



G through L Commands

global

Create or delete entries from a pool of global addresses. (Configuration mode.)

global [(if_name)] nat_id {global_ip [-global_ip] [netmask global_mask]} | interface

no global [(if_name)] nat_id [global_ip [-global_ip] [netmask global_mask]] | [interface]

show global

clear global

Syntax Description

<i>if_name</i>	The external network where you use these global addresses.
<i>nat_id</i>	A positive number shared with the nat command that groups the nat and global command statements together. The valid ID numbers can be any positive number up to 2,147,483,647.
<i>global_ip</i>	One or more global IP addresses that the PIX Firewall shares among its connections. If the external network is connected to the Internet, each global IP address must be registered with the Network Information Center (NIC). You can specify a range of IP addresses by separating the addresses with a dash (-). You can create a Port Address Translation (PAT) global command statement by specifying a single IP address. You can have multiple PAT global command statements per interface. A PAT can support up to 65,535 xlate objects.
netmask	Reserved word that prefaces the network <i>global_mask</i> variable.
<i>global_mask</i>	The network mask for <i>global_ip</i> . If subnetting is in effect, use the subnet mask; for example, 255.255.255.128. If you specify an address range that overlaps subnets, global will not use the broadcast or network addresses in the pool of global addresses. For example, if you use 255.255.255.224 and an address range of 209.165.201.1-209.165.201.30, the 209.165.201.31 broadcast address and the 209.165.201.0 network address will not be included in the pool of global addresses.
interface	Specifies PAT using the IP address at the interface.
clear	Removes global command statements from the configuration.

Usage Guidelines

The **global** command defines a pool of global addresses. The global addresses in the pool provide an IP address for each outbound connection, and for those inbound connections resulting from outbound connections. Ensure that associated **nat** and **global** command statements have the same *nat_id*.

After changing or removing a **global** command statement, use the **clear xlate** command.

Use the **no global** command to remove access to a *nat_id*, or to a Port Address Translation (PAT) address, or address range within a *nat_id*. Use the **show global** command to view the **global** command statements in the configuration.

Usage Notes

1. You can enable the Port Address Translation (PAT) feature by entering a single IP address with the **global** command. PAT lets multiple outbound sessions appear to originate from a single IP address. With PAT enabled, the PIX Firewall chooses a unique port number from the PAT IP address for each outbound xlate (translation slot). This feature is valuable when an Internet service provider cannot allocate enough unique IP addresses for your outbound connections. An IP address you specify for a PAT cannot be used in another global address pool.
2. When a PAT augments a pool of global addresses, first the addresses from the global pool are used, then the next connection is taken from the PAT address. If a global pool address frees, the next connection takes that address. The global pool addresses always come first, before a PAT address is used. Augment a pool of global addresses with a PAT by using the same *nat_id* in the **global** command statements that create the global pools and the PAT.

For example:

```
global (outside) 1 209.165.201.1-209.165.201.10 netmask 255.255.255.224
global (outside) 1 209.165.201.22 netmask 255.255.255.224
```

3. PAT does not work with H.323 applications and caching nameservers. Do not use a PAT when multimedia applications need to be run through the PIX Firewall. Multimedia applications can conflict with port mappings provided by PAT.
4. PAT does not work with the **established** command.
5. PAT works with DNS, FTP and passive FTP, HTTP, email, RPC, rshell, Telnet, URL filtering, and outbound traceroute.

However for use with passive FTP, use the **fixup protocol ftp strict** command statement with an **access-list** command statement to permit outbound FTP traffic, as shown in the following example:

```
fixup protocol ftp strict ftp
access-list acl_in permit tcp any any eq ftp
access-group acl_in in interface inside
nat (inside) 1 0 0
global (outside) 1 209.165.201.5 netmask 255.255.255.224
```

6. IP addresses in the pool of global addresses specified with the **global** command require reverse DNS entries to ensure that all external network addresses are accessible through the PIX Firewall. To create reverse DNS mappings, use a DNS PTR record in the address-to-name mapping file for each global address. For more information on DNS, refer to *DNS and BIND*, by Paul Albitz and Cricket Liu, O'Reilly & Associates, Inc., ISBN 1-56592-010-4. Without the PTR entries, sites can experience slow or intermittent Internet connectivity and FTP requests that consistently fail. For example, if a global IP address is 209.165.201.1 and the domain for the PIX Firewall is pix.example.com, the PTR record would be as follows.

```
1.201.165.209.in-addr.arpa. IN PTR pix.example.com
```

7. A DNS server on a higher level security interface needing to get updates from a root name server on the outside interface cannot use PAT (Port Address Translation). Instead, a **static** command statement must be added to map the DNS server to a global address on the outside interface.

For example, PAT is enabled with these commands:

```
nat (inside) 1 192.168.1.0 255.255.255.0
global (inside) 1 209.165.202.128 netmask 255.255.255.224
```

However, a DNS server on the inside at IP address 192.168.1.5 cannot correctly reach the root name server on the outside at IP address 209.165.202.130.

To ensure that the inside DNS server can access the root name server, insert the following **static** command statement:

```
static (inside,outside) 209.165.202.129 192.168.1.5
```

The global address 209.165.202.129 provides a translated address for the inside server at IP address 192.168.1.5.

8. The following example enables PAT using the IP address at the outside interface in global configuration mode:

```
ip address outside 192.150.49.1
nat (inside) 1 0 0
global (outside) 1 interface
```

The interface IP address used for PAT is the address associated with the interface when the xlate is created. This is important for configuring DHCP, allowing for the DHCP retrieved address to be used for PAT.

When PAT is enabled on an interface, there should be no termination of TCP, UDP, and ICMP services. These services allow for termination at the PIX Firewall's outside interface.

9. To specify PAT using the IP address of an interface, specify the **interface** keyword.

```
global [(int_name)] nat_id address | interface
```

The following example enables PAT using the IP address at the outside interface in global configuration mode:

```
ip address outside 192.150.49.1
nat (inside) 1 0 0
global (outside) 1 interface
```

The interface IP address used for PAT is the address associated with the interface when the xlate (translation slot) is created. This is important for configuring DHCP, allowing for the DHCP retrieved address to be used for PAT.

When PAT is enabled on an interface, there should be no loss of TCP, UDP, and ICMP services. These services allow for termination at the PIX Firewall unit's outside interface.

10. To track usage among different subnets, you can specify multiple PATs using the following supported configurations:

The following example maps hosts on the internal network 10.1.0.0/24 to global address 192.168.1.1 and hosts on the internal network 10.1.1.1/24 to global address 209.165.200.225 in global configuration mode.

```
nat (inside) 1 10.1.0.0 255.255.255.0
nat (inside) 2 10.1.1.0 255.255.255.0
global (outside) 1 192.168.1.1 netmask 255.255.255.0
global (outside) 2 209.165.200.225 netmask 255.255.255.224
```

The following example configures two port addresses for setting up PAT on hosts from the internal network 10.1.0.0/16 in global configuration mode.

```
nat (inside) 1 10.1.0.0 255.255.0.0
global (outside) 1 209.165.200.225 netmask 255.255.255.224
global (outside) 1 192.168.1.1 netmask 255.255.255.0
```

With this configuration, address 192.168.1.1 will only be used when the port pool from address 209.165.200.225 is at maximum capacity.

Examples

The following example declares two global pool ranges and a PAT address. Then the **nat** command permits all inside users to start connections to the outside network:

```
global (outside) 1 209.165.201.1-209.165.201.10 netmask 255.255.255.224
global (outside) 1 209.165.201.12 netmask 255.255.255.224
Global 209.165.201.12 will be Port Address Translated
nat (inside) 1 0 0
clear xlate
```

The next example creates a global pool from two contiguous pieces of a Class C address and gives the perimeter hosts access to this pool of addresses to start connections on the outside interface:

```
global (outside) 1000 209.165.201.1-209.165.201.14 netmask 255.255.255.240
global (outside) 1000 209.165.201.17-209.165.201.30 netmask 255.255.255.240
nat (perimeter) 1000 0 0
```

help

Display help information. (Unprivileged mode.)

help

?

Usage Guidelines

The **help** or **?** command displays help information about all commands. You can view help for an individual command by entering the command name followed by a question mark or just the command name and pressing the **Enter** key.

If the **pager** command is enabled and when 24 lines display, the listing pauses, and the following prompt appears:

```
<--- More --->
```

The More prompt uses syntax similar to the UNIX **more** command:

- To view another screenful, press the Space bar.
- To view the next line, press the **Enter** key.
- To return to the command line, press the **q** key.

Examples

The following example shows how you can display help information by following the command name with a question mark:

```
enable ?  
usage: enable password <pw> [encrypted]
```

Help information is available on the core commands (not the **show**, **no**, or **clear** commands) by entering **?** at the command prompt:

```
?  
aaa          Enable, disable, or view TACACS+ or RADIUS  
             user authentication, authorization and accounting  
...
```

hostname

Change the host name in the PIX Firewall command line prompt. (Configuration mode.)

hostname *newname*

Syntax Description

newname New host name for the PIX Firewall prompt. This name can be up to 16 alphanumeric characters and mixed case.

Usage Guidelines

The **hostname** command changes the host name label on prompts. The default host name is pixfirewall.



