptag1 permit 8
  set metric 5
  set metric-type type-2
  match metric 5
route outside 0.0.0.0 0.0.0.0 172.23.59.225 1
timeout xlate 3:00:00
timeout conn 1:00:00 half-closed 0:10:00 udp 0:02:00 rpc 0:10:00 h225 1:00:00
timeout h323 0:05:00 mgcp 0:05:00 sip 0:30:00 sip_media 0:02:00
timeout uauth 0:05:00 absolute
aaa-server TACACS+ protocol tacacs+
aaa-server RADIUS protocol radius
aaa-server LOCAL protocol local
http server enable
http 10.1.1.2 255.255.255.255 inside
no snmp-server location
no snmp-server contact
snmp-server community public
no snmp-server enable traps
floodguard enable
telnet timeout 5
ssh timeout 5
console timeout 0
terminal width 80
banner exec working...
banner motd Haveagoodday
```

```
Cryptochecksum:00000000000000000000000000000000
: end
```

show tcpstat

Displays the status of the firewall TCP stack and the TCP connections terminated on the firewall (for debugging).

show tcpstat

Syntax Description

tcpstat	TCP connection statistics.

Defaults

None.

Command Modes

The **show tcpstat** command is available in privileged mode.

Usage Guidelines

The **show tcpstat** command displays the status of the TCP stack and TCP connections terminated on the firewall. The TCP statistics displayed are described in Table 8-4:

Table 8-4 TCP Statistics in the show tcpstat Command

Statistic	Description
tcb_cnt	The number of TCP users.
proxy_cnt	The number of TCP proxies. TCP proxies are used by user authorization.
tcp_xmt pkts	The number of packets that were transmitted by the TCP stack.
tcp_rcv good pkts	The number of good packets that were received by the TCP stack.
tcp_rcv drop pkts	The number of received packets that the TCP stack dropped.
tcp bad chksum	The number of received packets that had a bad checksum.
tcp user hash add	The number of TCP users that were added to the hash table.
tcp user hash add dup	The number of times a TCP user was already in the hash table when trying to add a new user.
tcp user srch hash hit	The number of times a TCP user was found in the hash table when searching.
tcp user srch hash miss	The number of times a TCP user was not found in the hash table when searching.
tcp user hash delete	The number of times a TCP user was deleted from the hash table.

Table 8-4 TCP Statistics in the show tcpstat Command (continued)

Statistic	Description
tcp user hash delete miss	The number of times a TCP user was not found in the hash table when trying to delete the user.
lip	The local IP address of the TCP user.
fip	The foreign IP address of the TCP user.
lp	The local port of the TCP user.
fp	The foreign port of the TCP user.
st	The state (see RFC 793) of the TCP user. The possible values are as follows: 1 CLOSED 2 LISTEN 3 SYN_SENT 4 SYN_RCVD 5 ESTABLISHED 6 FIN_WAIT_1 7 FIN_WAIT_2 8 CLOSE_WAIT 9 CLOSING 10 LAST_ACK 11 TIME_WAIT
rexqlen	The length of the retransmit queue of the TCP user.
inqlen	The length of the input queue of the TCP user.
tw_timer	The value of the time_wait timer (in milliseconds) of the TCP user.
to_timer	The value of the inactivity timeout timer (in milliseconds) of the TCP user.
cl_timer	The value of the close request timer (in milliseconds) of the TCP user.
per_timer	The value of the persist timer (in milliseconds) of the TCP user.
rt_timer	The value of the retransmit timer (in milliseconds) of the TCP user.
tries	The retransmit count of the TCP user.

Examples

The following example shows the output from the **show tcpstat** command:

```
pixfirewall(config)# show tcpstat
                CURRENT MAX     TOTAL
tcb_cnt         2       12      320
proxy_cnt       0       0       160

tcp_xmt pkts = 540591
tcp_rcv good pkts = 6583
tcp_rcv drop pkts = 2
tcp bad chksum = 0
tcp user hash add = 2028
tcp user hash add dup = 0
tcp user srch hash hit = 316753
```

```
tcp user srch hash miss = 6663
tcp user hash delete = 2027
tcp user hash delete miss = 0

lip = 172.23.59.230 fip = 10.21.96.254 lp = 443 fp = 2567 st
= 4 rexqlen = 0
in0
  tw_timer = 0 to_timer = 179000 cl_timer = 0 per_timer = 0
rt_timer = 0
tries 0
```

Related Commands

show conn	Displays all active connections.

show traffic/clear traffic

Shows interface transmit and receive activity.

clear traffic

show traffic

Syntax Description

traffic	The packets and bytes moving through an interface.

Command Modes

Privileged mode.

Usage Guidelines

The **show traffic** command lists the number of packets and bytes moving through each interface. The number of seconds is the duration the PIX Firewall has been online since the last reboot. The **clear traffic** command clears counters for the **show traffic** command output.

Examples

The following is sample output from the **show traffic** command:

```
show traffic
outside:
        received (in 3786 secs):
                97 packets      6191 bytes
                42 pkts/sec     1 bytes/sec
        transmitted (in 3786 secs):
                99 packets      10590 bytes
                0 pkts/sec      2 bytes/sec ...
```

show uauth/clear uauth

Display or delete all authorization caches for a user.

clear uauth [*username*]

show uauth [*username*]

Syntax Description

username	Clear or view user authentication information by username.

Command Modes

Privileged mode.

Usage Guidelines

The **show uauth** command displays one or all currently authenticated users, the host IP to which they are bound, and, if applicable, any cached IP and port authorization information.

The **clear uauth** command deletes one user's, or all users, AAA authorization and authentication caches, which forces the user or users to reauthenticate the next time they create a connection. The **show uauth** command also lists CiscoSecure 2.1 and later idletime and timeout values, which can be set for different user groups.

This command is used in conjunction with the **timeout** command.

Each user host's IP address has an authorization cache attached to it. If the user attempts to access a service that has been cached from the correct host, the firewall considers it preauthorized and immediately proxies the connection. This means that once you are authorized to access a website, for example, the authorization server is not contacted for each of the images as they are loaded (assuming they come from the same IP address). This significantly increases performance and reduces load on the authorization server.

The cache allows up to 16 address and service pairs for each user host.

The output from the **show uauth** command displays the username provided to the authorization server for authentication and authorization purposes, the IP address that the username is bound to, and whether the user is authenticated only, or has cached services.

Note Normally, when Xauth is enabled, an entry is added to the uauth table (as shown by the **show uauth/clear uauth** command) for the IP address assigned to the client. However, when using Xauth with the Easy VPN Remote feature in Network Extension Mode, the IPSec tunnel is created from network-to-network, so the users behind the firewall cannot be associated with a single IP address. For this reason, a uauth entry cannot be created upon completion of Xauth. If AAA authorization or accounting services are required, you can enable the AAA authentication proxy to authenticate users behind the firewall. For more information on AAA authentication proxies, please refer to the **aaa** commands.

Use the **timeout uauth** command to specify how long the cache should be kept after the user connections become idle. Use the **clear uauth** command to delete all authorization caches for all users, which will cause them to have to reauthenticate the next time they create a connection.

Examples

The following is sample output from the **show uauth** command when no users are authenticated and one user authentication is in progress:

```
pixfirewall(config)# show uauth
                        Current    Most Seen
Authenticated Users       0          0
Authen In Progress        0          1
```

The following is sample output from the **show uauth** command when three users are authenticated and authorized to use services through the PIX Firewall:

```
pixfirewall(config)# show uauth
user 'pat' from 209.165.201.2 authenticated
user 'robin' from 209.165.201.4 authorized to:
  port 192.168.67.34/telnet  192.168.67.11/http  192.168.67.33/tcp/8001
  192.168.67.56/tcp/25  192.168.67.42/ftp
user 'terry' from 209.165.201.7 authorized to:
  port 192.168.1.50/http  209.165.201.8/http
```

In this example, Pat has authenticated with the server but has not completed authorization. Robin has preauthorized connections to the Telnet, Web (HTTP), sendmail, FTP services, and to TCP port 8001 on 192.168.67.33.

Terry has been browsing the Web and is authorized for Web browsing to the two sites shown.

The next example causes Pat to reauthenticate:

```
clear uauth pat
```

Related Commands

aaa authorization	Enable or disable LOCAL or TACACS+ user authorization services.
timeout	Sets the maximum idle times.

show version

View the PIX Firewall operating information.

show version

Syntax Description

version	The PIX Firewall software version, hardware configuration, license key, and related uptime data.

Command Modes

Unprivileged mode.

Usage Guidelines

The **show version** command displays the PIX Firewall unit's software version, operating time since last reboot, processor type, Flash memory type, interface boards, serial number (BIOS ID), activation key value, license type (R or UR), and timestamp for when the configuration was last modified.

The serial number listed with the **show version** command in PIX Firewall software Version 5.3 and higher is for the Flash memory BIOS. This number is different from the serial number on the chassis. When you get a software upgrade, you will need the serial number that appears in the **show version** command, not the chassis number.

For PIX Firewall software Version 6.3(2) and higher, the **show version** command shows the maximum number of physical interfaces as well as the maximum number of logical interfaces for use with VLANs.

For PIX Firewall software Version 6.2 and higher, the **show version** command output appears as follows:

