



## Accessing and Monitoring PIX Firewall

---

This chapter describes how to configure and use the tools and features provided by the PIX Firewall for monitoring and configuring the system, and for monitoring network activity. It contains the following sections:

- [Command Authorization and LOCAL User Authentication](#)
- [Using Network Time Protocol](#)
- [Managing the PIX Firewall Clock](#)
- [Using Telnet for Remote System Management](#)
- [Using SSH for Remote System Management](#)
- [Enabling Auto Update Support](#)
- [Capturing Packets](#)
- [IDS Syslog Messages](#)
- [Using SNMP](#)

### Command Authorization and LOCAL User Authentication

This section describes the Command Authorization feature and related topics, introduced with PIX Firewall version 6.2. It includes the following topics:

- [Privilege Levels](#)
- [User Authentication](#)
- [Command Authorization](#)
- [Recovering from Lockout](#)

#### Privilege Levels

PIX Firewall version 6.2 introduces support for up to 16 privilege levels. This is similar to what is available with Cisco IOS software. With this feature, you can assign PIX Firewall commands to one of 16 levels. Also, users logging into the PIX Firewall are assigned privilege levels.

**Note**

Users with a privilege level greater than or equal to 2 have access to the enable and configuration mode and therefore the PIX Firewall prompt changes to #. Users with a privilege level 0 or 1 see the prompt >.

When a user tries to access enable mode, if the message “T+ enable privilege too low” appears on the AAA server, set the Max privilege of the AAA client to Level1 in the Advanced TACACS options.

To enable different privilege levels on the PIX Firewall, use the **enable** command in configuration mode. To assign a password to a privilege level, enter the following command:

```
pix(config)# enable password [password] [level level] [encrypted]
```

Replace *password* with a character string from three to sixteen characters long, with no spaces. Replace *level* with the privilege level you want to assign to the enable password.

**Note**

The **encrypted** keyword indicates to the PIX Firewall that the password supplied with the **enable** command is already encrypted.

For example, the following command assigns the enable password Passw0rD to privilege Level 10:

```
enable password Passw0rD level 10
```

The following example shows the usage of the **enable password** command with the **encrypted** keyword:

```
enable password .SUTWWLlTTApDYx level 9 encrypted
```

**Note**

Encrypted passwords that are associated with a level can only be moved among PIX Firewall units along with the associated levels.

Once the different privilege levels are created, you can gain access to a particular privilege level from the > prompt by entering the enable command, as shown below:

```
pix> enable [privilege level]
```

Replace *privilege level* with the privilege level to which you want to gain access. If the privilege level is not specified, the default of 15 is used. By default, privilege level 15 is assigned the password **cisco**. It will always have a password associated with it unless someone assigns it a blank password using the **enable password** command.

## User Authentication

This section describes how to configure the PIX Firewall to use LOCAL user authentication. It includes the following topics:

- [Creating User Accounts in the LOCAL Database](#)
- [User Authentication Using the LOCAL Database](#)
- [Viewing the Current User Account](#)

## Creating User Accounts in the LOCAL Database

To define a user account in the LOCAL database, enter the following command:

```
username username {nopassword|password password [encrypted]} [privilege level]
```

Replace *username* with a character string from four to fifteen characters long. Replace *password* with a character string from three to sixteen characters long. Replace *privilege level* with the privilege level you want to assign to the new user account (from 0 to 15). Use the **nopassword** keyword to create a user account with no password. Use the **encrypted** keyword if the password you are supplying is already encrypted.

**Note**

The username database that you configure can be moved among PIX Firewall units with the rest of the configuration. Encrypted passwords can only be moved along with the associated username in the database.

For example, the following command assigns a privilege level of 15 to the user account *admin*.

```
username admin password passw0rd privilege 15
```

If no privilege level is specified, the user account is created with a privilege level of 2. You can define as many user accounts as you need.

Use the following command to create a user account with no password:

```
username username nopassword
```

Replace *username* with the user account that you want to create without a password.

To delete an existing user account, enter the following command:

```
no username username
```

Replace *username* with the user account that you want to delete. For example, the following command deletes the user account **admin**.

```
no username admin
```

To remove all the entries from the user database, enter the following command:

```
clear username
```

## User Authentication Using the LOCAL Database

User authentication can be completed using the LOCAL database after user accounts are created in this database.

**Note**

The LOCAL database can be used only for controlling access *to* the PIX Firewall, and not for controlling access *through* the PIX Firewall.