Note

The change of the host name causes the change of the fully qualified domain name. Once the fully qualified domain name is changed, delete the RSA key pairs with the **ca zeroize rsa** command and delete related certificates with the **no ca identity ca_nickname** command.

Examples

The following example shows how to change a host name:

```
pixfirewall(config)# hostname spinner
spinner(config)# hostname pixfirewall
pixfirewall(config)#
```

http

New **http** commands allow you to enable the PIX Firewall HTTP server and specify the clients that are allowed to access it. (Configuration mode.)

http *ip_address* [*netmask*] [*if_name*]

no http *ip_address netmask if_name*

[no] http server enable

clear http

show http



Note

The HTTP server must be enabled to configure and monitor the PIX Firewall through PDM.

Syntax Description

http	Relating to the Hypertext Transfer Protocol.
<i>ip_address</i>	Specifies the host or network authorized to initiate an HTTP connection to the PIX Firewall.
<i>netmask</i>	Specifies the network mask for the http <i>ip_address</i> .
<i>if_name</i>	PIX Firewall interface name on which the host or network initiating the HTTP connection resides.
http server enable	Enables the HTTP server required to run PDM.
clear http	Removes all HTTP hosts and disables the server.
show http	Lists the allowed hosts and the enable state of the HTTP server.

Defaults

If you do not specify a netmask, the default is **255.255.255.255** regardless of the class of IP address. The default *if_name* is **inside**.

Usage Guidelines

Access from any host will be allowed if **0.0.0.0 0.0.0.0** (or **0 0**) is specified for *ip_address* and *netmask*.

Examples

The following **http** command example is used for one host:

```
http 16.152.1.11 255.255.255.255 outside
```

The following **http** command example is used for any host:

```
http 0.0.0.0 0.0.0.0 inside
```

icmp

Defines the control list for ICMP traffic that terminates at an interface. (Configuration mode.)

icmp permit | deny [**host**] *src_addr* [*src_mask*] [*type*] *int_name*

no icmp permit | deny [**host**] *src_addr* [*src_mask*] [*type*] *int_name*

clear icmp

show icmp

Syntax Description

permit deny	Permit or deny the ability to ping a PIX Firewall interface.
<i>src_addr</i>	Address that is either permitted or denied ability to ping an interface. Use host <i>src_addr</i> to specify a single host.
<i>src_mask</i>	Network mask. Specify if a network address is specified.
<i>type</i>	ICMP message type as described in Table 6-1 .
<i>int_name</i>	Interface name that can be pinged.

Usage Guidelines

Enable or disable pinging to an interface. With pinging disabled, the PIX Firewall cannot be detected on the network. The new **icmp** command implements this feature. This feature is also referred to as configurable proxy pinging.

To use the **icmp** command, configure an **access-list** command statement that permits or denies ICMP traffic that terminates at the PIX Firewall unit.

If the first matched entry is a permit entry, the ICMP packet continues to be processed. If the first matched entry is a deny entry or an entry is not matched, PIX Firewall discards the ICMP packet and generates the %PIX-3-313001 syslog message. An exception is when an ICMP **access-list** command statement is not configured; then, permit is assumed.

We recommend that you grant permission for ICMP unreachable message type (type 3). Denying ICMP unreachable messages disables ICMP Path MTU discovery, which can halt IPsec and PPTP traffic. See RFC 1195 and RFC 1435 for details about Path MTU Discovery.

The syslog message is as follows:

```
%PIX-3-313001: Denied ICMP type=type, code=code from source_address on interface interface_number
```

If this message appears, contact the peer's administrator.

ICMP Message Types

[Table 6-1](#) lists possible ICMP type values.

Table 6-1 ICMP Type Literals

ICMP Type	Literal
0	echo-reply
3	unreachable
4	source-quench

Table 6-1 ICMP Type Literals (continued)

ICMP Type	Literal
5	redirect
6	alternate-address
8	echo
9	router-advertisement
10	router-solicitation
11	time-exceeded
12	parameter-problem
13	timestamp-reply
14	timestamp-request
15	information-request
16	information-reply
17	mask-request
18	mask-reply
31	conversion-error
32	mobile-redirect

Examples

1. Deny all ping requests and permit all unreachable messages at the outside interface:

```
icmp permit any unreachable outside
```

The default behavior of the PIX Firewall is to deny ICMP messages to the outside interface.

2. Permit host 172.16.2.15 or hosts on subnet 172.22.1.0/16 to ping the outside interface:

```
icmp permit host 172.16.2.15 echo-reply outside
icmp permit 171.22.1.0 255.255.255.0 echo-reply outside
icmp permit any unreachable outside
```

interface

Identify network interface speed and duplex. (Configuration mode.)

interface *hardware_id* [*hardware_speed*] [**shutdown**]

clear interface

show interface

Syntax Description

<i>hardware_id</i>	Identifies the network interface type. Possible values are ethernet0 , ethernet1 to ethernetn , gb-ethernetn , fddi0 or fddi1 , token-ring0 , token-ring1 , to token-ringn , depending on how many network interfaces are in the PIX Firewall.
<i>hardware_speed</i>	Network interface speed (optional). Do not specify a <i>hardware_speed</i> for a FDDI interface. Possible Ethernet values are: 10baset —Set for 10 Mbps Ethernet half-duplex communication. 10full —Set for 10 Mbps Ethernet full-duplex communication. 100basetx —Set for 100 Mbps Ethernet half-duplex communication. 100full —Set for 100 Mbps Ethernet full-duplex communication. 1000sxfull —Set for 1000 Mbps Gigabit Ethernet full-duplex operation. 1000basesx —Set for 1000 Mbps Gigabit Ethernet half-duplex operation. 1000auto —Set for 1000 Mbps Gigabit Ethernet to auto-negotiate full or half duplex. We recommend that you do not use this option to maintain compatibility with switches and other devices in your network. au i—Set 10 for Mbps Ethernet half-duplex communication with an AUI cable interface. auto —Set Ethernet speed automatically. The auto keyword can only be used with the Intel 10/100 automatic speed sensing network interface card. We recommend that you do not use this option to maintain compatibility with switches and other devices in your network. bnc —Set for 10 Mbps Ethernet half-duplex communication with a BNC cable interface. Possible Token Ring values are as follows: 4mbps —4 Mbps data transfer speed. You can specify this as 4 . 16mbps —(Default) 16 Mbps data transfer speed. You can specify this as 16 .
show interface	The show interface command displays network interface information. The show interface command has been enhanced to include buffer counters. The buffer counters are only valid for Ethernet interfaces.
shutdown	Disable an interface.

Usage Guidelines

The **interface** command identifies the speed and duplex settings of the network interface boards. Use the **show interface** command to view information about the interface. The **show interface** command displays the packet drop count of Unicast RPF for each interface. This value appears as the “unicast rpf drops” counter.

The **clear interface** command clears all interface statistics except the number of input bytes. This command no longer shuts down all system interfaces. The **clear interface** command works with all interface types except gigabit Ethernet. The **clear interface** command also clears the packet drop count of Unicast RPF for all interfaces.

The **shutdown** option allows you to disable an interface. When you first install PIX Firewall, all interfaces are shut down by default. You must explicitly enable an interface by entering the command without the **shutdown** option. If the **shutdown** option does not exist in the command, packets are passed by the driver to and from the card.

If the **shutdown** option does exist, packets are dropped in either direction. Inserting a new card defaults to the default interface command containing the **shutdown** option. (That is, if you add a new card and then enter the **write memory** command, the **shutdown** option is saved into Flash memory for the interface.) When upgrading from a previous version to the current version, interfaces are enabled.

The configuration of the interface affects buffer allocation (the PIX Firewall will allocate more buffers for higher line speeds). Buffer allocation can be checked with the **show blocks** command.

For failover, set the Stateful Failover dedicated interface to 100 Mbps full duplex using the **100full** option to the **interface** command.

The **show interface** command reports “line protocol down” for BNC cable connections and for 3Com cards.



Note

Even though the default is to set automatic speed sensing for the interfaces with the **interface hardware_id auto** command, we recommend that you specify the speed of the network interfaces; for example, **10baset** or **100basetx**. This lets PIX Firewall operate in network environments that may include switches or other devices that do not handle auto sensing correctly.

Usage Notes

1. When you use the **interface token-ring** command, also use the **mtu** command to set the block size depending on the interface speed.
2. After changing an **interface** command, use the **clear xlate** command.

show interface Notes

The **show interface** command allows you to view network interface information for both Ethernet and Token Ring, depending on which is installed in your PIX Firewall. This is one of the first commands you should use when establishing network connectivity after installing a PIX Firewall.

The information in the **show interface** display is as follows:

- The ethernet, fddi, or token-ring interface strings indicate that you have used the **interface** command to configure the interface. The statement indicates either outside or inside and whether the interface is available (“up”) or not available (“down”).
- “line protocol up” means a working cable is plugged into the network interface. If the message is “line protocol down,” either the cable is incorrect or not plugged into the interface connector.
- Network interface type.