```
Running Activation Key: activation-key-four-tuple
```

to indicate the activation key that is currently running PIX Firewall image.

The amount of Flash memory is indicated at the end of the line showing the version of Flash installed on the PIX Firewall.

Throughput Limited indicates that the speed of the PIX Firewall interface is limited due to platform or version restrictions. ISAKMP peers Limited indicates that the number of IPSec peers is limited due to platform restrictions.

Note The uptime value indicates how long a failover set has been running. If one unit stops running, the uptime value will continue to increase as long as the other unit continues to operate.

Examples

The following is sample output from the **version** command:

```
pixfirewall(config)# show version

Cisco PIX Firewall Version 6.3(1)
Cisco PIX Device Manager Version 3.0(1)

Compiled on Wed 06-Nov-02 11:22 by root

pixfirewall up 4 days 22 hours

Hardware:   PIX-515E, 64 MB RAM, CPU Pentium 200 MHz
Flash i28F640J5 @ 0x300, 16MB
BIOS Flash AT29C257 @ 0xfffd8000, 32KB

0: ethernet0: address is 0003.e300.73fd, irq 10
1: ethernet1: address is 0003.e300.73fe, irq 7
2: ethernet2: address is 00d0.b7c8.139e, irq 9
Licensed Features:
Failover: Disabled
VPN-DES: Enabled
VPN-3DES-AES:       Disabled
Maximum Physical Interfaces: 6
Maximum Interfaces: 10
Cut-through Proxy:  Enabled
Guards:             Enabled
URL-filtering:      Enabled
Inside Hosts:       Unlimited
Throughput:         Unlimited
IKE peers:          Unlimited

This PIX has a Restricted (R) license.

Serial Number: 480430455 (0x1ca2c977)
```

```
Running Activation Key: 0xc2e94182 0xc21d8206 0x15353200 0x633f6734
Configuration last modified by enable_15 at 16:36:30.480 UTC Mon Nov 11 2002
```

Note The output of the **show version** command indicates whether the PIX Firewall has a Restricted (R) or Unrestricted (UR) license. A PIX Firewall with an R license cannot be used in a failover pair, and it has one half as much RAM as a PIX Firewall of the same platform with a UR license. Also, a PIX Firewall with an R license supports fewer physical interfaces and fewer logical interfaces (VLANs) than the same platform with a UR license. The number of interfaces allowed varies by platform.

show xlate/clear xlate

View or clear translation slot information.

clear xlate [**global** | **local** *ip1*[**-***ip2*] [**netmask** *mask*]] **lport** | **gport** *port*[**-***port*]] [**interface** *if1*[**,***if2*][**,***ifn*]] [**state static** [**,dump**] [**,portmap**] [**,norandomseq**] [**,identity**]]

show xlate [**detail**] [**global** | **local** *ip1* [**-***ip2*] [**netmask** *mask*]] **lport** | **gport** *port* [**-***port*]] [**interface** *if1* [**,***if2*] [**,***ifn*]] [**state static** [**,dump**] [**,portmap**] [**,norandomseq**] [**,identity**]] [**debug**] [**count**]

Syntax Description

detail	If specified, displays translation type and interface information.
[**global** | **local** *ip1* [**-***ip2*] [**netmask** *mask*]	Display active translations by global IP address or local IP address using the network mask to qualify the IP addresses.
interface *if1* [**,***if2*] [**,***ifn*]	Display active translations by interface.
lport | **gport** *port* [**-***port*]	Display active translations by local and global port specifications. See "Ports" in Chapter 2, "Using PIX Firewall Commands" for a list of valid port literal names.
state	Display active translations by state; **static** translation (**static**), **dump** (cleanup), PAT **global** (**portmap**), a **nat** or **static** translation with the **norandomseq** setting (**norandomseq**), or the use of the **nat 0**, identity feature (**identity**).
debug	Display translation type and interface information.
count	Display the number of active translations.

Command Modes

Privileged mode.

Usage Guidelines

The **clear xlate** command clears the contents of the translation slots. ("xlate" means translation slot.) The **show xlate** command displays the contents of only the translation slots.

Translation slots can persist after key changes have been made. Always use the **clear xlate** command after adding, changing, or removing the **aaa-server**, **access-list**, **alias**, **conduit**, **global**, **nat**, **route**, or **static** commands in your configuration.

Note When the **vpnclient** configuration is enabled and the inside host is sending out DNS requests, the **show xlate** command may list multiple xlates for a static translation.

The **show xlate detail** command displays the following information:

{**ICMP**|**TCP**|**UDP**} **PAT from** *interface*:*real-address*/*real-port* **to** *interface [acl-name]*:*mapped-address*/*mapped-port* **flags** *translation-flags*

NAT from *interface*:*real-address*/*real-port* **to** *interface [acl-name]*:*mapped-address*/*mapped-port* **flags** *translation-flags*

The translation flags are defined in Table 8-5.

Table 8-5 Translation Flags

Flag	Description
s	static translation slot
d	dump translation slot on next cleaning cycle
r	portmap translation (Port Address Translation)
n	no randomization of TCP sequence number
o	outside address translation
i	inside address translation
D	DNS A RR rewrite
I	identity translation from **nat 0**

Examples

The following is sample output from the **show xlate** command with three active Port Address Translations (PATs):

```
pixfirewall(config)# show xlate
3 in use, 3 most used
PAT Global 192.150.49.1(0) Local 10.1.1.15 ICMP id 340
PAT Global 192.150.49.1(1024) Local 10.1.1.15(1028)
PAT Global 192.150.49.1(1024) Local 10.1.1.15(516)
```

The following is sample output from the **show xlate detail** command with three active Port Address Translations (PATs):

```
pixfirewall(config)# show xlate detail
3 in use, 3 most used
Flags: D - DNS, d - dump, I - identity, i - inside, n - no random,
       o - outside, r - portmap, s - static
TCP PAT from inside:10.1.1.15/1026 to outside:192.150.49.1/1024 flags ri
UDP PAT from inside:10.1.1.15/1028 to outside:192.150.49.1/1024 flags ri
ICMP PAT from inside:10.1.1.15/21505 to outside:192.150.49.1/0 flags ri
```

The first entry is a TCP Port Address Translation for host-port (10.1.1.15, 1025) on the inside network to host-port (192.150.49.1, 1024) on the outside network. The flag "r" denotes the translation is a Port Address Translation. The "i" flags denotes that the translation applies to the inside address-port.

The second entry is a UDP Port Address Translation for host-port (10.1.1.15, 1028) on the inside network to host-port (192.150.49.1, 1024) on the outside network. The flag "r" denotes the translation is a Port Address Translation. The "i" flags denotes that the translation applies to the inside address-port.

The third entry is an ICMP Port Address Translation for host-ICMP-id (10.1.1.15, 21505) on the inside network to host-ICMP-id (192.150.49.1, 0) on the outside network. The flag "r" denotes the translation is a Port Address Translation. The "i" flags denotes that the translation applies to the inside address-ICMP-id.

The inside address fields appear as source addresses on packets traversing from the more secure interface to the less secure interface. Conversely, they appear as destination addresses on packets traversing from the less secure interface to the more secure interface.

The following is sample output from two static translations, the first with two associated connections (called “nconns”) and the second with four.

```
show xlate
Global 209.165.201.10 Local 209.165.201.10 static nconns 1 econns 0
Global 209.165.201.30 Local 209.165.201.30 static nconns 4 econns 0
```

The following is sample output from the **show xlate debug** command:

```
show xlate debug
1 in use, 1 most used
Flags: D - DNS, d - dump, I - identity, i - inside, n - no random,
       o - outside, r - portmap, s - static
NAT from inside:8.0.0.2 to outside:11.0.0.254 flags si idle 0:00:06 timeout 3:00:00
--------------------------------------------------------------------------------
```

Related Commands

show conn	Display all active connections.
show uauth/clear uauth	Display or delete all authorization caches for a user.
timeout	Sets the maximum idle times.

shun

The **shun** command enables a dynamic response to an attacking host by preventing new connections and disallowing packets from any existing connection.

[**no**] **shun** *src_ip* [*dst_ip sport dport* [*protocol*]]

clear shun [*statistics*]

show shun [*src_ip* | *statistics*]

Syntax Description

clear	Disable all shuns currently enabled and clears shun statistics. Specifying statistics only clears the counters for that interface.
dport	The destination port of the connection causing the shun.
dst_ip	The address of the of the target host.
no	Disable a shun based on *src_ip*, the actual address used by the PIX Firewall for shun lookups.

protocol	The optional IP protocol, such as UDP or TCP.
shun	Enable a blocking function (shun) based on *src_ip*.
sport	The source port of the connection causing the shun.
src_ip	The address of the attacking host.
statistics	Clear only interface counters.

Command Modes

Configuration mode.

Usage Guidelines

The **shun** command applies a blocking function to the interface receiving the attack. Packets containing the IP source address of the attacking host will be dropped and logged until the blocking function is removed manually or by the Cisco IDS master unit. No traffic from the IP source address will be allowed to traverse the PIX Firewall unit and any remaining connections will time out as part of the normal architecture. The blocking function of the **shun** command is applied whether or not a connection with the specified host address is currently active.

If the **shun** command is used only with the source IP address of the host, then the other defaults will be 0. No further traffic from the offending host will be allowed.

The **shun** command only blocks packets inbound from that IP on the local interface. Packets destined to the IP address are still allowed.

Because the **shun** command is used to block attacks dynamically, it is not displayed in your PIX Firewall configuration.

Examples

In the following example, the offending host (10.1.1.27) makes a connection with the victim (10.2.2.89) with TCP. The connection in the PIX Firewall connection table reads:

10.1.1.27, 555-> 10.2.2.89, 666 PROT TCP

If the **shun** command is applied in the following way:

```
shun 10.1.1.27 10.2.2.89 555 666 tcp
```

The preceding command would delete the connection from the PIX Firewall connection table, and it would also prevent packets from 10.1.1.27 from going through the PIX Firewall. The offending host can be inside or outside of the PIX Firewall.

The following is sample output of the **show shun** command with the **shun** command applied to the outside interface:

```
outside=ON, cnt=4,time=(0:04:13)
```

The first value indicates if the **shun** command is applied to the interface, the second value (**cnt**) indicates the number of packets that have been dropped since the **shun** command was applied. The third value (**time**) indicates the elapsed time since the **shun** command was applied to the interface.

sip ip-address-privacy

SIP address privacy provides the ability to hide phone IP addresses from one another. SIP fixup will retain outside IP addresses in the SIP header and SDP data of inbound packets. By default this command is turned off. When the command is turned on, SIP fixup will retain outside IP addresses in the SIP header and SDP data of inbound SIP packets.

sip ip-address privacy

[no] sip ip-address-privacy

Syntax Description

Field Name	Field Description
Via	The phone which originates the call puts in its IP address in the Via field.
From(if it contains an IP address)	If PAT is configured and this field does not contain a port, this field is not NATted (in the outbound direction).
Call-ID (if it contains an IP address)	If PAT is configured and this field does not contain a port, this field is not NATted.
o=	This contains the originator's IP address. This is a best effort in case of PAT. i.e, this field does not contain a port, so we do a 'best effort ' to PAT it by checking to see if it matches the connection address, and if it does, we use the m= port as the port to do the PAT. The SDP specification specifies the o= and m= as mandatory parameters in the SDP portion of the SIP packet. So, in a SIP packet conforming to the SDP specification, we will NAT/PAT the o= field with the port from the m= field (as described above).
·c=	This contains the connection IP address.
·m=	If PAT is configured, the PATted port should be retained.
	Record-route contains IP address.

Note By default, this feature is not turned on.

Command Modes

Global configuration

Usage Guidelines

The fixup can be enabled or disabled via the *[no]* **sip ip-address privacy** command.

Examples

INVITE sip:bob@Proxy SIP/2.0

Via: SIP/2.0/UDP A:5060 ==========> A':patport#

From: terry@A ==========> terry@A'

To: robin@Proxy

Call-ID:

Contact:terry@A ==========> terry@A'

SDP

o=A ==========> A'

c=IN IP4 A ==========> A'

m=port# ==========> patport# (if applicable)

When the Proxy sends the INVITE to B:

INVITE sip:robin@Proxy SIP/2.0

Via: SIP/2.0/UDP A':5060 ==========>Has to remain as A':patport#

From: terry@A' ==========>Has to remain as A'

To:robin@Proxy

Call-ID:

Contact:terry@A' ==========>Has to remain as A'

SDP

o=A' ==========>Has to remain as A'

c=IN IP4 A' ==========>Has to remain as A'

m=patport#

Note When this feature is turned on outside NAT will not work. When a packet from the lower security level (eg., outside) comes to the higher security level (eg., inside), since we retain the NATted IP addresses in it and don't send the packet through the NAT engine, outside NAT will not be performed for the inbound SIP packets.

- ·When this feature is off, regular SIP Fixup will work as it does under PIX 6.3.3
- ·When this feature is turned on with sip ip-address privacy, all messages/responses are inspected and NATted IP addresses are retained for all relevant fields.
- · RTP traffic between phones on the same interface must go through the PIX Firewall. Thus, necessary pinholes for RTP traffic must be opened on the PIX.

Related Commands

show running-config can be used to see if the **sip ip-address privacy** command is turned on. Debug messages are available when outside IP addresses are retained in a system message when this feature is enabled.

snmp deny version

snmp deny version filters out traffic based on the protocol version field in SNMP packets with the variable <version-string>. To disable, use the **no** form of this command.

[no] snmp deny version [1 | 2 | 2c | 3]

Syntax Description

1	Specifies SNMP Version 1.
2	Specifies SNMP Version 2.
2c	Specifies SNMP Version 2c.
3	Specifies SNMP Version 3.

Defaults

No default behavior or values.

Command Modes

Global configuration

Usage Guidelines

The fixup can be enabled or disable via the fixup cmd:

[no] fixup protocol snmp 161-162

Note Existing connections will retain present fixup configurations fromtheir initial creation.

So, if you toggle the configuration, you need to either:

- Wait for the connections to time out
- Manually clear the connections

Use **clear xlate** or **clear local** to clear connections for the fixup configuration to take effect.

fixup protocol sqlnet

PIX Firewall uses port 1521 for SQL*Net. This is the default port used by Oracle for SQL*Net; however, this value does not agree with IANA port assignments.

Examples

The following example filters out SNMP Version 2c traffic:

```
pix# snmp deny version 2c
```

Related Commands

fixup protocol snmp

snmp-server

Provide PIX Firewall event information through SNMP.

[**no**] **snmp-server community** *key*

[**no**] **snmp-server** {**contact | location**} *text*

[**no**] **snmp-server host** [*if_name*] *ip_addr* [**trap | poll**]

[**no**] **snmp-server enable traps**

clear snmp-server

show snmp-server

Syntax Description

community *key*	Enter the password key value in use at the SNMP management station. The SNMP community string is a shared secret among the SNMP management station and the network nodes being managed. PIX Firewall uses the key to determine if the incoming SNMP request is valid. For example, you could designate a site with a community string and then configure the routers, firewall, and the management station with this same string. The PIX Firewall then honors SNMP requests using this string and does not respond to requests with an invalid community string. The *key* is a case-sensitive value up to 32 characters in length. Spaces are not permitted. The default is **public** if *key* is not set. Consequently, it is important to specify a (new) value for *key* for security reasons.
contact *text*	Supply your name or that of the PIX Firewall system administrator. The text is case-sensitive and can be up to 127 characters. Spaces are accepted, but multiple spaces are shortened to a single space.
enable traps	Enable or disable sending log messages as SNMP trap notifications.
host	Specify an IP address of the SNMP management station to which traps should be sent and/or from which the SNMP requests come. You can specify up to 32 SNMP management stations.
if_name	The interface name where the SNMP management station resides.
ip_addr	The IP address of a host to which SNMP traps should be sent and/or from which the SNMP requests come.
location *text*	Specify your PIX Firewall location. The text is case-sensitive and can be up to 127 characters. Spaces are accepted, but multiple spaces are shortened to a single space.
trap | poll	Specify whether traps, polls, or both are acted upon. Use with these parameters: • **trap**—Only traps will be sent. This host will not be allowed to poll. • **poll**—Traps will not be sent. This host will be allowed to poll. The default allows both traps and polls to be acted upon.

Command Modes

Configuration mode.

Usage Guidelines

Use the **snmp-server** command to identify site, management station, community string, and user information.

Note In the **snmp-server community** *key* command, the default value for *key* is **public**. Consequently, it is important to specify a (new) value for *key* for security reasons.

The **clear snmp-server** and **no snmp-server** commands disable the SNMP commands in the configuration as follows:

```
no snmp-server location
no snmp-server contact
snmp-server community public
no snmp-server enable traps
```

In understanding SNMP use, the PIX Firewall is considered the SNMP agent or SNMP server. The management station is the system running the SNMP program that receives and processes the SNMP information that the PIX Firewall sends.

An SNMP object ID (OID) for PIX Firewall displays in SNMP event traps sent from the PIX Firewall. The OIDs for the PIX Firewall platforms are listed in Table 8-6.

Table 8-6 System OID in PIX Firewall Platforms

PIX Firewall Platform	System OID
PIX 501	.1.3.6.1.4.1.9.1.417
PIX 506	.1.3.6.1.4.1.9.1.389
PIX 506E	.1.3.6.1.4.1.9.1.450
PIX 515	.1.3.6.1.4.1.9.1.390
PIX 515E	.1.3.6.1.4.1.9.1.451
PIX 520	.1.3.6.1.4.1.9.1.391
PIX 525	.1.3.6.1.4.1.9.1.392
PIX 535	.1.3.6.1.4.1.9.1.393
Others	.1.3.6.1.4.1.9.1.227

Use the **trap** and **poll** command options to configure hosts to participate only in specific SNMP activities. Poll responses and traps are sent only to the configured entities. Hosts configured with the **trap** command option will have traps sent to them, but will not be allowed to poll. Hosts configured with the **poll** command option will be allowed to poll, but will not have traps sent to them. Refer to the *Cisco PIX Firewall and VPN Configuration Guide* for more information on how to access and monitor the PIX Firewall using SNMP traps.

Accessibility to PIX Firewall Management Information Bases (MIBs) is based on configuration, MIB support, and authentication based on the community string. Unsuccessful polling attempts, except for failed community string authentication, are not logged or otherwise indicated. Community authentication failures result in a trap where applicable.

MIB Support

You can browse the System and Interface groups of MIB-II. All SNMP values in the PIX Firewall are read only (RO). The PIX Firewall does not support browsing of the Cisco syslog MIB.

Browsing a MIB is different from sending traps. Browsing means doing an **snmpget** or **snmpwalk** of the MIB tree from the management station to determine values. Traps are different; they are unsolicited "comments" from the managed device to the management station for certain events, such as link up, link down, syslog event generated, and so on.

The Cisco Firewall MIB, Cisco Memory Pool MIB, Cisco Process MIB provide the following PIX Firewall information through SNMP:

- Buffer usage from the **show block** command
- Connection count from the **show conn** command
- CPU usage through the **show cpu usage** command
- Failover status
- Memory usage from the **show memory** command

Receiving SNMP Requests from an SNMP Management Station

To receive SNMP requests from a management station, perform the following steps:

Step 1 Identify the management station with an **snmp-server host** command statement.

Step 2 Specify **snmp-server** command options for the **location**, **contact**, and **community**.

Step 3 Start the SNMP software on the management station and begin issuing SNMP requests to the PIX Firewall.

Defaults

If you do not specify an option, the **snmp-server host** command behaves as in previous versions. The polling is permitted from all configured hosts on the affected interface. Traps are sent to all configured hosts on the affected interface.

Examples

The following example shows commands you would enter to start receiving SNMP requests from a management station:

```
snmp-server community wallawallabingbang
snmp-server location Building 42, Sector 54
snmp-server contact Sherlock Holmes
snmp-server host perimeter 10.1.2.42
```

The next example is sample output from the **show snmp-server** command:

```
show snmp
snmp-server host perimeter 10.1.2.42
snmp-server location Building 42, Sector 54
snmp-server contact Sherlock Holmes
snmp-server community wallawallabingbang
```

ssh

Specify a host for PIX Firewall console access through Secure Shell (SSH).

[**no**] **ssh** *ip_address* [*netmask*] [*interface_name*]

ssh timeout *mm*

ssh disconnect *session_id*

clear ssh

show ssh [**sessions** [*ip_address*]]

show ssh timeout

Syntax Description

interface_name	PIX Firewall interface name on which the host or network initiating the SSH connection resides.
ip_address	IP address of the host or network authorized to initiate an SSH connection to the PIX Firewall.
mm	The duration in minutes that a session can be idle before being disconnected. The default duration is 5 minutes. The allowable range is from 1 to 60 minutes.
netmask	Network mask for *ip_address*. If you do not specify a *netmask*, the default is 255.255.255.255 regardless of the class of *ip_address*.
session_id	SSH session ID number, viewable with the **show ssh sessions** command.

Command Modes

Configuration mode.

Usage Guidelines

The **ssh** *ip_address* command specifies the host or network authorized to initiate an SSH connection to the PIX Firewall. The **ssh timeout** command lets you specify the duration in minutes that a session can be idle before being disconnected. The default duration is 5 minutes. Use the **show ssh sessions** command to list all active SSH sessions on the PIX Firewall. The **ssh disconnect** command lets you disconnect a specific session you observed from the **show ssh sessions** command. Use the **clear ssh** command to remove all **ssh** command statements from the configuration. Use the **no ssh** command to remove selected **ssh** command statements from the configuration.

Note You must generate an RSA key-pair for the PIX Firewall before clients can connect to the PIX Firewall console. After generating the RSA key-pair, save the key-pair using the **ca save all** command. To use SSH, your PIX Firewall must have a DES or 3DES activation key.

To gain access to the PIX Firewall console via SSH, at the SSH client, enter the username as **pix** and enter the Telnet password. You can set the Telnet password with the **passwd** command; the default Telnet password is **cisco**. To authenticate using the AAA server instead, configure the **aaa authenticate ssh console** command.

SSH permits up to 100 characters in a username and up to 50 characters in a password.

When starting an SSH session, a dot (.) displays on the PIX Firewall console before the SSH user authentication prompt appears.

The dot appears as follows:

```
pixfirewall(config)# .
pixfirewall(config)# .
```

The display of the dot does not affect the functionality of SSH. The dot appears on at the console when generating a server key or decrypting a message using private keys during SSH key exchange, before user authentication occurs. These tasks can take up to two minutes or longer. The dot is a progress indicator that verifies that the PIX Firewall is busy and has not hung.

show ssh sessions Command

The **show ssh sessions** command provides the following display:

```
Session ID      Client IP       Version Encryption      State   Username
    0           172.16.25.15    1.5     3DES            4       -
    1           172.16.38.112   1.5     DES             6       pix
    2           172.16.25.11    1.5     3DES            4       -
```

The Session ID is a unique number that identifies an SSH session. The Client IP is the IP address of the system running an SSH client. The Version lists the protocol version number that the SSH client supports. The Encryption column lists the type of encryption the SSH client is using. The State column lists the progress the client is making as it interacts with the PIX Firewall. The Username column lists the login username that has been authenticated for the session. The "pix" username appears when non-AAA authentication is used.

The following table lists the SSH states that appear in the State column:

Number	SSH State
0	SSH_CLOSED
1	SSH_OPEN
2	SSH_VERSION_OK
3	SSH_SESSION_KEY_RECEIVED
4	SSH_KEYS_EXCHANGED
5	SSH_AUTHENTICATED
6	SSH_SESSION_OPEN
7	SSH_TERMINATE
8	SSH_SESSION_DISCONNECTING
9	SSH_SESSION_DISCONNECTED
10	SSH_SESSION_CLOSED

SSH Syslog Messages

Syslog messages 315001, 315002, 315003, 315004, 315005, and 315011 were added for SSH. Refer to *Cisco PIX Firewall System Log Messages* for more information.

Obtaining an SSH Client

The following sites let you download an SSH v1.*x* client. Because SSH Version 1.*x* and 2 are entirely different protocols and are not compatible, be sure you download a client that supports SSH v1.*x*.

- Windows 3.1, Windows CE, Windows 95, and Windows NT 4.0—download the free Tera Term Pro SSH v1.x client from the following website:

  http://hp.vector.co.jp/authors/VA002416/teraterm.html

  The TTSSH security enhancement for Tera Term Pro is available at the following website:

  http://www.zip.com.au/~roca/ttssh.html

  Note You must download TTSSH to use Tera Term Pro with SSH. TTSSH provides a Zip file you copy to your system. Extract the zipped files into the same folder that you installed Tera Term Pro. For a Windows 95 system, by default, this would be the C:\Program Files\Ttempro folder.

- Linux, Solaris, OpenBSD, AIX, IRIX, HP/UX, FreeBSD, and NetBSD—download the SSH v1.x client from the following website:

  http://www.openssh.com

- Macintosh (international users only)—download the Nifty Telnet 1.1 SSH client from the following website:

  http://www.lysator.liu.se/~jonasw/freeware/niftyssh/

Changed aaa Command for SSH

The **aaa** command adds the **ssh** option for use with SSH:

aaa authentication [**serial** | **enable** | **telnet** | **ssh**] **console** *group_tag*

The new **ssh** option specifies the group of AAA servers to be used for SSH user authentication. The authentication protocol and AAA server IP addresses are defined with the **aaa-server** command statement.

Similar to the Telnet model, if the **aaa authentication ssh console** *group_tag* command statement is not defined, you can gain access to the PIX Firewall console with the username **pix** and with the PIX Firewall Telnet password (set with the **passwd** command). If the **aaa** command is defined, but the SSH authentication request times out, this implies that the AAA server may be down or not available. You can gain access to the PIX Firewall using the username **pix** and the enable password (set with the **enable password** command). By default, the Telnet password is **cisco** and the enable password is not set. If the enable password is empty (null), even if you enter the password correctly, you are not granted access to the SSH session.

The user authentication attempt limit is set to 3. Note that the Linux version of the SSH Version 1 client available from http://www.openssh.com only allows one user authentication attempt.

Examples

Create an RSA key-pair with a modulus size of 1024 bits (recommended for use with Cisco IOS software):

```
hostname cisco-pix
domain-name example.com
ca generate rsa key 1024
show ca mypubkey rsa
ca save all
```

These command statements set the host name and domain name for the PIX Firewall, generate the RSA key-pair, display the RSA key-pair, and save the RSA key-pair to Flash memory.

Start an SSH session so clients on the outside interface can access the PIX Firewall console remotely over a secure shell:

```
ssh 10.1.1.1 255.255.255.255 outside
ssh timeout 60
```

Configure the PIX Firewall to perform user authentication using AAA servers. The protocol is the protocol used by the AAA-server to perform the authentication. The following example uses the TACACS+ authentication protocol.