To enable authentication using the LOCAL database, enter the following command:

```
pix(config)# aaa authentication serial|telnet|ssh|http|enable console LOCAL
```

After entering this command, the LOCAL user accounts are used for authentication.

You can also use the **login** command, as follows, to access the PIX Firewall with a particular username and password:

```
pix> login
```

The **login** command only checks the local database while authenticating a user and does not check any authentication or authorization (AAA) server.

When you enter the **login** command, the system prompts for a username and password as follows:

```
Username:admin
Password:*****
```



#### Note

Users with a privilege level greater than or equal to 2 have access to the enable and configuration modes and the PIX Firewall prompt changes to #. Users with the privilege level 0 or 1 see the prompt >.

Use the following command to log out from the currently logged in user account:

```
logout
```

## Viewing the Current User Account

The PIX Firewall maintains usernames in the following authentication mechanisms:

- LOCAL
- TACACS+
- RADIUS

To view the user account that is currently logged in, enter the following command:

```
show curpriv
```

The system displays the current user name and privilege level, as follows:

```
Username:admin
Current privilege level: 15
Current Mode/s:P_PRIV
```

As mentioned in the section “[Privilege Levels](#),” you use the **enable** command to obtain access to different privilege levels with the following command:

```
pix> enable [privilege level]
```

When you assign a password to a privilege level, the privilege level is associated with the password in the LOCAL database in the same way a username is associated with a password. When you obtain access to a privilege level using the **enable** command, the **show curpriv** command displays the current privilege level as a username in the format **enable\_n**, where *n* is a privilege level from 1 to 15.

An example follows:

```
pix# show curpriv
Username : enable_9
Current privilege level : 9
Current Mode/s : P_PRIV
```

When you enter the **enable** command without specifying the privilege level, the default privilege level (15) is assumed and the username is set to **enable\_15**.

When you log into the PIX Firewall for the first time or exit from the current session, the default user name is **enable\_1**, as follows:

```
pix> show curpriv
Username : enable_1
Current privilege level : 1
Current Mode/s : P_UNPR
```

## Command Authorization

This section describes how to assign commands to different privilege levels. It includes the following topics:

- [Overview](#)
- [Configuring LOCAL Command Authorization](#)
- [Enabling LOCAL Command Authorization](#)
- [Viewing LOCAL Command Authorization Settings](#)
- [TACACS+ Command Authorization](#)

### Overview

LOCAL and TACACS+ Command Authorization is supported in PIX Firewall version 6.2. With the LOCAL command authorization feature, you can assign PIX Firewall commands to one of 16 levels.



#### Caution

When configuring the Command Authorization feature, *do not* save your configuration until you are sure it works the way you want. If you get locked out because of a mistake, you can usually recover access by simply restarting the PIX Firewall from the configuration that is saved in Flash memory. If you still get locked out, refer to the section “[Recovering from Lockout](#).”

## Configuring LOCAL Command Authorization

In the default configuration, each PIX Firewall command is assigned to either privilege level 0 or privilege level 15. To reassign a specific command to a different privilege level, enter the following command:

```
[no] privilege [{show | clear | configure}] level level [mode {enable|configure}] command
command
```

Replace *level* with the privilege level and *command* with the command you want to assign to the specified level. You can use the **show**, **clear**, or **configure** parameter to optionally set the privilege level for the **show**, **clear**, or **configure** command modifiers of the specified command. Replace *command* with the command for which you wish to assign privileges. For the full syntax of this command, including additional options, refer to the *PIX Firewall Command Reference Guide*.

For example, the following commands set the privilege of the different command modifiers of the **access-list** command:

```
privilege show level 10 command access-list
privilege configure level 12 command access-list
privilege clear level 11 command access-list
```

The first line sets the privilege of **show access-list** (**show** modifier of **cmd access-list**) to **10**. The second line sets the privilege level of the **configure** modifier to 12, and the last line sets the privilege level of the **clear** modifier to 11.

To set the privilege of all the modifiers of the **access-list** command to a single privilege level of 10, you would enter the following command:

```
privilege level 10 command access-list
```

For commands that are available in multiple modes, use the **mode** parameter to specify the mode in which the privilege level applies.