- Interrupt vector. It is acceptable for interface cards to have the same interrupts because PIX Firewall uses interrupts to get Token Ring information, but polls Ethernet cards.
- MAC address. Intel cards start with “i” and 3Com cards with “3c.”
- MTU (maximum transmission unit): The size in bytes that data can best be sent over the network.
- “*nn* packets input” Indicates that packets are being received in the PIX Firewall.
- “*nn* packets output” Indicates that packets are being sent from the PIX Firewall.
- Line duplex status: Half duplex indicates that the network interface switches back and forth between sending and receiving information; full duplex indicates that the network interface can send or receive information simultaneously.
- Line speed: **10baset** is listed as 10,000 Kbit; **100basetx** is listed as 100,000 Kbit.
- Interface problems:
 - no buffer, the PIX Firewall is out of memory or slowed down due to heavy traffic and cannot keep up with the received data.
 - runts are packets with less information than expected.
 - giants are packets with more information than expected.
 - input errors.
 - CRC (cyclic redundancy check) are packets that contain corrupted data (checksum error).
 - frame errors are framing errors.
 - overruns occur when the network interface card is overwhelmed and cannot buffer received information before more needs to be sent.
 - ignored and aborted errors are provided for future use, but are not currently checked; the PIX Firewall does not ignore or abort frames.
 - underruns occur when the PIX Firewall is overwhelmed and cannot get data fast enough to the network interface card.
 - unicast rpf drops—When packets sent to a single network destination using reverse path forwarding are dropped.
 - output errors—(maximum collisions). The number of frames not transmitted because the configured maximum number of collisions was exceeded. This counter should only increment during heavy network traffic.
 - collisions—(single and multiple collisions). The number of messages retransmitted due to an Ethernet collision. This usually occurs on an overextended LAN (Ethernet or transceiver cable too long, more than two repeaters between stations, or too many cascaded multiport transceivers). A packet that collides is counted only once by the output packets.
 - interface resets—The number of times an interface has been reset. If an interface is unable to transmit for three seconds, PIX Firewall resets the interface to restart transmission. During this interval, connection state is maintained. An interface reset can also happen when an interface is looped back or shut down.
 - babbles—Unused. (“babble” means that the transmitter has been on the interface longer than the time taken to transmit the largest frame.)
 - late collisions—The number of frames that were not transmitted because a collision occurred outside the normal collision window. A late collision is a collision that is detected late in the transmission of the packet. Normally, these should never happen. When two Ethernet hosts try to talk at once, they should collide early in the packet and both back off, or the second host should see that the first one is talking and wait.

If you get a late collision, a device is jumping in and trying to send on the Ethernet while the PIX Firewall is partly finished sending the packet. The PIX Firewall does not resend the packet, because it may have freed the buffers that held the first part of the packet.

This is not a real problem because networking protocols are designed to cope with collisions by resending packets. However, late collisions indicate a problem exists in your network. Common problems are large repeated networks and Ethernet networks running beyond the specification.

- deferred—The number of frames that were deferred before transmission due to activity on the link.
- lost carrier—The number of times the carrier signal was lost during transmission.
- no carrier—Unused.
- Gigabit interface cards do not provide information for the extended **show interface** command counters introduced in version 5.0(3).
- The **show interface** command has been enhanced to include eight additional status counters. The new counters are only valid for Ethernet interfaces. The following example shows the new output:

```
show interface
interface ethernet0 "outside" is up, line protocol is up
  Hardware is i82559 ethernet, address is 00aa.0000.003b
  IP address 209.165.201.7, subnet mask 255.255.255.224
  MTU 1500 bytes, BW 100000 Kbit half duplex
    1184342 packets input, 1222298001 bytes, 0 no buffer
    Received 26 broadcasts, 27 runts, 0 giants
    4 input errors, 0 CRC, 4 frame, 0 overrun, 0 ignored, 0 abort
    1310091 packets output, 547097270 bytes, 0 underruns, 0 unicast rpf drops
    0 output errors, 28075 collisions, 0 interface resets
    0 babbles, 0 late collisions, 117573 deferred
    0 lost carrier, 0 no carrier
input queue (curr/max blocks): hardware (128/128) software (0/1)
  output queue (curr/max blocks): hardware (0/2) software (0/1)
...
```

The counters in lines 9 to 11 are as follows:

- output errors—(maximum collisions). The number of frames not transmitted because the configured maximum number of collisions was exceeded. This counter should only increment during heavy network traffic.
- collisions—(single and multiple collisions). The number of messages retransmitted due to an Ethernet collision. This usually occurs on an overextended LAN (Ethernet or transceiver cable too long, more than two repeaters between stations, or too many cascaded multiport transceivers). A packet that collides is counted only once by the output packets.
- interface resets—The number of times an interface has been reset. If an interface is unable to transmit for three seconds, PIX Firewall resets the interface to restart transmission. During this interval, connection state is maintained. An interface reset can also happen when an interface is looped back or shut down.
- babbles—Unused. (“babble” means that the transmitter has been on the interface longer than the time taken to transmit the largest frame.)
- late collisions—The number of frames that were not transmitted because a collision occurred outside the normal collision window. A late collision is a collision that is detected late in the transmission of the packet. Normally, these should never happen. When two Ethernet hosts try to talk at once, they should collide early in the packet and both back off, or the second host should see that the first one is talking and wait.

- If you get a late collision, a device is jumping in and trying to send the packet on the Ethernet while the PIX Firewall is partly finished sending the packet. The PIX Firewall does not resend the packet, because it may have freed the buffers that held the first part of the packet. This is not a real problem because networking protocols are designed to cope with collisions by resending packets. However, late collisions indicate a problem exists in your network. Common problems are large repeated networks and Ethernet networks running beyond the specification.
- deferred—The number of frames that were deferred before transmission due to activity on the link.
- lost carrier—The number of times the carrier signal was lost during transmission.
- no carrier—Unused.

The counters in the last two lines are as follows:

- Input queue—The input (receive) hardware and software queue.
 - Hardware—(current and maximum blocks). The number of blocks currently present on the input hardware queue, and the maximum number of blocks previously present on that queue. In the example, there are currently 128 blocks on the input hardware queue, and the maximum number of blocks ever present on this queue was 128.
 - Software—(current and maximum blocks). The number of blocks currently present on the input software queue, and the maximum number of blocks previously present on that queue. In the example, there are currently 0 blocks on the input software queue, and the maximum number of blocks ever present on this queue was 1.
- Output queue—The output (transmit) hardware and software queue.
 - Hardware—(current and maximum blocks). The number of blocks currently present on the output hardware queue, and the maximum number of blocks previously present on that queue. In the example, there are currently 0 blocks on the output hardware queue, and the maximum number of blocks ever present on this queue was 2.
 - Software—(current and maximum blocks). The number of blocks currently present on the output software queue, and the maximum number of blocks previously present on that queue. In the example, there are currently 0 blocks on the output software queue, and the maximum number of blocks ever present on this queue was 1.

For Fast Ethernet and Gigabit Ethernet interfaces, the current and maximum count for the number of blocks on the input (receive) queue will always be the same. Currently the count is 128 for Fast Ethernet and 63 for Gigabit Ethernet. The number of blocks on the receive queue is always fixed.

Examples

The following example assigns names to each interface, enables auto detection for the interface parameters, and then shows interface activity:

```
show interface
interface ethernet0 "outside" is up, line protocol is up
  Hardware is i82557 ethernet, irq 10, address is 0060.7380.2f16
  IP address 209.165.201.1, subnet mask 255.255.255.224
  MTU 1500 bytes, BW 100000 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, 0 bytes, 0 underruns, 0 unicast rpf drops
    0 output errors, 28075 collisions, 0 interface resets
    0 babbles, 0 late collisions, 117573 deferred
    0 lost carrier, 0 no carrier
interface token-ring0 "inside" is up, line protocol is up
  Hardware is o3137 token-ring, irq 9, address is 0000.8326.72c6
  IP address 10.0.0.1, subnet mask 255.0.0.0
  MTU 8192 bytes, BW 16000 Kbit, Ring-speed: 16Mbps
    116 packets input, 27099 bytes, 0 no buffer
    Received 116 broadcasts, 0 runts, 0 giants
    0 input errors, 0 CRC, 116 frame, 0 overrun, 0 ignored, 0 abort
    3 packets output, 150 bytes, 0 underruns, 0 unicast rpf drops
    0 output errors, 28075 collisions, 0 interface resets
    0 babbles, 0 late collisions, 117573 deferred
    0 lost carrier, 0 no carrier
interface ethernet1 "DMZ" is up, line protocol is up
  Hardware is i82557 ethernet, irq 9, address is 00a0.c95d.0282
  IP address 127.0.0.1, 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
    0 packets output, 0 bytes, 0 underruns, 0 unicast rpf drops
    0 output errors, 28075 collisions, 0 interface resets
    0 babbles, 0 late collisions, 117573 deferred
    0 lost carrier, 0 no carrier
```

ip address

Identify addresses for network interfaces. (Configuration mode.)

```
ip address if_name ip_address [netmask]

ip address if_name dhcp [setroute]

show ip address if_name [dhcp]

show ip

clear ip

ip address outside dhcp [setroute] [retryretry_cnt]
```



Note

The **ip address** command has been enhanced to allow you to enter the number of times the PIX Firewall will poll for DHCP information.