```
aaa-server ssh123 (inside) host 10.1.1.200 mysecure
aaa-server ssh123 protocol tacacs+
aaa authenticate ssh console ssh123
```

Related Commands

- **aaa accounting**
- ca
- domain-name
- **hostname**
- **password**

static

Configure a one-to-one address translation rule by mapping a local IP address to a global IP address, or a local port to a global port.

[**no**] **static** [(*local_ifc*,*global_ifc*)] {*global_ip* | **interface**} {*local_ip* [**netmask** *mask*] | **access-list** *acl_name*} [**dns**] [**norandomseq**] [*max_conns* [*emb_limit*]]

[**no**] **static** [(*local_ifc*,*global_ifc*)] {**tcp** | **udp**} {*global_ip* | **interface**} *global_port* {*local_ip local_port* **[netmask** *mask***]** | **access-list** *acl_name*} [**dns**] [**norandomseq**] [*max_conns* [*emb_limit*]]

show static

Syntax Description

access-list	Lets you identify local traffic for network address translation (NAT) by specifying the local and destination addresses (or ports). This feature is known as policy NAT. The subnet mask used in the access list is also used for the *global_ip*. You can only include **permit** statements in the access list.
acl_name	Specifies the access list name.
dns	Rewrites the local address in DNS replies to the global address.

emb_limit	Specifies the maximum number of embryonic connections per host. An embryonic connection is a connection request that has not finished the necessary handshake between source and destination. Set a small value for slower systems, and a higher value for faster systems. The default is 0, which means unlimited embryonic connections. The embryonic connection limit lets you prevent a type of attack where processes are started without being completed. When the embryonic limit is surpassed, the TCP intercept feature intercepts TCP synchronization (SYN) packets from clients to servers on a higher security level. The software establishes a connection with the client on behalf of the destination server, and if successful, establishes the connection with the server on behalf of the client and combines the two half-connections together transparently. Thus, connection attempts from unreachable hosts never reach the server. The PIX firewall accomplishes TCP intercept functionality using SYN cookies. **Note** This option does not apply to outside NAT. The TCP intercept feature applies only to hosts or servers on a higher security level. If you set the embryonic limit for outside NAT, the embryonic limit is ignored.
global_ifc	Specifies the interface where you want to use the global address. For example, if you want to translate an inside address when it exits the outside interface, then the outside interface is the global interface. If this interface is a higher security level than the local interface, then this translation is known as outside NAT. Some options do not apply to outside NAT.
global_ip	Specifies the global IP address(es) to which you want to translate the local address(es). You can map a single global address to a single local address, or map a range of global addresses to a range of local addresses. This address cannot be used as a dynamic Port Address Translation (PAT) IP address in the **global** command unless you use static PAT, in which case the two addresses can be the same.
global_port	Specifies the global TCP or UDP port. You can specify ports by either a literal name or a number in the range of 0 to 65535. You can view valid port numbers online at the following website: http://www.iana.org/assignments/port-numbers See "Ports" in "Using PIX Firewall Commands" for a list of valid port literal names in port ranges; for example, **ftp** or **h323**. You can also specify numbers.
interface	Specifies the interface IP address for the global address. Use this keyword if you want to use the interface address, but the address is dynamically assigned using DHCP. **Note** You must use the **interface** keyword instead of specifying the actual IP address when you want to include the IP address of a PIX Firewall interface in a static PAT entry.
local_ifc	Specifies the interface that is connected to the local address. For example, if you want to translate an inside address when it exits the outside interface, then the inside interface is the local interface. If this interface is a lower security level than the global interface, then this translation is known as outside NAT. Some options do not apply to outside NAT (such as **norandomseq** and *emb_limit*).

local_ip	Specifies the addresses to translate. You can map a single local address to a single global address or map a range of local addresses to a range of global addresses.
local_port	Specifies the local TCP or UDP port. You can specify ports by either a literal name or a number in the range of 0 to 65535. You can view valid port numbers online at the following website: http://www.iana.org/assignments/port-numbers See "Ports" in "Using PIX Firewall Commands" for a list of valid port literal names in port ranges; for example, **ftp** or **h323**. You can also specify numbers.
mask	Specifies the network mask used for both *global_ip* and *local_ip*. For single hosts, use 255.255.255.255. If you use the **access-list** option instead of the *local_ip*, then the subnet mask used in the access list is also used for the *global_ip*.
max_conns	Specifies the maximum number of simultaneous TCP and UDP connections for the entire subnet. The default is 0, which means unlimited connections. (Idle connections are closed after the idle timeout specified by the **timeout conn** command.) **Note** This option does not apply to outside NAT. The firewall only tracks connections from a higher security interface to a lower security interface. If you set *max_conns* for outside NAT, the *max_conns* option is ignored.
netmask	Specifies the keyword required before specifying the network mask. If you do not enter a mask, then the default mask for the IP address class is used.
norandomseq	Disables TCP Initial Sequence Number (ISN) randomization protection. Only use this option if another inline firewall is also randomizing sequence numbers and the result is scrambling the data. Without this protection, inside hosts with weak self-ISN protection become more vulnerable to TCP connection hijacking. Unless you enable the **norandomseq** option, RCP connections may show a noticable delay with TCP/IP stacks that quickly reuse TCP connections before the timewait state has expired (such as IBM AIX or HP-UX) . **Note** The **norandomseq** option does not apply to outside NAT. The firewall only randomizes the ISN that is generated by the host/server on the higher security interface. If you set **norandomseq** for outside NAT, the **norandomseq** option is ignored.
tcp	Specifies a TCP port.
udp	Specifies a UDP port.

Command Modes

Configuration mode.

Usage Guidelines

The **static** command creates a one-to-one address translation rule (called a static translation slot or "xlate"). Each local address is translated to a fixed global address. With dynamic NAT and PAT, each host uses a different address or port for each consecutive connection. Because the global address is the same for each consecutive connection, and a persistent translation rule exists, the **static** command allows hosts on the global network to initiate traffic to a local host (if the access list allows it).

Static Port Address Translation (PAT) is the same as static NAT, except it allows you to specify the protocol (TCP or UDP) and port for the local and global addresses.

After changing or removing a **static** command statement, use the **clear xlate** command to clear the translations.

Unless you use static PAT, you cannot create multiple **static** commands with the same global IP addresses.

Static Port Address Translation (Static PAT)

This feature allows you to identify the same global address across many different static statements, so long as the port is different for each statement (you cannot use the same global address for multiple static *NAT* statements). For example, if you want to provide a single address for global users to access FTP, HTTP, and SMTP, but these are all actually different servers on the local network, you can specify static statements for:

- local_ip_A/global_ip_A/FTP
- local_ip_B/global_ip_A/HTTP
- local_ip_C/global_ip_A/SMTP

Note To include the IP address of a PIX Firewall interface in a static PAT entry, use the **interface** keyword instead of specifying the actual IP address.

You can also use this feature to translate a well-known port to a lesser-known port or vice versa. For example, if your inside web servers use port 8080, you can allow outside users to connect to port 80, and then translate them to the correct port. Similarly, if you want to provide extra security, you can tell your web users to connect to lesser-known port 6785, and then translate them to port 80 on the local network.

Note PIX Firewall Version 6.2 introduced support for PAT and static PAT of H.323 application traffic; PAT is not supported for H.323 in earlier versions.

Static PAT supports all applications that are supported by dynamic PAT, including the same application constraints. The Telnet port 23 and PFM port 1467 of the PIX Firewall interface cannot be used for Static PAT because the PIX Firewall requires that traffic to these ports be protected by IPSec.

static access-list (Policy NAT)

When you use an access list with the **static** command, then you enable policy NAT.

Policy NAT lets you identify local traffic for address translation by specifying the source and destination addresses (or ports) in an access list. Regular NAT uses source addresses/ports only, whereas policy NAT uses both source and destination addresses/ports.

With policy NAT, you can create multiple **static** statements that identify the same local address as long as the source/port and destination/port combination is unique for each statement. You can then match different global addresses to each source/port and destination/port pair.

While static PAT already allowed you to identify the local and global ports, policy NAT enhances this feature (as well as static NAT) by allowing you to identify the destination address for the local traffic.

Identity NAT

If you want to bypass NAT and allow the local address to appear unchanged on the global network, you can enter the same address for the local and global addresses:

static (*local_ifc*, *global_ifc*) *local_ip local_ip ...*

You can use policy NAT with identity NAT to bypass NAT only for traffic going to a particular destination.

Permitting Inbound Traffic with Access Lists

In addition to using the **static** command, you must also use an **access-list** command to allow outside traffic to access inside hosts or servers.

For example, the host you want to make accessible on the dmz2 network is 192.168.1.1. The static command maps this address to 10.1.1.1:

```
static (dmz2,dmz1) 10.1.1.1 192.168.1.1 netmask 255.255.255.255
```

The **access-list** and **access-group** commands allow traffic from the dmz1 network to access this host on the dmz2 network. Note that the host that dmz1 users want to access is the translated global address 10.1.1.1.