The following are examples of setting privilege levels for mode-specific commands:

```
privilege show level 15 mode configure command configure
privilege clear level 15 mode configure command configure
privilege configure level 15 mode configure command configure
privilege configure level 15 mode enable command configure
```

```
privilege configure level 0 mode enable command enable
privilege show level 15 mode configure command enable
privilege configure level 15 mode configure command enable
```

```
privilege configure level 15 mode configure command igmp
privilege show level 15 mode configure command igmp
privilege clear level 15 mode configure command igmp
```

```
privilege show level 15 mode configure command logging
privilege clear level 15 mode configure command logging
privilege configure level 15 mode configure command logging
privilege clear level 15 mode enable command logging
privilege configure level 15 mode enable command logging
```



#### Note

---

Do not use the **mode** parameter for commands that are not mode-specific.

---

By default, the following commands are assigned to privilege level 0:

```
privilege show level 0 command checksum
privilege show level 0 command curpriv
privilege configure level 0 command help
privilege show level 0 command history
privilege configure level 0 command login
privilege configure level 0 command logout
privilege show level 0 command pager
privilege clear level 0 command pager
privilege configure level 0 command pager
privilege configure level 0 command quit
privilege show level 0 command version
```

## Enabling LOCAL Command Authorization

Once you have reassigned privileges to commands from the defaults, as necessary, enable the command authorization feature by entering the following command:

```
aaa authorization command LOCAL
```

By specifying LOCAL, the user's privilege level and the privilege settings that have been assigned to the different commands are used to make authorization decisions.

When users log in to the PIX Firewall, they can enter any command assigned to their privilege level or to lower privilege levels. For example, a user account with a privilege level of 15 can access every command because this is the highest privilege level. A user account with a privilege level of 0 can only access the commands assigned to level 0.

## Viewing LOCAL Command Authorization Settings

To view the CLI command assignments for each privilege level, enter the following command:

```
show privilege all
```

The system displays the current assignment of each CLI command to a privilege level. The following example illustrates the first part of the display:

```
pix(config)# show privilege all
privilege show level 15 command aaa
privilege clear level 15 command aaa
privilege configure level 15 command aaa
privilege show level 15 command aaa-server
privilege clear level 15 command aaa-server
privilege configure level 15 command aaa-server
privilege show level 15 command access-group
privilege clear level 15 command access-group
privilege configure level 15 command access-group
privilege show level 15 command access-list
privilege clear level 15 command access-list
privilege configure level 15 command access-list
privilege show level 15 command activation-key
privilege configure level 15 command activation-key
```

To view the command assignments for a specific privilege level, enter the following command:

```
show privilege level level
```

Replace *level* with the privilege level for which you want to display the command assignments.

For example, the following command displays the command assignments for privilege Level 15:

```
show privilege level 15
```

To view the privilege level assignment of a specific command, enter the following command:

```
show privilege command command
```

Replace *command* with the command for which you want to display the assigned privilege level.

For example, the following command displays the command assignment for the **access-list** command:

```
show privilege command access-list
```

## TACACS+ Command Authorization



### Caution

Only enable this feature with TACACS+ if you are absolutely sure that you have fulfilled the following requirements.

1. You have created entries for **enable\_1**, **enable\_15**, and any other levels to which you have assigned commands.
2. If you are enabling authentication with usernames:

- You have a user profile on the TACACS+ server with all the commands that the user is permitted to execute.
  - You have tested authentication with the TACACS+ server.
3. You are logged in as a user with the necessary privileges. You can see this by entering the **show curpriv** command.
  4. Your TACACS+ system is completely stable and reliable. The necessary level of reliability typically requires that you have a fully redundant TACACS+ server system and fully redundant connectivity to the PIX Firewall.

**Caution**

When configuring the Command Authorization feature, *do not* save your configuration until you are sure it works the way you want. If you get locked out because of a mistake, you can usually recover access by simply restarting the PIX Firewall from the configuration that is saved in Flash memory. If you still get locked out, refer to the section “[Recovering from Lockout](#).”

After command authorization with a TACACS+ server is enabled, for each command entered, the PIX Firewall sends the username, command, and command arguments to the TACACS+ server for authorization.

To enable command authorization with a TACACS+ server, enter the following command:

```
aaa authorization command tacacs_server_tag
```

To create the *tacacs\_server\_tag*, use the **aaa-server** command, as follows:

```
aaa-server tacacs_server_tag [(if_name)] host ip_address [key] [timeout seconds]
```

Use the *tacacs\_server\_tag* parameter to identify the TACACS+ server and use the *if\_name* parameter if you need to specifically identify the PIX Firewall interface connected to the TACACS+ server. Replace *ip\_address* with the IP address of the TACACS+ server. Replace the optional *key* parameter with a keyword of up to 127 characters (including special characters but excluding spaces) to use for encrypting data exchanged with the TACACS+ server. This value must match the keyword used on the TACACS+ server. Replace *seconds* with a number up to 30 that determines how long the PIX Firewall waits before retrying the connection to the TACACS+ server. The default value is 5 seconds.