Syntax Description

<i>if_name</i>	The internal or external interface name designated by the nameif command.
<i>ip_address</i>	PIX Firewall unit's network interface IP address.
<i>netmask</i>	Network mask of <i>ip_address</i> .
dhcp	Specifies PIX Firewall will use DHCP to poll for information. Enables the DHCP client feature on the specified interface.
outside	Interface from which the PIX Firewall will poll for information.
retry	Enables PIX Firewall to retry a poll for DHCP information.
<i>retry_cnt</i>	Specifies the number of times PIX Firewall will poll for DHCP information. The values available are 4 to 16. If no value is specified, the default is 4.
setroute	This option tells the PIX Firewall to set the default route using the default gateway parameter the DHCP server returns.
show ip	Display IP addresses assigned to the network interfaces.
clear ip	Resets all interface IP addresses to 127.0.0.1. The clear ip command does not affect the ip local pool or ip verify reverse-route commands.

Defaults

By default the PIX Firewall will not retry to poll for DHCP information. The default value for *retry_cnt* is 4.

Usage Guidelines

The **ip address** command lets you assign an IP address to each interface. Use the **show ip** command to view which addresses are assigned to the network interfaces. If you make a mistake while entering this command, re-enter the command with the correct information. The **clear ip** command resets all interface IP addresses to 127.0.0.1. The **clear ip** command does not affect the **ip local pool** or **ip verify reverse-route** commands.

**Note**

The **clear ip** command stops all traffic through the PIX Firewall unit.

After changing an **ip address** command, use the **clear xlate** command.

Always specify a network mask with the **ip address** command. If you let PIX Firewall assign a network mask based on the IP address, you may not be permitted to enter subsequent IP addresses if another interface's address is in the same range as the first address. For example, if you specify an inside interface address of 10.1.1.1 without specifying a network mask and then try to specify 10.1.2.2 for a perimeter interface mask, PIX Firewall displays the error message, "Sorry, not allowed to enter IP address on same network as interface *n*." To fix this problem, reenter the first command specifying the correct network mask.

Do not set the netmask to all 255s, such as 255.255.255.255. This stops access on the interface. Instead, use a network address of 255.255.255.0 for Class C addresses, 255.255.0.0 for Class B addresses, or 255.0.0.0 for Class A addresses.

The default address for an interface is 127.0.0.1.

PIX Firewall configurations using failover require a separate IP address for each network interface on the standby unit. The system IP address is the address of the active unit. When the **show ip** command is executed on the active unit, the current IP address is the same as the system IP address. When the **show ip** command is executed on the standby unit, the system IP address is the failover IP address configured for the standby unit.

The **ip address dhcp** command enables the DHCP client feature within the PIX Firewall. This command allows the PIX Firewall to be a DHCP client to a DHCP server that provides configuration parameters to the client. In this case, the configuration parameters the DHCP server provides is an IP address and a subnet mask to the interface on which the DHCP client feature is enabled. The optional **setroute** argument tells the PIX Firewall to set the default route using the default gateway parameter the DHCP server returns. If the **setroute** argument is configured, the **show route** command output shows the default route as being set by a DHCP server. To reset the interface and delete the DHCP lease from PIX Firewall, use the **clear ip** command. To clear the DHCP default route, use the **clear route static** command.

**Note**

Do not configure the PIX Firewall with a default route when using the **setroute** argument of the **ip address dhcp** command.

The **show ip address dhcp** command displays detailed information about the DHCP lease.

See "DHCP Client" within Chapter 7, "PIX Firewall System Management" in the *Cisco PIX Firewall and VPN Configuration Guide* for more information about the DHCP client feature.

Examples

The following is sample output for the **show ip** command:

```
show ip
System IP Addresses:
  ip address outside 209.165.201.2 255.255.255.224
  ip address inside 192.168.2.1 255.255.255.0
  ip address perimeter 192.168.70.3 255.255.255.0
Current IP Addresses:
  ip address outside 209.165.201.2 255.255.255.224
  ip address inside 192.168.2.1 255.255.255.0
  ip address perimeter 192.168.70.3 255.255.255.0
```

The Current IP Addresses are the same as the System IP Addresses on the failover active unit. When the primary unit fails, the Current IP Addresses become those of the standby unit.

The following is sample output for the **show ip address dhcp** command:

```
show ip address outside dhcp
```

```
Temp IP Addr:209.165.201.57 for peer on interface:outside  
Temp sub net mask:255.255.255.224  
DHCP Lease server:209.165.200.225, state:3 Bound  
DHCP Transaction id:0x4123  
Lease:259200 secs, Renewal:129600 secs, Rebind:226800 secs  
Temp default-gateway addr:209.165.201.1  
Next timer fires after:111797 secs  
Retry count:0, Client-ID:cisco-0000.0000.0000-outside
```

```
ip address outside dhcp retry 10
```

Related Commands

- [dhcpd](#)

ip audit

Configure IDS signature use. (Configuration mode.)

ip audit attack [action [alarm] [drop] [reset]]

no ip audit attack

show ip audit attack

ip audit info [action [alarm] [drop] [reset]]

no ip audit info

show ip audit info

ip audit interface *if_name* *audit_name*

no ip audit interface [*if_name*]

show ip audit interface

ip audit name *audit_name* **attack** [action [alarm] [drop] [reset]]

no ip audit name *audit_name* [**attack**]

show ip audit name [name [info | attack]]

ip audit name *audit_name* **info** [action [alarm] [drop] [reset]]

no ip audit name *audit_name* [**info**]

show ip audit name

ip audit signature *signature_number* **disable**

no ip audit signature *signature_number*

show ip audit signature [*signature_number*]

clear ip audit [name | signature | interface | attack | info]

Syntax Description	
audit attack	Specify the default actions to be taken for attack signatures.
audit info	Specify the default actions to be taken for informational signatures.
audit interface	Apply an audit specification or policy (via the ip audit name command) to an interface.
audit name	Specify informational signatures, except those disabled or excluded by the ip audit signature command, as part of the policy.
audit signature	Specify which messages to display, attach a global policy to a signature, and disable or exclude a signature from auditing.
action actions	The alarm option indicates that when a signature match is detected in a packet, PIX Firewall reports the event to all configured syslog servers. The drop option drops the offending packet. The reset option drops the offending packet and closes the connection if it is part of an active connection. The default is alarm .
clear	Resets name, signature, interface, attack, info to their default values.
<i>audit_name</i>	Audit policy name viewed with the show ip audit name command.
<i>signature_number</i>	IDS signature number.

Usage Guidelines

Cisco Intrusion Detection System (Cisco IDS) is an IP-only feature that provides some level of flexibility for the user to customize the amount of traffic that needs to be audited and logged.

The Cisco IDS features provide the following:

- Traffic auditing. Application level signatures will only be audited as part of an active session.
- Apply the audit to an interface.
- Support different audit policies. Traffic matching a signature triggers a range of configurable actions.
- Disable the signature audit.
- Enable IDS and still disable actions of a signature class (informational, attack).

Auditing is performed by looking at the IP packets as they arrive at an input interface, if a packet triggers a signature and the configured action does not drop the packet, then the same packet can trigger other signatures.

PIX Firewall supports both inbound and outbound auditing.

For a complete list of supported Cisco IDS signatures, their wording, and whether they are attack or informational messages, refer to *Cisco PIX Firewall System Log Messages*.

Refer to the *Cisco Secure Intrusion Detection System Version 2.2.1 User Guide* for detailed information on each signature. You can view the “NSDB and Signatures” chapter of this guide at the following website:

http://www.cisco.com/en/US/products/sw/secursw/ps5052/products_user_guide_chapter09186a00800d924d.html

The **ip audit** commands are described in the sections that follow.

ip audit attack

The **ip audit attack [action [alarm] [drop] [reset]]** command specifies the default actions to be taken for attack signatures. An audit policy (audit rule) defines the attributes for all signatures that can be applied to an interface along with a set of actions. Using an audit policy may limit the traffic that is audited or specify actions to be taken when the signature matches. Each audit policy is identified by a

name and can be defined for informational or attack signatures. Each interface can have two policies; one for informational signatures and one for attack signatures. If a policy is defined without actions, then the configured default actions will take effect. Each policy requires a different name.

The **no ip audit attack** command resets the action to be taken for attack signatures to the default action. The **show ip audit attack** command displays the default attack actions.

ip audit info

The **ip audit info [action [alarm] [drop] [reset]]** command specifies the default action to be taken for signatures classified as informational signatures.

The **no ip audit info** command sets the action to be taken for signatures classified as informational and reconnaissance to the default action. The **show ip audit info** displays the default informational actions.

To cancel event reactions, specify the **ip audit info** command without an **action** option.

ip audit interface

The **ip audit interface if_name audit_name** command applies an audit specification or policy (via the **ip audit name** command) to an interface. The **no ip audit interface [if_name]** command removes a policy from an interface. The **show ip audit interface** command displays the interface configuration.

ip audit name

The **ip audit name audit_name info [action [alarm] [drop] [reset]]** command specifies the informational signatures except those disabled or excluded by the **ip audit signature** command that are considered part of the policy. The **no ip audit name audit_name [info]** command removes the audit policy *audit_name*. The **show ip audit name [name [info|attack]]** command displays all audit policies or specific policies referenced by name and possibly type.

ip audit signature

The **ip audit signature signature_number disable** command specifies which messages to display, attaches a global policy to a signature, and disables or excludes a signature from auditing. The **no ip audit signature signature_number** command removes the policy from a signature. Used to reenable a signature. The **show ip audit signature [signature_number]** displays disabled signatures.