```
access-list acl_dmz1 permit tcp 10.1.1.0 255.255.255.0 host 10.1.1.1
access-group acl_dmz1 in interface dmz1
```

Note Always make **access-list** command statements as specific as possible. Using the **any** option to allow any host access should be used with caution for access lists used with statics.

Order of NAT Commands Used to Match Local Addresses

The firewall matches local traffic to NAT commands in the following order:

1. **nat 0 access-list** (NAT exemption)—In order, until the first match. For example, you could have overlapping local/destination addresses in multiple **nat** commands, but only the first command is matched.
2. **static** (static NAT)—In order, until the first match. Because you cannot use the same local address in static NAT or static PAT commands, the order of **static** commands does not matter. Similarly, for static policy NAT, you cannot use the same local/destination address and port across multiple statements.
3. **static** {**tcp** | **udp**} (static PAT)—In order, until the first match. Because you cannot use the same local address in static NAT or static PAT commands, the order of **static** commands does not matter. Similarly, for static policy NAT, you cannot use the same local/destination address and port across multiple statements.
4. **nat** *nat_id* **access-list** (policy NAT)—In order, until the first match. For example, you could have overlapping local/destination ports and addresses in multiple **nat** commands, but only the first command is matched.
5. **nat** (regular NAT)—Best match. The order of the NAT commands does not matter. The **nat** statement that best matches the local traffic is used. For example, you can create a general statement to translate all addresses (0.0.0.0) on an interface. If you also create a statement to translate only 10.1.1.1, when 10.1.1.1 makes a connection, the specific statement for 10.1.1.1 is used because it matches the local traffic best.

Failover and the static command

The **static** command, without a port specified, translates all traffic received on the interface, including failover messages sent by a standby failover unit. In this case, the standby failover unit sends messages to the active unit, but they bypass the active unit, so the standby failover unit receives no replies from the active unit and it assumes that the interface is down and becomes the active unit. When you specify the port number, only traffic to that port is translated, and this situation is avoided. (Because failover uses a unique port number, port 105, it is not translated when other specific ports are.)

statics and VoIP

In networks with VoIP traffic, pay close attention to any static translations in your configuration. VoIP calls can fail to pass through the firewall if, after configuring a static translation for a network, the third party endpoint has a global IP address that matches the static translation. For example, if the IP addresses are as follows:

```
inside IP phone: 10.132.60.231
outside IP phone 10.130.60.215
outside CM: 10.130.60.111
```

and the following command is used:

```
static (inside,outside) 10.130.60.0 10.132.60.0 netmask 255.255.255.0
```

Then, when the firewall receives a message from the outside CM to the inside phone, the firewall sees the outside phone's IP address as a global IP address of an inside phone and translates it (so the call does not connect).

TCP Intercept

Prior to Version 5.3, the PIX Firewall offered no mechanism to protect systems reachable via a static and TCP conduit from TCP SYN attacks. Previously, if an embryonic connection limit was configured in a **static** command statement, the firewall simply dropped new connection attempts once the embryonic threshold was reached. Given this, a modest attack could stop web traffic. For **static** command statements without an embryonic connection limit, the firewall passes all traffic. If the affected system does not have TCP SYN attack protection, and most operating systems do not offer sufficient protection, then the affected system's embryonic connection table overloads and all traffic stops.

With the TCP intercept feature, once the optional embryonic connection limit is reached, and until the embryonic connection count falls below this threshold, every SYN bound for the affected server is intercepted. For each SYN, PIX Firewall responds on behalf of the server with an empty SYN/ACK segment. PIX Firewall retains pertinent state information, drops the packet, and waits for the client's acknowledgement. If the ACK is received, then a copy of the client's SYN segment is sent to the server and the TCP three-way handshake is performed between PIX Firewall and the server. If and only if, this three-way handshake completes, may the connection resume as normal. If the client does not respond during any part of the connection phase, then PIX Firewall retransmits the necessary segment using exponential back-offs.

TCP intercept requires no change to the PIX Firewall command set. Note only that the embryonic connection limit on the **static** command now has a new behavior.

Note The TCP intercept feature applies only to hosts or servers on a higher security level. If you set the embryonic limit for outside NAT, the embryonic limit is ignored.

Deny Xlate for Network or Broadcast Address for Inbound Traffic

For all inbound traffic, the firewall denies translations for destination IP addresses identified as network address or broadcast addresses. The firewall utilizes the global IP and mask from a **static** command statement to differentiate regular IP addresses from network or broadcast addresses. If a global IP address is a valid network address with a matching network mask, then the firewall disallows the translation for network or broadcast IP addresses with inbound packet.

Interfaces on Which to Use Static NAT or Dynamic NAT

The rules for which command to use with an interface is summarized in Table 8-7. Table 8-7 assumes that the security levels are 40 for dmz1 and 60 for dmz2.

Table 8-7 Interface Access Commands by Interface

From This Interface	To This Interface	Use This Command
inside	outside	**nat**
inside	dmz1	**nat**
inside	dmz2	**nat**
dmz1	outside	**nat**
dmz1	dmz2	**static**
dmz1	inside	**static**
dmz2	outside	**nat**
dmz2	dmz1	**nat**
dmz2	inside	**static**
outside	dmz1	**static**
outside	dmz2	**static**
outside	inside	**static**

Examples

Basic Static NAT Examples

The following example permits a finite number of users to call in through H.323 using an Intel Internet Phone, CU-SeeMe, CU-SeeMe Pro, MeetingPoint, or MS NetMeeting. The **static** command maps addresses 209.165.201.1 through 209.165.201.30 to local addresses 10.1.1.1 through 10.1.1.30 (209.165.201.2 maps to 10.1.1.2, 209.165.201.10 maps to 10.1.1.10, and so on). The accompanying **access-list** and **access-group** commands allow traffic from a lower security interface to a higher security interface.

```
static (inside,outside) 209.165.201.0 10.1.1.0 netmask 255.255.255.224
access-list acl_out permit tcp any 209.165.201.0 255.255.255.224 eq h323
access-group acl_out in interface outside
```

The following example shows the commands used to disable Mail Guard:

```
static (dmz1,outside) 209.165.201.1 10.1.1.1 netmask 255.255.255.255
access-list acl_out permit tcp any host 209.165.201.1 eq smtp
access-group acl_out in interface outside
no fixup protocol smtp 25
```

In this example, the **static** command sets up a global address to permit outside hosts access to the 10.1.1.1 mail server host on the dmz1 interface. (The MX record for DNS needs to point to the 209.165.201.1 address so that mail is sent to this address.) The **access-list** command lets any outside users access the global address through the SMTP port (25). The **no fixup protocol** command disables Mail Guard.

Static PAT Examples

To redirect Telnet traffic from the PIX Firewall outside interface to the inside host at 10.1.1.15, enter:

```
static (inside,outside) tcp interface telnet 10.1.1.15 telnet netmask 255.255.255.255
```

To redirect FTP traffic from the PIX Firewall outside interface to the inside host at 10.1.1.30, enter:

```
static (inside,outside) tcp interface ftp 10.1.1.30 ftp netmask 255.255.255.255
```

To redirect DNS traffic from the PIX Firewall outside interface to the inside host at 10.1.1.30, enter:

```
static (inside,outside) udp interface domain 10.1.1.30 domain netmask 255.255.255.255
```

To allow the local Telnet server to initiate connections other than Telnet, you need to provide additional translation. For example, to translate all other types of traffic to the same address used in the static translation for Telnet (the interface address, for example), enter the following commands:

```
static (inside,outside) tcp 10.1.2.14 telnet 10.1.1.15 telnet netmask 255.255.255.255
nat (inside) 1 10.1.1.15 255.255.255.255
global (outside) 1 10.1.2.14 netmask 255.255.255.255
```

The **static** command provides the translation for Telnet. The **nat** and **global** commands provide PAT for all other outbound connections from the server.

If you have a separate translation for all inside traffic that uses a different global address, you can still configure the Telnet server to use the same address as the static statement by creating a more exclusive **nat** statement just for that server. Because **nat** statements are read for the best match, more exclusive **nat** statements are matched before general statements.