The PIX Firewall only expands the command and the command modifier (**show**, **clear**, **no**) when it sends these to the TACACS+ server. The command arguments are *not* expanded.

For effective operation, it is a good idea to permit the following basic commands on the AAA server:

- **show curpriv**
- **show version**
- **show aaa**
- **enable**
- **disable**
- **quit**
- **exit**
- **login**
- **logout**
- **help**

For Cisco PIX Device Manager (PDM) to work with Command Authorization using a TACACS+ Server, the AAA server administrator should authorize the user for the following commands:

- **write terminal** or **show running-config**
- **show pdm**
- **show version**
- **show curpriv**

## Recovering from Lockout

If you get locked out because of a mistake in configuring Command Authorization, you can usually recover access by simply restarting the PIX Firewall from the configuration that is saved in Flash memory.

If you have already saved your configuration and you find that you configured authentication using the LOCAL database but did not configure any usernames you created a lockout problem. You can also encounter a lockout problem by configuring command authorization using a TACACS+ server if the TACACS+ server is unavailable, down or misconfigured.

If you cannot recover access to the PIX Firewall by restarting your PIX Firewall, use your web browser to access the following website:

<http://www.cisco.com/warp/customer/110/34.shtml>

This website provides a downloadable file with instructions for using it to remove the lines in the PIX Firewall configuration that enable authentication and cause the lockout problem.

You can encounter a different type of lockout problem if you use the **aaa authorization command** *tacacs\_server\_tag* command and you are not logged as the correct user. For every command you type, the PIX Firewall will display the following message:

```
Command Authorization failed
```

This occurs because the TACACS+ server does not have a user profile for the user account that you used for logging in. To prevent this problem, make sure that the TACACS+ server has all the users configured with the commands that they can execute. Also make sure that you are logged in as a user with the required profile on the TACACS+ server.

## Using Network Time Protocol

This section describes how to use the Network Time Protocol (NTP) client, introduced with PIX Firewall version 6.2. It includes the following topics:

- [Overview](#)
- [Enabling NTP](#)
- [Viewing NTP Status and Configuration](#)

### Overview

The Network Time Protocol (NTP) is used to implement a hierarchical system of servers that provide a source for precisely synchronized time among network systems. This kind of accuracy is required for time-sensitive operations such as validating a certificate revocation lists (CRL), which includes a precise time stamp.

PIX Firewall version 6.2 introduces an NTP client that allows the PIX Firewall to obtain its system time from NTP version 3 servers, like those provided with Cisco IOS routers.

## Enabling NTP

To enable the PIX Firewall NTP client, enter the following command:

```
[no] ntp server ip_address [key number] source if_name [prefer]
```

This command causes the PIX Firewall to synchronize with the time server identified by *ip\_address*. The **key** option requires a authentication key when sending packets to this server. When using this option, replace *number* with the authentication key. The interface specified by *if\_name* is used to send packets to the time server. If the **source** keyword is not specified, the routing table will be used to determine the interface. The **prefer** option makes the specified server the preferred server to provide synchronization, which reduces switching back and forth between servers.

To enable authentication for NTP messages, enter the following command:

```
[no] ntp authenticate
[no] ntp authentication-key number md5 value
[no] ntp trusted-key number
```

The **ntp authenticate** command enables NTP authentication. If you enter this command, the PIX Firewall will not synchronize to an NTP server unless the server is configured with one of the authentication keys specified using the **ntp trusted-key** command.

The **ntp authentication-key** command is used to define authentication keys for use with other NTP commands to provide a higher degree of security. The *number* parameter is the key number (1 to 4294967295). The *value* parameter is the key value (an arbitrary string of up to 32 characters). The key value will be replaced with ‘\*\*\*\*\*’ when the configuration is viewed with either the **write terminal**, **show configuration**, or **show tech-support** commands.

Use the **ntp trusted-key** command to define one or more key numbers corresponding to the keys defined with the **ntp authentication-key** command. The PIX Firewall will require the NTP server to provide this key number in its NTP packets. This provides protection against synchronizing the PIX Firewall system clock with an NTP server that is not trusted.

To remove NTP configuration, enter the following command:

```
clear ntp
```

This command removes the NTP configuration, disables authentication, and removes all the authentication keys.