Supported IDS Signatures

PIX Firewall lists the following single-packet IDS signature messages: 1000-1006, 1100, 1102, 1103, 2000-2012, 2150, 2151, 2154, 3040-3042, 4050-4052, 6050-6053, 6100-6103, 6150-6155, 6175, 6180, and 6190. All signature messages are not supported by PIX Firewall in this release. IDS syslog messages all start with *%PIX-4-4000nn* and have the following format:

```
%PIX-4-4000nn IDS:sig_num sig_msg from faddr to laddr on interface int_name
```

For example:

```
%PIX-4-400013 IDS:2003 ICMP redirect from 10.4.1.2 to 10.2.1.1 on interface dmz
%PIX-4-400032 IDS:4051 UDP Snork attack from 10.1.1.1 to 192.168.1.1 on interface outside
```

Options:

<i>sig_num</i>	The signature number.
<i>sig_msg</i>	The signature message—approximately the same as the Cisco IDS signature message.
<i>faddr</i>	The IP address of the foreign host initiating the attack. (“Foreign” is relative; attacks can be perpetrated either from outside to an inside host, or from the inside to an outside host.)

<code>laddr</code>	The IP address of the local host to which the attack is directed. (“Local” is relative; attacks can be perpetrated either from outside to an inside host, or from the inside to an outside host.)
--------------------	---

<code>int_name</code>	The name of the interface on which the signature originated.
-----------------------	--

Examples

Disable signature 6102 globally:

```
ip audit signature 6102 disable
```

Specify default informational actions:

```
ip audit name attack1 info
```

Specify an attack policy:

```
ip audit name attack2 attack action alarm drop reset
```

Apply a policy to an interface:

```
ip audit interface outside attack1
```

```
ip audit interface inside attack2
```

ip local pool

Identify addresses for a local pool. (Configuration mode.)

ip local pool *pool_name pool_start-address[-pool_end-address]*

no ip local pool *pool_name pool_start-address[-pool_end-address]*

show ip local pool *pool_name ip_address[-ip_address]*

show ip local pool *pool_name ip_address[-ip_address]*

clear ip local pool *pool_name ip_address[-ip_address]*

Syntax Description

ip local pool	Creates a pool of local addresses to be used for assigning dynamic IP addresses to remote VPN clients. The address range of this pool of local addresses must not overlap with any command statement that allows you to specify an IP address.
no ip local pool	Deletes a local address pool.
show ip local pool	Shows usage information about a local address pool.
clear ip local pool	Resets IP addresses in a local pool to their default values.
<i>pool_name</i>	Local pool name.
<i>pool_start_address</i>	Local pool IP address range.
<i>pool_end_address</i>	
<i>ip_address</i>	Single IP address or used with <i>-ip_address</i> to specify a list of IP addresses.
<i>-ip_address</i>	Optional ending IP address.

Usage Guidelines

The **ip local pool** command allows you to create a pool of local addresses to be used for assigning dynamic IP addresses to remote VPN clients. The address range of this pool of local addresses must not overlap with any command statement that allows you to specify an IP address. To delete an address pool, use the **no ip local pool** command. Use the **show ip local pool** command to view usage information about the pool of local addresses.

When a pool of addresses set by the **ip local pool** command is empty, the following syslog message appears:

```
%PIX-4-404101: ISAKMP: Failed to allocate address for client from pool poolname
```

To reference this pool of local addresses, use the **isakmp client configuration address-pool** command. Refer to the *Cisco PIX Firewall and VPN Configuration Guide* for information on the **isakmp** command.

Examples

The following example creates a pool of IP addresses and then displays the pool contents:

```
ip local pool mypool 10.0.0.10-10.0.0.20
show ip local pool mypool
```

Pool	Begin	End	Free	In use
mypool	10.0.0.10	10.0.0.20	11	0

Available Addresses:

```
10.0.0.10
10.0.0.11
10.0.0.12
10.0.0.13
10.0.0.14
10.0.0.15
10.0.0.16
10.0.0.17
10.0.0.18
10.0.0.19
10.0.0.20
```

ipsec

The **ipsec** command is a shortened form of the **crypto ipsec** command. (Configuration mode.)

clear ipsec

show ipsec

Usage Guidelines

The **clear ipsec** command removes all **ipsec** commands from the configuration. The **show ipsec** command lists all the **ipsec** commands in the configuration.



Note

See the [crypto ipsec](#) command page for information on all other command options and examples.

ip verify reverse-path

Implement Unicast RPF IP spoofing protection. (Configuration mode.)

ip verify reverse-path interface *int_name*

no ip verify reverse-path interface *int_name*

show ip verify [**reverse-path** [**interface** *int_name*]]

clear ip verify

clear ip verify reverse-path interface *int_name*

Syntax Description

ip verify reverse-path interface	Protects an individual interface against IP spoofing by enabling both ingress and egress filtering to verify addressing and route integrity. This command depends upon a default route previously defined in the configuration. See RFC 2267 for more information.
no ip verify reverse-path interface	Disables ip verify reverse-path filtering for an individual interface from the configuration.
show ip verify	Displays a list of the ip verify commands in the configuration, including ip verify reverse-path for all interfaces or one interface.
clear ip verify	Removes ip verify commands from the configuration.
clear ip verify reverse-path interface	Removes ip verify reverse-path commands for an individual interface from the configuration.
<i>int_name</i>	Name of an interface you want to protect from a DoS attack.

Usage Guidelines

The **ip verify reverse-path** command lets you specify which interfaces to protect from an IP spoofing attack using network ingress and egress filtering, which is described in RFC 2267. This command is disabled by default and provides Unicast Reverse Path Forwarding (RPF) functionality for the PIX Firewall. The **show ip verify** command lists the **ip verify** commands in the configuration. The **clear ip verify** command removes **ip verify** commands from the configuration. Unicast RPF is a unidirectional input function that screens inbound packets arriving on an interface. Outbound packets are not screened.

Because of the danger of IP spoofing in the IP protocol, measures need to be taken to reduce this risk when possible. Unicast RPF, or reverse route lookups, prevents such manipulation under certain circumstances.



Note

The **ip verify reverse-path** command depends on the existence of a default route statement in the configuration for the outside interface that has 0.0.0.0 0.0.0.0 in the **route** command statement for the IP address and network mask.

The **ip verify reverse-path** command provides both ingress and egress filtering. Ingress filtering checks inbound packets for IP source address integrity, and is limited to addresses for networks in the enforcing entity's local routing table. If the incoming packet does not have a source address represented by a route, then it is impossible to know whether the packet has arrived on the best possible path back to its origin. This is often the case when routing entities cannot maintain routes for every network.

Egress filtering verifies that packets destined for hosts outside the managed domain have IP source addresses verifiable by routes in the enforcing entity's local routing table. If an exiting packet does not arrive on the best return path back to the originator, then the packet is dropped and the activity is logged. Egress filtering prevents internal users from launching attacks using IP source addresses outside of the local domain because most attacks use IP spoofing to hide the identity of the attacking host. Egress filtering makes the task of tracing the origin of an attack much easier. When employed, egress filtering enforces what IP source addresses are obtained from a valid pool of network addresses. Addresses are kept local to the enforcing entity and are therefore easily traceable.

Unicast RPF is implemented as follows:

- ICMP packets have no session so each packet is checked.
- UDP and TCP have sessions, so the initial packet requires a reverse route lookup. Subsequent packets arriving during the session are checked using an existing state maintained as part of the session. Non-initial packets are checked to ensure they arrived on the same interface used by the initial packet.

**Note**

Before using this command, add static **route** command statements for every network that can be accessed on the interfaces you wish to protect. Only enable this command if routing is fully specified. Otherwise, PIX Firewall will stop traffic on the interface you specify if routing is not in place.

Use the **show interface** command to view the number dropped packets, which appears in the “unicast rpf drops” counter.