```
static (inside,outside) tcp 10.1.2.14 telnet 10.1.1.15 telnet netmask 255.255.255.255
nat (inside) 1 10.1.1.15 255.255.255.255
global (outside) 1 10.1.2.14 netmask 255.255.255.255
nat (inside) 2 0.0.0.0 0.0.0.0
global (outside) 2 10.1.2.78 netmask 255.255.255.255
```

To translate a well-known port (80) to another port (8080), enter:

```
static (inside,outside) tcp 10.1.2.45 80 10.1.1.16 8080 netmask 255.255.255.255
```

Policy NAT Examples

The following example shows a Policy NAT configuration. In this example, traffic destined for the 172.16.1.0/24 from host 10.1.1.10 is translated as 192.150.49.10, and traffic destined for the 172.16.2.0/24 from host 10.1.1.10 is translated as 192.150.49.20:

```
access-list network-1 permit ip host 10.1.1.10 172.16.1.0 255.255.255.0
access-list network-2 permit ip host 10.1.1.10 172.16.2.0 255.255.255.0
static (inside,outside) 192.150.49.10 access-list network-1
static (inside,outside) 192.150.49.20 access-list network-2
```

If you want to use identity NAT from traffic going from 10.1.1.1 to 10.2.2.3, but you want to translate 10.1.1.1 to 10.4.5.6 when going to 10.3.1.0/24, you could enter:

```
access-list IDENTITY permit ip host 10.1.1.1 host 10.2.2.3
access-list TRANSLATE permit ip host 10.1.1.1 10.3.1.0 255.255.255.0
static (inside,outside) 10.1.1.1 access-list IDENTITY
static (inside,outside) 10.4.5.6 access-list TRANSLATE
```

Identity NAT Examples

For example, a web server on the **dmz**, 209.165.201.5 needs to be accessible by users on the outside. The **static** and **access-list** command statements are as follows:

```
static (dmz,outside) 209.165.201.5 209.165.201.5 netmask 255.255.255.255
access-list acl_out permit tcp any host 209.165.201.5 eq www
access-group acl_out in interface outside
```

The **static** command presents the 209.165.201.5 address on the outside interface. The DNS server on the outside would map this IP address to the domain of the company; for example, example.com. Users accessing example.com are permitted to access the web server via port 80 by the **access-list** command.

Another example of identity NAT statics is when users on dmz1 need to access a web server on dmz2. The network uses a Class C address and the .240 subnet. Addresses 209.165.201.1 to 209.165.201.14 are on dmz1, and addresses 209.165.201.17 to 209.165.201.30 are on dmz2. The web server is at 209.165.201.25. The **static** and **access-list** command statements are as follows:

```
static (dmz2,dmz1) 209.165.201.25 209.165.201.25 netmask 255.255.255.255
access-list acl_dmz1 permit tcp any host 209.165.201.25 eq www
access-group acl_dmz1 in interface dmz1
```

The **static** command statement opens access to the web server at 209.165.201.25. The **access-list** command statement permits access to the web server only on port 80 (**www**).

Related Commands

- **access-list**
- **show xlate/clear xlate**

syslog

Enable syslog message facility. Obsolete command replaced by the **logging** command.

See the **logging** command for more information. The **syslog** command is available for backward compatibility.

sysopt

Change PIX Firewall system options.

[no] sysopt connection {permit-pptp | permit-l2tp | permit-ipsec}

[no] sysopt connection tcpmss [minimum] *bytes*

[no] sysopt connection timewait

[no] sysopt ipsec pl-compatible

[no] sysopt nodnsalias {inbound | outbound}

[no] sysopt noproxyarp *if_name*

[no] sysopt radius ignore-secret

[no] sysopt uauth allow-http-cache

clear sysopt

show sysopt

Syntax Description

connection permit-ipsec	Implicitly permit any packet that came from an IPSec tunnel and bypass the checking of an associated **access-list, conduit,** or **access-group** command statement for IPSec connections.
connection permit-l2tp	Implicitly permit any packet that came from an L2TP/IPSec tunnel and bypass the checking of an associated **access-list, conduit,** or **access-group** command statement for L2TP/IPSec connections.
connection permit-pptp	Allow PPTP traffic to bypass **conduit** or **access-list** command statement checking.
connection tcpmss [minimum] *bytes*	Overrides the maximum TCP segment size to be no greater than *bytes*. The **minimum** keyword overrides the maximum segment size to be no less than *bytes*. The minimum value is 48 bytes. The default value is 1380 bytes.
connection timewait	Force each TCP connection to linger in a shortened TIME_WAIT state of at least 15 seconds after the final normal TCP close-down sequence.
ipsec pl-compatible	Enable IPSec packets to bypass the PIX Firewall unit's NAT and ASA features and allows incoming IPSec packets to terminate on the inside interface.
nodnsalias inbound	Disable inbound embedded DNS A record fixups according to aliases that apply to the A record address.
nodnsalias outbound	Disable outbound DNS A record replies.
noproxyarp *if_name*	Disable proxy-ARPs on a PIX Firewall interface.
radius ignore-secret	Ignore authenticator key to avoid retransmit caveat.
uauth allow-http-cache	Allows the web browser to supply a username and password from its cache for AAA authentication.

Command Modes

Configuration mode.

Usage Guidelines

The **sysopt** commands let you tune various PIX Firewall security and configuration features. In addition, you can use this command to disable the PIX Firewall IP Frag Guard feature.

There is no need to enter the **sysopt connection permit-12tp** command if the **sysopt connection permit-ipsec** command is present.

sysopt connection permit-ipsec

Use the **sysopt connection permit-ipsec** command in IPSec configurations to permit IPSec traffic to pass through the PIX Firewall without a check of **conduit** or **access-list** command statements.

An **access-list** or **conduit** command statement must be available for inbound sessions.

By default, any inbound session must be explicitly permitted by a **conduit** or **access-list** command statement. With IPSec protected traffic, the secondary access list check could be redundant. To enable IPSec authenticated/cipher inbound sessions to always be permitted, use the **sysopt connection permit-ipsec command.**

If both the **sysopt ipsec pl-compatible** command and the **sysopt connection permit-ipsec** command are used within your configuration, the **sysopt ipsec pl-compatible** command will take precedence.

Note The **sysopt ipsec pl-compatible** command is deprecated. In its place, we recommend using the **nat 0 access-list** command to exempt IPSec from NAT.

If the **sysopt connection permit-ipsec** command is not configured, you must explicitly configure an **access-list** command statement to permit IPSec traffic to traverse the PIX Firewall.

The **no sysopt connection permit-ipsec** command disables the option.

sysopt connection permit-pptp

Let PPTP traffic bypass **conduit** and **access-list** command statement checking. Use the **vpdn** command to implement PPTP.

sysopt connection permit-l2tp

This command allows L2TP traffic to bypass conduit or access list checking. Because L2TP traffic can only come from IPSec, the **sysopt connection permit-ipsec** command will allow L2TP traffic to pass as well.

sysopt ipsec pl-compatible

Note The **sysopt ipsec pl-compatible** command provides a migration path for Private Link users from Private Link tunnels to IPSec tunnels.

The **sysopt ipsec pl-compatible** command enables the IPSec feature to simulate the Private Link feature supported in PIX Firewall Version 4. The Private Link feature provides encrypted tunnels to be established across an unsecured network between Private-Link equipped PIX Firewall units. The **sysopt ipsec pl-compatible** command allows IPSec packets to bypass the NAT and ASA features and enables incoming IPSec packets to terminate on the sending interface.

The **sysopt ipsec pl-compatible** command is not available on a PIX 501.

The **no sysopt ipsec pl-compatible** command disables the option, which is off by default.

Note When using the **sysopt ipsec pl-compatible** command, all PIX Firewall features, such as access list control, stateful inspection, and user authentication, are bypassed for IPSec packets only.

If both the **sysopt ipsec pl-compatible** command and the **sysopt connection permit-ipsec** command are used within your configuration, the **sysopt ipsec pl-compatible** command will take precedence.

If the **alias** command is used with the **sysopt ipsec pl-compatible** command, a static **route** command statement must be added for each IP address specified in the **alias** command statement.

sysopt connection tcpmss

The **sysopt connection tcpmss** command allows you to set the minimum and the maximum TCP segment size. Both the host and the server can set the maximum segment size when they first establish a connection. If either maximum exceeds the value you set with the **sysopt connection tcpmss** command, then the PIX firewall overrides the maximum and inserts the value you set. If either maximum is less than the value you set with the **sysopt connection tcpmss minimum** command, then the PIX firewall overrides the maximum and inserts the minimum value you set. For example, if you set a maximum size of 1200 bytes and a minimum size of 400 bytes, when a host requests a maximum size of 1300 bytes, then the PIX firewall alters the packet to request 1200 bytes (the maximum). If another host requests a maximum value of 300 bytes, then the PIX firewall alters the packet to request 400 bytes (the minimum).

The *bytes* value can be a minimum of 48 and any maximum number. You can disable this feature by setting *bytes* to 0. By default, the PIX firewall sets 1380 bytes as the **sysopt connection tcpmss** maximum limit and 48 bytes as the minimum limit, even though this command does not appear in the default configuration. The default of 1380 bytes allows room for header information so that the total packet size does not exceed 1500 bytes, which is the default MTU for Ethernet. See the following calculation:

```
1380 data + 20 TCP + 20 IP + 24 AH + 24 ESP_CIPHER + 12 ESP_AUTH + 20 IP = 1500 bytes
```

If the host or server does not request a maximum segment size, the PIX firewall assumes that the RFC 793 default value of 536 bytes is in effect.

You might want to set the maximum segment size using this command so that the size is less than the MTU and packets are not fragmented. Large numbers of fragments can impact the performance of the PIX firewall when it uses the Frag Guard feature. Setting the minimum size prevents the TCP server from sending many small TCP data packets to the client and impacting the performance of the server and the network.

Note Although, not advised for normal use of this feature, if you encounter the syslog IPFRAG messages 209001 and 209002, you can raise the *bytes* value.

sysopt connection timewait

By default the PIX Firewall does not use the **timewait** option.

Use the **sysopt connection timewait** command to enable the **timewait** option when you have an end host application whose default TCP terminating sequence is a simultaneous close.

This is recommended because the default behavior of the PIX Firewall is to track the shutdown sequence and release the connection after two FINs and the ACK (acknowledgment) of the last FIN segment. This quick release heuristic enables the PIX Firewall to sustain a high connection rate, based on the most common closing sequence, known as the normal close sequence. However, in a simultaneous close, both ends of the transaction initiate the closing sequence, as opposed to the normal close sequence where one end closes and the other end acknowledges prior to initiating its own closing sequence (see RFC 793). Thus, in a simultaneous close, the quick release forces one side of the connection to linger in the CLOSING state. Having many sockets in the CLOSING state can degrade the performance of an end host. For instance, some WinSock mainframe clients are known to exhibit this behavior and degrade the performance of the mainframe server. Old versions of HP/UX are also susceptible to this behavior. Using the **sysopt connection timewait** command creates a window for the simultaneous close down sequence to complete.

The **no sysopt connection timewait** command removes the **sysopt connection timewait** command from your configuration. In other words, if you enable the **timewait** option with the **sysopt connection timewait** command, you can disable it using the **no sysopt connection timewait** command.

Note The **sysopt connection timewait** command requires more system resources than default processing and, when in use, may impact PIX Firewall performance. Noticeable performance impact is most likely when there is limited memory available, and when there is highly dynamic traffic such as HTTP.

sysopt nodnsalias

The **sysopt nodnsalias inbound** disables inbound embedded DNS A record fixups according to aliases that apply to the A record address. **sysopt nodnsalias outbound** affects outbound replies.

This command remedies the case when a DNS server is on the outside and users on the inside need to access a server on a perimeter interface. In the past, you would use the **alias** command to permit DNS responses to resolve correctly through the PIX Firewall, but formerly you had to reverse the parameters for the local IP address and foreign IP address.

For example, you would normally code the **alias** command as follows:

```
alias (inside) 192.168.1.4 209.165.201.11 255.255.255.255
```

Inside host 192.168.1.5 needs access to www.example.com, which resolves at an outside ISP DNS to 209.165.201.11. The PIX Firewall fixes this DNS response sending the host a response of 192.168.1.4. The host uses its gateway (the PIX Firewall) to go to 192.168.1.4, which the PIX Firewall now aliases back to the 209.165.201.11. Because this is actually 192.168.1.4, a server on the perimeter interface of the PIX Firewall, the packet is dropped because the PIX Firewall sent the packet to the outside interface, which is the incorrect interface.

The **sysopt nodnsalias inbound** command has the same effect as reversing the **alias** command statement parameters as follows:

```
alias (inside) 209.165.201.11 192.168.1.4 255.255.255.255
```

This works properly because everything happens in reverse. The DNS is now modified to 209.165.201.11 and the host inside uses its gateway (the PIX Firewall) to get there, the PIX Firewall aliases this back to 192.168.1.4 and routes it out the perimeter interface to the correct host and the TCP connection is established.

sysopt noproxyarp

ARP (Address Resolution Protocol) is a layer two protocol that resolves an IP address to a physical address, also called a Media Access Controller (MAC) address. A host sends an ARP request asking "Who is this IP?" The device owning the IP should reply with "Hey, I am the one, here's my MAC address."

Proxy ARP refers to a gateway device, in this case, the firewall, "impersonating" an IP address and returning its own MAC address to answer an ARP request for another device.

The firewall builds a table from responses to ARP requests to map physical addresses to IP addresses. A periodic ARP function is enabled in the default configuration. The presence of entries in the ARP cache indicates that the firewall has network connectivity. The show arp command lists the entries in the ARP table. Usually, administrators do not need to manually manipulate ARP entries on the firewall. This is done only when troubleshooting or solving network connectivity problems.

The arp command is used to add a permanent entry for host on a network. If one host is exchanged for another host with the same IP address then the "clear arp" command can be used to clear the ARP cache on the PIX. Alternatively, you can wait for the duration specified with the arp timeout command to expire and the ARP table rebuilds itself automatically with the new host information.

The sysopt noproxyarp command is used to disable Proxy ARPs on an interface from the command-line interface. By default, the PIX Firewall responds to ARP requests directed at the PIX Firewall's interface IP addresses as well as to ARP requests for any static or global address defined on the PIX Firewall interface (which are proxy ARP requests).

The **sysopt noproxyarp** *if_name* command lets you disable proxy ARP request responses on a PIX Firewall interface. However, this command does not disable (non-proxy) ARP requests on the PIX Firewall interface itself. Consequently, if you use the **sysopt noproxyarp** *if_name* command, the PIX Firewall no longer responds to ARP requests for the addresses in the **static**, **global**, and **nat 0** commands for that interface but does respond to ARP requests for its interface IP addresses.

To disable Proxy ARPs on the inside interface:

```
sysopt noproxyarp inside
```

To enable Proxy ARPs on the inside interface:

```
no sysopt noproxyarp inside
```

sysopt radius ignore-secret

Some commonly used RADIUS servers, such as Livingston Version 1.16, have a usage caveat where they do not include the key in the authenticator hash in the accounting acknowledgment response. This can cause the PIX Firewall to continually retransmit the accounting request. Use the **sysopt radius ignore-secret** command to cause the PIX Firewall to ignore the key in the authenticator of accounting acknowledgments thus avoiding the retransmit problem. (The key described here is the key you set with the **aaa-server** command.)

show sysopt

The **show sysopt** command lists the **sysopt** commands in the configuration. The **clear sysopt** command resets the **sysopt** command to default settings.

Deprecated Commands

The **sysopt route dnat** and **sysopt security fragguard** commands are deprecated commands.

Examples

The following displays the default sysopt configuration:

```
pixfirewall(config)# show sysopt
no sysopt connection timewait
sysopt connection tcpmss 1380
sysopt connection tcpmss minimum 0
no sysopt nodnsalias inbound
no sysopt nodnsalias outbound
no sysopt radius ignore-secret
no sysopt uauth allow-http-cache
no sysopt connection permit-ipsec
no sysopt connection permit-pptp
no sysopt connection permit-l2tp
no sysopt ipsec pl-compatible
```

In the following example, a PPTP client authenticates using MS-CHAP, negotiates MPPE encryption, receives the DNS and WINS server addresses, and Telnets to the host 192.168.0.2 directly through the **nat 0** command.

```
ip local pool my-addr-pool 10.1.1.1-10.1.1.254
aaa-server my-aaa-server-group (inside) host 192.168.0.10 key 12345678
aaa-server my-aaa-server-group protocol radius
vpdn group 1 accept dialin pptp
vpdn group 1 ppp authentication mschap
vpdn group 1 ppp encryption mppe auto required
vpdn group 1 client configuration address local my-addr-pool
vpdn group 1 client authentication aaa my-aaa-server-group
vpdn group 1 client configuration dns 10.2.2.99
vpdn group 1 client configuration wins 10.2.2.100
vpdn enable outside
access-list nonat permit ip 10.1.1.0 255.255.255.0 host 192.168.0.2
access-list nonat permit ip 10.1.1.0 255.255.255.0 host 10.2.2.99
access-list nonat permit ip 10.1.1.0 255.255.255.0 host 10.2.2.100
nat (inside) 0 access-list nonat
sysopt connection permit-pptp
```

sysopt connection permit-ipsec

The following is a minimal IPSec configuration to enable a session to be connected from host 172.21.100.123 to host 172.21.200.67 across an IPSec tunnel that terminates from peer 209.165.201.1 to peer 201.165.200.225.

With **sysopt connection permit-ipsec** and **access-list** command statements:

On peer 209.165.201.1:

```
static 172.21.100.123 172.21.100.123
access-list 10 permit ip host 172.21.200.67 host 172.21.100.123
crypto ipsec transform-set t1 esp-des esp-md5-hmac
crypto map mymap 10 ipsec-isakmp
crypto map mymap 10 match address 10
crypto map mymap 10 set transform-set t1
crypto map mymap 10 set peer 172.21.200.1
crypto map mymap interface outside
```

On peer 201.165.200.225:

```
static 172.21.200.67 172.21.200.67
access-list 10 permit ip host 172.21.100.123 host 172.21.200.67
crypto ipsec transform-set t1 esp-des esp-md5-hmac
crypto map mymap 10 ipsec-isakmp
crypto map mymap 10 match address 10
crypto map mymap 10 set transform-set t1
crypto map mymap 10 set peer 172.21.100.1
crypto map mymap interface outside
```

With **sysopt connection permit-ipsec** and without **conduit** command statements:

On peer 209.165.201.1:

```
static 172.21.100.123 172.21.100.123
access-list 10 permit ip host 172.21.200.67 host 172.21.100.123
crypto ipsec transform-set t1 esp-des esp-md5-hmac
crypto map mymap 10 ipsec-isakmp
crypto map mymap 10 match address 10
crypto map mymap 10 set transform-set t1
crypto map mymap 10 set peer 172.21.200.1
crypto map mymap interface outside
sysopt connection permit-ipsec
```

On peer 201.165.200.225:

```
static 172.21.200.67 172.21.200.67
access-list 10 permit ip host 172.21.100.123 host 172.21.200.67
crypto ipsec transform-set t1 esp-des esp-md5-hmac
crypto map mymap 10 ipsec-isakmp
crypto map mymap 10 match address 10
crypto map mymap 10 set transform-set t1
crypto map mymap 10 set peer 172.21.100.1
crypto map mymap interface outside
sysopt connection permit-ipsec
```