## Viewing NTP Status and Configuration

This section describes the information available about NTP status and associations. To view information about NTP status and configuration, use any of the following commands:

- **show ntp associations**—displays information about the configured time servers.
- **show ntp associations detail**—provides detailed information.
- **show ntp status**—displays information about the NTP clock.

The following examples show sample output for each command and the following tables define the meaning of the values in each column of the output.

Example 9-1 shows sample output from the **show ntp associations** command:

**Example 9-1 Sample Output for show ntp association Command**

```
PIX> show ntp associations
      address      ref clock      st when poll reach delay offset disp
~172.31.32.2      172.31.32.1    5  29 1024 377   4.2  -8.59  1.6
+~192.168.13.33  192.168.1.111  3  69  128 377   4.1   3.48  2.3
*~192.168.13.57  192.168.1.111  3  32  128 377   7.9  11.18  3.6
* master (synced), # master (unsynced), + selected, - candidate, ~ configured
```

The first characters in a display line can be one or more of the following characters:

- \*—Synchronized to this peer
- #—Almost synchronized to this peer
- +—Peer selected for possible synchronization
- -—Peer is a candidate for selection
- ~—Peer is statically configured
- Table 9-1 describes the meaning of the values in each column:

**Table 9-1 Output Description for show ntp association Command**

Output Column Heading	Description
address	Address of peer.
ref clock	Address of reference clock of peer.
st	Stratum of peer.
when	Time since last NTP packet was received from peer.
poll	Polling interval (in seconds).
reach	Peer reachability (bit string, in octal).
delay	Round-trip delay to peer (in milliseconds).
offset	Relative time of peer clock to local clock (in milliseconds).
disp	Dispersion.

Example 9-2 provides sample output for the **show ntp association detail** command:

**Example 9-2 Sample Output for show ntp association detail Command**

```
pix(config)# show ntp associations detail
172.23.56.249 configured, our_master, sane, valid, stratum 4
ref ID 172.23.56.225, time c0212639.2ecfc9e0 (20:19:05.182 UTC Fri Feb 22
2002)
our mode client, peer mode server, our poll intvl 128, peer poll intvl 128
root delay 38.04 msec, root disp 9.55, reach 177, sync dist 156.021
delay 4.47 msec, offset -0.2403 msec, dispersion 125.21
precision 2**19, version 3
org time c02128a9.731f127b (20:29:29.449 UTC Fri Feb 22 2002)
rcv time c02128a9.73c1954b (20:29:29.452 UTC Fri Feb 22 2002)
xmt time c02128a9.6b3f729e (20:29:29.418 UTC Fri Feb 22 2002)
filtdelay =      4.47   4.58   4.97   5.63   4.79   5.52   5.87
0.00
```

```

filtoffset =  -0.24  -0.36  -0.37   0.30  -0.17   0.57  -0.74
0.00
filterror =   0.02   0.99   1.71   2.69   3.66   4.64   5.62
16000.0

```

Table 9-2 describes the meaning of the values in each column:

**Table 9-2** Output Description for show ntp association detail Command

Output Column Heading	Description
configured	Peer was statically configured.
dynamic	Peer was dynamically discovered.
our_master	Local machine is synchronized to this peer.
selected	Peer is selected for possible synchronization.
candidate	Peer is a candidate for selection.
sane	Peer passes basic sanity checks.
insane	Peer fails basic sanity checks.
valid	Peer time is believed to be valid.
invalid	Peer time is believed to be invalid.
leap_add	Peer is signalling that a leap second will be added.
leap-sub	Peer is signalling that a leap second will be subtracted.
unsynced	Peer is not synchronized to any other machine.
ref ID	Address of machine peer is synchronized to.
time	Last time stamp peer received from its master.
our mode	Our mode relative to peer (active/passive/client/server/bdcast/bdcast client).
peer mode	Peer's mode relative to us.
our poll intvl	Our poll interval to peer.
peer poll intvl	Peer's poll interval to us.
root delay	Delay along path to root (ultimate stratum 1 time source).
root disp	Dispersion of path to root.
reach	Peer reachability (bit string in octal).
sync dist	Peer synchronization distance.
delay	Round-trip delay to peer.
offset	Offset of peer clock relative to our clock.
dispersion	Dispersion of peer clock.
precision	Precision of peer clock in hertz.
version	NTP version number that peer is using.
org time	Originate time stamp.
rcv time	Receive time stamp.
xmt time	Transmit time stamp.
filtdelay	Round-trip delay (in milliseconds) of each sample.