Examples

The following example protects traffic between the inside and outside interfaces and provides **route** command statements for two networks 10.1.2.0 and 10.1.3.0 that connect to the inside interface via a hub:

```
ip address inside 10.1.1.1 255.255.0.0
route inside 10.1.2.0 255.255.0.0 10.1.1.1 1
route inside 10.1.3.0 255.255.0.0 10.1.1.1 1
ip verify reverse-path interface outside
ip verify reverse-path interface inside
```

The **ip verify reverse-path interface outside** command statement protects the outside interface from network ingress attacks from the Internet, whereas the **ip verify reverse-path interface inside** command statement protects the inside interface from network egress attacks from users on the internal network.

isakmp

Negotiate IPSec security associations and enable IPSec secure communications.
(Configuration mode.)

```

isakmp client configuration address-pool local pool-name [interface-name]
no isakmp client configuration address-pool local pool-name
isakmp enable interface-name
no isakmp enable interface-name
isakmp identity address | hostname
no isakmp identity address | hostname
isakmp keepalive seconds [retry seconds]
isakmp key keystring address peer-address [netmask mask] [no-xauth] [no-config-mode]
no isakmp key keystring address peer-address [netmask mask][no-xauth] [no-config-mode]
isakmp peer fqdn fqdn no-xauth no-config-mode
no isakmp peer fqdn fqdn no-xauth no-config-mode
isakmp policy priority authentication pre-share | rsa-sig
no isakmp policy priority authentication pre-share | rsa-sig
isakmp policy priority encryption des | 3des
no isakmp policy priority encryption des | 3des
isakmp policy priority group 1 | 2
no isakmp policy priority group 1 | 2
isakmp policy priority hash md5 | sha
no isakmp policy priority hash md5 | sha
isakmp policy priority lifetime seconds
no isakmp policy priority lifetime seconds
show isakmp policy
show isakmp sa
clear [crypto] isakmp sa
clear isakmp

```

Syntax Description

<i>pool-name</i>	Specify the name of a local address pool to allocate the dynamic client IP.
<i>interface-name</i>	The name of the interface on which to enable ISAKMP negotiation.
<i>peer-address</i>	Specify the IP address of the IPsec peer.
<i>peer-hostname</i>	Specify the host name of the IPsec peer.
key <i>keystring</i>	Specify the authentication pre-shared key. Use any combination of alphanumeric characters up to 128 bytes. This pre-shared key must be identical at both peers.
address <i>peer-address</i>	Specify the IPsec peer's IP address for the pre-shared key.
netmask <i>mask</i>	(Optional) The netmask of 0.0.0.0. can be entered as a wildcard indicating the key could be used for any peer that does not have a key associated with its specific IP address.
no-xauth	This is only to be used if you enabled the Xauth feature, and you have an IPsec peer that is a gateway. This option associates a given pre-shared key with a gateway and allows an exception to the Xauth feature enabled by the crypto map client authentication command.
no-config-mode	This is only to be used if you enabled the IKE Mode Configuration feature, and you have an IPsec peer that is a gateway. This option associates a given pre-shared key with a gateway and allows an exception to the IKE Mode Configuration feature enabled by the crypto map client configuration address command.
fqdn <i>fqdn</i>	The fully qualified domain name of the peer. This is used to identify a peer that is a security gateway.
policy <i>priority</i>	Uniquely identifies the IKE policy and assigns a priority to the policy. Use an integer from 1 to 65,534, with 1 being the highest priority and 65,534 the lowest.
authentication <i>pre-share</i>	Specify pre-shared keys as the authentication method.
authentication <i>rsa-sig</i>	Specify RSA signatures as the authentication method. RSA signatures provide non-repudiation for the IKE negotiation. This basically means you can prove to a third party whether you had an IKE negotiation with the peer.
encryption <i>des</i>	Specify 56-bit DES-CBC as the encryption algorithm to be used in the IKE policy.
encryption <i>3des</i>	Specify that the Triple DES encryption algorithm is to be used in the IKE policy.
group 1	Specify that the 768-bit Diffie-Hellman group is to be used in the IKE policy. This is the default value.
group 2	Specify that the 1024-bit Diffie-Hellman group is to be used in the IKE policy.
hash <i>md5</i>	Specify MD5 (HMAC variant) as the hash algorithm to be used in the IKE policy.
hash <i>sha</i>	Specify SHA-1 (HMAC variant) as the hash algorithm to be used in the IKE policy. This is the default hash algorithm.
lifetime <i>seconds</i>	Specify how many seconds each security association should exist before expiring. Use an integer from 120 to 86,400 seconds (one day).

Usage Guidelines

The sections that follow describe each **isakmp** command.

isakmp client configuration address-pool local

The **isakmp client configuration address-pool local** command is used to configure the IP address local pool to reference IKE. Use the **no crypto isakmp client configuration address-pool local** command to restore to the default value.

Before using this command, use the **ip local pool** command to define a pool of local addresses to be assigned to a remote IPsec peer.

Examples

The following example references IP address local pools to IKE with “mypool” as the pool-name:

```
isakmp client configuration address-pool local mypool outside
```

isakmp enable

The **isakmp enable** command is used to enable ISAKMP negotiation on the interface on which the IPsec peer will communicate with the PIX Firewall. ISAKMP is enabled by default. Use the **no isakmp enable** command to disable IKE.

Examples

The following example shows how to disable IKE on the inside interface:

```
no isakmp enable inside
```

isakmp identity address | hostname

To define the ISAKMP identity the PIX Firewall uses when participating in the IKE protocol, use the **isakmp identity address | hostname** command. Use **no isakmp identity address | hostname** command to reset the ISAKMP identity to the default value of IP address.

When two peers use IKE to establish IPsec security associations, each peer sends its ISAKMP identity to the remote peer. It will send either its IP address or host name depending on how each has its ISAKMP identity set. By default, the PIX Firewall unit’s ISAKMP identity is set to the IP address. As a general rule, set the PIX Firewall and its peer’s identities in the same way to avoid an IKE negotiation failure. This failure could be due to either the PIX Firewall or its peer not recognizing its peer’s identity.

**Note**

If you are using RSA signatures as your authentication method in your IKE policies, we recommend that you set each participating peer’s identity to hostname. Otherwise, the ISAKMP security association to be established during Phase 1 of IKE may fail.

The following example uses pre-shared keys between the two PIX Firewall units (PIX Firewall 1 and PIX Firewall 2) that are peers, and sets both their ISAKMP identities to host name.

At the PIX Firewall 1, the ISAKMP identity is set to hostname:

```
isakmp identity hostname
```

At the PIX Firewall 2, the ISAKMP identity is set to hostname:

```
isakmp identity hostname
```

isakmp keepalive seconds [retry seconds]

The keepalive interval can be between 10 and 3600 seconds. The retry interval can be between 2 and 10 seconds, with the default being 2 seconds. The retry interval is the interval between retries after a keepalive response has not been received. You can specify the keepalive interval without specifying the retry interval, but cannot specify the retry interval without specifying the keepalive interval.

isakmp key address

To configure a pre-shared authentication key and associate the key with an IPsec peer address or host name, use the **isakmp key address** command. Use the **no isakmp key address** command to delete a pre-shared authentication key and its associated IPsec peer address.

You would configure the pre-shared key at both peers whenever you specify pre-shared key in an IKE policy. Otherwise, the policy cannot be used because it will not be submitted for matching by the IKE process.

A netmask of 0.0.0.0. can be entered as a wildcard indicating that any IPsec peer with a given valid pre-shared key is a valid peer.

**Note**

The PIX Firewall or any IPsec peer can use the same authentication key with multiple peers, but this is not as secure as using a unique authentication key between each pair of peers.

Configure a pre-shared key associated with a given security gateway to be distinct from a wildcard, pre-shared key (pre-shared key plus a netmask of 0.0.0.0) used to identify and authenticate the remote VPN clients.

The **no-xauth** or **no-config-mode** command options are to be used only if the following criteria are met:

- You are using the pre-shared key authentication method within your IKE policy.
- The security gateway and VPN client peers terminate on the same interface.
- The Xauth or IKE Mode Configuration feature is enabled for VPN client peers.

The **isakmp key keystring address ip-address [no-xauth] [no-config-mode]** command allows you to configure a pre-shared authentication key, associate the key with a given security gateway's address, and make an exception to the enabled Xauth feature, IKE Mode Configuration feature, or both (the most common case) for this peer.

Both the Xauth and IKE Mode Configuration features are specifically designed for remote VPN clients. The Xauth feature allows the PIX Firewall to challenge the peer for a username and password during IKE negotiation. The IKE Mode Configuration enables the PIX Firewall to download an IP address to the peer for dynamic IP address assignment. Most security gateways do not support the Xauth and IKE Mode Configuration features.

If you have the **no-xauth** command option configured, the PIX Firewall will not challenge the peer for a username and password. Similarly, if you have the **no-config-mode** command option configured, the PIX Firewall will not attempt to download an IP address to the peer for dynamic IP address assignment.

Use the **no key keystring address ip-address [no-xauth] [no-config-mode]** command to disable the **key keystring address ip-address [no-xauth] [no-config-mode]** command that you previously enabled.

See the **crypto map client authentication** command within the **crypto map** command page for more information about the Xauth feature. See the **crypto map client configuration address** command within the **crypto map** command page for more information about the IKE Mode Config feature.

The following example shows “sharedkeystring” as the authentication key to share between the PIX Firewall and its peer specified by an IP address of 10.1.0.0:

```
isakmp key sharedkeystring address 10.1.0.0
```

The following example shows use of a wildcard, pre-shared key. The “sharedkeystring” is the authentication key to share between the PIX Firewall and its peer (in this case a VPN client) specified by an IP address of 0.0.0.0. and a netmask of 0.0.0.0.

```
isakmp key sharedkeystring address 0.0.0.0 netmask 0.0.0.0
```

The following example shows use of the command options **no-xauth** and **no-config-mode** in relation to three PIX Firewall peers that are security gateways. These security gateways terminate IPsec on the same interface as the VPN clients. Both the Xauth and IKE Mode Config features are enabled. This means there is a need to make an exception to these two features for each security gateway. The example shows each security gateway peer has a unique pre-shared key to share with the PIX Firewall. The peers' IP addresses are 10.1.1.1, 10.1.1.2, 10.1.1.3, and the netmask of 255.255.255.255 is specified.

```
isakmp key secretkey1234 address 10.1.1.1 netmask 255.255.255.255 no-xauth no-config-mode
isakmp key secretkey4567 address 10.1.1.2 netmask 255.255.255.255 no-xauth no-config-mode
isakmp key secretkey7890 address 10.1.1.3 netmask 255.255.255.255 no-xauth no-config-mode
```

isakmp peer fqdn no-xauth | no-config-mode

The **isakmp peer fqdn fqdn no-xauth | no-config-mode** command is to be used only if the following criteria are met:

- You are using the RSA signatures authentication method within your IKE policy.
- The security gateway and VPN client peers terminate on the same interface.
- The Xauth or IKE Mode Configuration feature is enabled for VPN client peers.

The **isakmp peer fqdn fqdn no-xauth | no-config-mode** command allows you identify a peer that is a security gateway and make an exception to the enabled Xauth feature, IKE Mode Configuration feature, or both (the most common case) for this peer.

Both the Xauth and IKE Mode Configuration features are specifically designed for remote VPN clients. The Xauth feature allows the PIX Firewall to challenge the peer for a username and password during IKE negotiation. The IKE Mode Configuration feature enables the PIX Firewall to download an IP address to the peer for dynamic IP address assignment. Most security gateways do not support the Xauth and IKE Mode Configuration features.

If you have the **no-xauth** command option configured, the PIX Firewall will not challenge the peer for a username and password. Similarly, if you have the **no-config-mode** command option configured, the PIX Firewall will not attempt to download an IP address to the peer for dynamic IP address assignment.



Note

If you are using RSA signatures as your authentication method in your IKE policies, we recommend that you set each participating peer's identity to hostname using the **isakmp identity hostname** command. Otherwise, the ISAKMP security association to be established during Phase 1 of IKE may fail.

Use the **no isakmp peer fqdn fqdn no-xauth | no-config-mode** command to disable the **isakmp peer fqdn fqdn no-xauth | no-config-mode** command that you previously enabled.

See the [crypto map client authentication](#) within the [crypto map](#) command page for more information about the Xauth feature. See the [crypto map client configuration address](#) command within the [crypto map](#) command page for more information about the IKE Mode Config feature.

The following example shows use of the command options **no-xauth** and **no-config-mode** in relation to three PIX Firewall peers that are security gateways. These security gateways terminate IPsec on the same interface as the VPN clients. Both the Xauth and IKE Mode Config features are enabled. This means there is a need to make an exception to these two features for each security gateway. Each security gateway peer's fully qualified domain name is specified.

```
isakmp peer fqdn hostname1.example.com no-xauth no-config-mode
isakmp peer fqdn hostname2.example.com no-xauth no-config-mode
isakmp peer fqdn hostname3.example.com no-xauth no-config-mode
```

isakmp policy authentication

The **isakmp policy authentication** command allows you to specify the authentication method within an IKE policy. IKE policies define a set of parameters to be used during IKE negotiation.

If you specify RSA signatures, you must configure the PIX Firewall and its peer to obtain certificates from a CA. If you specify pre-shared keys, you must separately configure these pre-shared keys within the PIX Firewall and its peer.

Use the **no isakmp policy authentication** command to reset the authentication method to the default value of RSA signatures.

The following example shows use of the **isakmp policy authentication** command. This example sets the authentication method of rsa-signatures to be used within the IKE policy with the priority number of 40.

```
isakmp policy 40 authentication rsa-sig
```

isakmp policy encryption

To specify the encryption algorithm within an IKE policy, use the **isakmp policy encryption** command. IKE policies define a set of parameters to be used during IKE negotiation.

DES and 3DES are the two encryption algorithm options available.

Use the **no isakmp policy encryption** command to reset the encryption algorithm to the default value, which is **des**.

The following example shows use of the **isakmp policy encryption** command. This example sets the 3DES algorithm to be used within the IKE policy with the priority number of 40.

```
isakmp policy 40 encryption 3des
```

isakmp policy group

Use the **isakmp policy group** command to specify the Diffie-Hellman group to be used in an IKE policy. IKE policies define a set of parameters to be used during IKE negotiation.

There are two group options: 768-bit or 1024-bit. The 1024-bit Diffie Hellman provides stronger security, but it requires more CPU time to execute.

Use the **no isakmp policy group** command to reset the Diffie-Hellman group identifier to the default value of group 1, 768-bit Diffie Hellman.

The following example shows use of the **isakmp policy group** command. This example sets group 2, the 1024-bit Diffie Hellman, to be used within the IKE policy with the priority number of 40.

```
isakmp policy 40 group 2
```

isakmp policy hash

Use the **isakmp policy hash** command to specify the hash algorithm to be used in an IKE policy. IKE policies define a set of parameters to be used during IKE negotiation.

There are two hash algorithm options: SHA-1 and MD5. MD5 has a smaller digest and is considered to be slightly faster than SHA-1.

To reset the hash algorithm to the default value of SHA-1, use the **no isakmp policy hash** command.

The following example shows use of the **isakmp policy hash** command. This example sets the MD5 hash algorithm to be used within the IKE policy with the priority number of 40.

```
isakmp policy 40 hash md5
```

isakmp policy lifetime

To specify the lifetime of an IKE security association before it expires, use the **isakmp policy lifetime** command. Use the **no isakmp policy lifetime** command to reset the security association lifetime to the default value of 86,400 seconds (one day).

When IKE begins negotiations, it looks to agree upon the security parameters for its own session. The agreed-upon parameters are then referenced by a security association at each peer. The security association is retained by each peer until the security association's lifetime expires. Before a security association expires, it can be reused by subsequent IKE negotiations, which can save time when setting up new IPSec security associations. New security associations are negotiated before current security associations expire.

To save setup time for IPSec, configure a longer IKE security association lifetime. However, the shorter the lifetime (up to a point), the more secure the IKE negotiation is likely to be.

**Note**

When PIX Firewall initiates an IKE negotiation between itself and an IPSec peer, an IKE policy can be selected only if the lifetime of the peer's policy is shorter than or equal to the lifetime of its policy. Then, if the lifetimes are not equal, the shorter lifetime will be selected.

The following example shows use of the **isakmp policy lifetime** command. This example sets the lifetime of the IKE security association to 50,400 seconds (14 hours) within the IKE policy with the priority number of 40.

```
isakmp policy 40 lifetime 50400
```

show isakmp policy

To view the parameters for each IKE policy including the default parameters, use the **show isakmp policy** command.

The following is sample output from the **show isakmp policy** command after two IKE policies were configured (with priorities 70 and 90 respectively):

```
show isakmp policy
```

```
Protection suite priority 70
  encryption algorithm:  DES - Data Encryption Standard (56 bit keys)
  hash algorithm:       Message Digest 5
  authentication method: Rivest-Shamir-Adleman Signature
  Diffie-Hellman group: #2 (1024 bit)
  lifetime:             5000 seconds, no volume limit
Protection suite priority 90
  encryption algorithm:  DES - Data Encryption Standard (56 bit keys)
  hash algorithm:       Secure Hash Standard
  authentication method: Pre-Shared Key
  Diffie-Hellman group: #1 (768 bit)
  lifetime:             10000 seconds, no volume limit
Default protection suite
  encryption algorithm:  DES - Data Encryption Standard (56 bit keys)
  hash algorithm:       Secure Hash Standard
  authentication method: Rivest-Shamir-Adleman Signature
  Diffie-Hellman group: #1 (768 bit)
  lifetime:             86400 seconds, no volume limit
```

**Note**

Although the output shows “no volume limit” for the lifetimes, you can currently only configure a time lifetime (such as 86,400 seconds); volume limit lifetimes are not currently configurable.

show isakmp sa

To view all current IKE security associations between the PIX Firewall and its peer, use the **show isakmp sa** command.

The following is sample output from the **show isakmp sa** command after IKE negotiations were successfully completed between the PIX Firewall and its peer:

```
show isakmp sa
      dst          src          state    pending    created
  16.132.40.2     16.132.30.2    QM_IDLE      0          1
```

clear isakmp

The **clear isakmp** command removes all **isakmp** command statements from the configuration.

clear [crypto] isakmp sa

The **clear [crypto] isakmp sa** command deletes active IKE security associations. The keyword **crypto** is optional.

isakmp policy

The **isakmp policy** command allows you to negotiate IPsec security associations and enable IPsec secure communications. (Configuration Mode.)

isakmp policy [*priority*] **group 2**

Syntax Description

<i>priority</i>	Uniquely identifies the Internet Key Exchange (IKE) policy and assigns a priority to the policy. Use an integer from 1 to 65,534, with 1 being the highest priority and 65,534 the lowest.
group 2	Specifies that the 1024-bit Diffie-Hellman group 2 be used in the IKE policy.

Usage Guidelines

Cisco VPN Client version 3.0 uses Diffie-Hellman group 2 and Cisco VPN Client 3000 version 2.5 uses Diffie-Hellman group 1. If you are using Cisco VPN Client version 3.0, configure Diffie-Hellman group 2 by using the **isakmp policy** command.



Note

The Cisco VPN client version 3.0 does not support the **crypto map map-name client configuration address initiate | respond** command.

Examples

The following is an example of the **isakmp policy** command.

```
isakmp policy 93 group 2
```

kill

Terminate a Telnet session. (Privileged mode.)

```
kill telnet_id
```

Syntax Description

telnet_id Telnet session ID.

Usage Guidelines

The **kill** command terminates a Telnet session. Use the **who** command to view the Telnet session ID value. When you kill a Telnet session, the PIX Firewall lets any active commands terminate and then drops the connection without warning the user.

Examples

The following is sample output from the **show who** command, which is used to list the active Telnet sessions, and the use of the **kill** command to end Telnet session 2:

```
show who
2: From 10.10.54.0
kill 2
```

Related Commands

- **show who**
- **telnet**

local-host (clear and show)

View local host network states. (Privileged mode (**show**), configuration mode (**clear**).)

clear local-host [*ip_address*]

show local-host [*ip_address*]

Syntax Description

ip_address Local host IP address.

Usage Guidelines

The **show local-host** command allows you to view the network states of local hosts. Local hosts are any hosts on the same subnet as an internal PIX Firewall interface (not the outside interface). Hosts beyond the next hop routers are not affected by this command.

This command allows you to show the translation and connection slots for the local hosts, or stop all traffic on these hosts. This command provides information for hosts configured with the **nat 0** command when normal translation and connection states may not apply.

Use the *ip_address* option to limit the display to a single host. The **clear local-host** command clears the information displayed for the local host.

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 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.

logging

Enable or disable syslog and SNMP logging. (Configuration mode.)

logging on

no logging on

logging buffered *level*

no logging buffered

logging console *level*

no logging console

logging facility *facility*

no logging facility *facility*

logging history *level*

no logging history *level*

logging host [*in_if_name*] *ip_address* [*protocol/port*]

no logging host [*in_if_name*] *ip_address*

logging message *syslog_id*

no logging message *syslog_id*

clear logging disabled

show logging disabled

logging monitor *level*

no logging monitor *level*

logging queue *queue_size*

show logging queue

logging standby

no logging standby

logging timestamp

no logging timestamp

logging trap *level*

no logging trap *level*

show logging

clear logging

Syntax	Description
on	Start sending syslog messages to all output locations. Stop all logging with the no logging on command.
buffered	Send syslog messages to an internal buffer that can be viewed with the show logging command. Use the clear logging command to clear the message buffer. New messages append to the end of the buffer.
<i>level</i>	Specify the syslog message level as a number or string. The <i>level</i> you specify means that you want that <i>level</i> and those less than the <i>level</i> . For example, if <i>level</i> is 3 , syslog displays 0 , 1 , 2 , and 3 messages. Possible number and string <i>level</i> values are: <ul style="list-style-type: none"> • 0—emergencies—System unusable messages • 1—alerts—Take immediate action • 2—critical—Critical condition • 3—errors—Error message • 4—warnings—Warning message • 5—notifications—Normal but significant condition • 6—informational—Information message • 7—debugging—Debug messages and log FTP commands and WWW URLs
console	Specify that syslog messages appear on the PIX Firewall console as each message occurs. You can limit the types of messages that appear on the console with <i>level</i> . We recommend that you do not use this command in production mode because its use degrades PIX Firewall performance.

facility	Specify the syslog facility. The default is 20.
<i>facility</i>	Eight facilities LOCAL0(16) through LOCAL7(23); the default is LOCAL4(20). Hosts file the messages based on the <i>facility</i> number in the message.
history	Set the SNMP message level for sending syslog traps.
host	Specify a syslog server that will receive the messages sent from the PIX Firewall. You can use multiple logging host commands to specify additional servers that would all receive the syslog messages. However a server can only be specified to receive either UDP or TCP, not both. PIX Firewall only sends TCP syslog messages to the PIX Firewall Syslog Server.
<i>in_if_name</i>	Interface on which the syslog server resides.
<i>ip_address</i>	Syslog server's IP address.
<i>protocol</i>	The protocol over which the syslog message is sent; either tcp or udp . PIX Firewall only sends TCP syslog messages to the PIX Firewall Syslog Server. You can only view the port and protocol values you previously entered by using the write terminal command and finding the command in the listing—the TCP protocol is listed as 6 and the UDP protocol is listed as 17.
<i>port</i>	The port from which the PIX Firewall sends either UDP or TCP syslog messages. This must be same port at which the syslog server listens. For the UDP port, the default is 514 and the allowable range for changing the value is 1025 through 65535. For the TCP port, the default is 1470, and the allowable range is 1025 through 65535. TCP ports only work with the PIX Firewall Syslog Server.
message	Specify a message to be allowed. Use the no logging message command to suppress a syslog message. Use the clear logging disabled command to reset the disallowed messages to the original set. Use the show message disabled command to list the suppressed messages. All syslog messages are permitted unless explicitly disallowed. The “PIX Startup begin” message cannot be blocked and neither can more than one message per command statement.
<i>syslog_id</i>	Specify a message number to disallow or allow. If a message is listed in syslog as %PIX-1-101001, use “101001” as the <i>syslog_id</i> . Refer to <i>Cisco PIX Firewall System Log Messages</i> for message numbers. You can view this document online at the following website: http://www.cisco.com/univercd/cc/td/doc/product/iaabu/pix/pix_v61/syslog/index.htm
disabled	Clear or display suppressed messages. You can suppress messages with the no logging message command.
monitor	Specify that syslog messages appear on Telnet sessions to the PIX Firewall console.
queue <i>queue_size</i>	Specifies the size of the queue for storing syslog messages. Use this parameter before the syslog messages are processed. The queue parameter defaults to 512 messages, 0 (zero) indicates unlimited (subject to available block memory), and the minimum is one message. Use the show logging queue command to determine the current number of messages in the queue, highest number recorded, and number of messages discarded because block memory was not available to process them.
standby	Let the failover standby unit also send syslog messages. This option is disabled by default. You can enable it to ensure that the standby unit's syslog messages stay synchronized should failover occur. However, this option causes twice as much traffic on the syslog server. Disable with the no logging standby command.

timestamp	Specify that syslog messages sent to the syslog server should have a time stamp value on each message.
trap	Set logging level only for syslog messages.
clear	Clear the buffer for use with the logging buffered command.
show	List which logging options are enabled. If the logging buffered command is in use, the show logging command lists the current message buffer.

Usage Guidelines

The **logging** command lets you enable or disable sending informational messages to the console, to a syslog server, or to an SNMP management station. Set the SNMP message level with the **logging history** command, and set the syslog message level with the **logging trap** command.

The **logging queue** command lets you specify the size of the syslog message queue for the messages waiting to be processed. When traffic is heavy, messages may be discarded.

The **show logging queue** command lists:

- Number of messages in the queue
- Highest number of messages recorded in the queue
- Number of messages discarded because block memory was not available to process them

The **logging standby** command lets the failover standby unit send syslog messages. This option is disabled by default. You can enable it to ensure that the standby unit's syslog messages stay synchronized should failover occur. However, this option causes twice as much traffic on the syslog server. Disable with the **no logging standby** command.

For more information on syslog and the use of the **logging** command, refer to *Cisco PIX Firewall System Log Messages* at the following website.

http://www.cisco.com/univercd/cc/td/doc/product/iaabu/pix/pix_60/syslog/index.htm

You can also use *Cisco PIX Firewall System Log Messages* to get the message numbers that can be individually suppressed with the **logging message** command. Use the **show logging disabled** command to view suppressed syslog messages.

Important Notes

1. Do not use the **logging console** command when the PIX Firewall is in production mode because it degrades system performance. By default, this command is disabled. Instead, use the **logging buffered** command to start logging, the **show logging** command to view the messages, and the **clear logging** command to clear the buffer to make viewing the most current messages easier.
2. PIX Firewall provides more information in messages sent to a syslog server than at the console, but the console provides enough information to permit effective troubleshooting.
3. The **logging timestamp** command requires that the **clock** command be set.
4. The **no logging message** command cannot block the "%PIX-6-199002: PIX startup completed. Beginning operation." syslog message.
5. The **aaa authentication enable console** command causes syslog messages to be sent (at syslog level 4) each time the configuration is changed from the serial console.

Examples

The following example shows how to start console logging and view the results:

```
logging buffered debugging
show logging
Syslog logging: enabled
  Timestamp logging: disabled
  Console logging: disabled
  Monitor logging: disabled
  Buffer logging: level debugging, 37 messages logged
  Trap logging: disabled
305001: Portmapped translation built for gaddr 209.165.201.5/0 laddr 192.168.1.2/256
...
```

The line of output starting with 305001 shows a translation to a PAT global through global address 209.165.201.5 from a host at 192.168.1.2. The “305001” identifies a syslog message for creating a translation through a PAT global. Refer to *Cisco PIX Firewall System Log Messages* for more information on syslog messages.

The next example lists the output of the **logging queue** and **show logging queue** commands:

```
logging queue 0
show logging queue
Logging Queue length limit : Unlimited
Current 5 msg on queue, 3513 msgs most on queue, 1 msg discard.
```

In this example, the **logging queue** command is set to 0, which means you want an unlimited number of messages; in other words, all syslog messages, to be processed. The **show logging queue** command shows that 5 messages are queued, 3513 messages was the greatest number of messages in the queue at one time since the PIX Firewall was last booted, and that 1 message was discarded. Even though set for unlimited, should the amount of block memory be exhausted, messages can still be discarded.

Related Commands

- [clear Commands](#)
- [telnet](#)
- [terminal](#)