**Table 9-2** Output Description for show ntp association detail Command (continued)

Output Column Heading	Description
filtoffset	Clock offset (in milliseconds) of each sample.
filterror	Approximate error of each sample.

[Example 9-3](#) provides sample output for the **show ntp status** command:

**Example 9-3** Output of the show ntp status Command

```

pixfirewall(config)# show ntp status
Clock is synchronized, stratum 5, reference is 172.23.56.249
nominal freq is 99.9984 Hz, actual freq is 100.0266 Hz, precision is 2**6
reference time is c02128a9.73c1954b (20:29:29.452 UTC Fri Feb 22 2002)
clock offset is -0.2403 msec, root delay is 42.51 msec
root dispersion is 135.01 msec, peer dispersion is 125.21 msec

```

[Table 9-3](#) describes the meaning of the values in each column:

**Table 9-3** Output Description for show ntp status Command

Output Column Heading	Description
synchronized	System is synchronized to an NTP peer.
unsynchronized	System is not synchronized to any NTP peer.
stratum	NTP stratum of this system.
reference	Address of peer to which the system is synchronized.
nominal freq	Nominal frequency of system hardware clock.
actual freq	Measured frequency of system hardware clock.
precision	Precision of the clock of this system (in hertz).
reference time	Reference time stamp.
clock offset	Offset of the system clock to synchronized peer.
root delay	Total delay along path to root clock.
root dispersion	Dispersion of root path.
peer dispersion	Dispersion of synchronized peer.

## Managing the PIX Firewall Clock

This section describes how to manage the PIX Firewall system clock and includes the following topics:

- [Viewing System Time](#)
- [Setting the System Clock](#)
- [Setting Daylight Savings Time and Timezones](#)

## Viewing System Time

To view the current system time, enter the following command:

```
show clock [detail]
```

This command displays the system time. The **detail** option displays the clock source and the current summer-time setting. PIX Firewall version 6.2 provides milliseconds, timezone, and day.

For example:

```
16:52:47.823 PST Wed Feb 21 2001
```

## Setting the System Clock

To set the system time, enter the following command:

```
clock set hh:mm:ss month day year
```

Replace *hh:mm:ss* with the current hours (1-24), minutes, and seconds. Replace *month* with the first three characters of the current month. Replace *day* with the numeric date within the month (1-31), and replace *year* with the four-digit year (permitted range is 1993 to 2035).

## Setting Daylight Savings Time and Timezones

PIX Firewall version 6.2 also provides enhancements to the **clock** command to support daylight savings (summer) time and time zones.

To configure daylight savings (summer) time, enter the following command:

```
clock summer-time zone recurring [week weekday month hh:mm week weekday month hh:mm [offset]]
```

The **summer-time** keyword automatically switches to summer time (for display purposes only).

The **recurring** keyword indicates that summer time should start and end on the days specified by the values that follow this keyword. If no values are specified, the summer time rules default to United States rules. The *week* option is the week of the month (1 to 5 or **last**). The *weekday* option is the day of the week (Sunday, Monday,...). The *month* parameter is the full name of the month (January, February,...). The *hh:mm* parameter is the time (24-hour military format) in hours and minutes. The *offset* option is the number of minutes to add during summer time (default is 60).

Use either of the following commands when the **recurring** keyword cannot be used:

```
clock summer-time zone date date month year hh:mm date month year hh:mm [offset]
clock summer-time zone date month date year hh:mm month date year hh:mm [offset]
```

The **date** keyword causes summer time to start on the first date listed in the command and to end on the second specific date in the command. Two forms of the command are included to enter dates either in the form *month date* (for example, January 31) or *date month* (for example, 31 January).

In both forms of the command, the first part of the command specifies when summer time begins, and the second part specifies when it ends. All times are relative to the local time zone.

If the starting month is after the ending month, the Southern Hemisphere is assumed.

The *zone* parameter is the name of the time zone (for example, PDT) to be displayed when summer time is in effect. The *week* option is the week of the month (1 to 5 or **last**). The *weekday* option is the day of the week (Sunday, Monday,...). The *date* parameter is the date of the month (1 to 31). The *month*

parameter is the full name of the month (January, February,...). The *year* parameter is the four-digit year (1993 to 2035). The *hh:mm* parameter is the time (24-hour military format) in hours and minutes. The *offset* option is the number of minutes to add during summer time (default is 60).

To set the time zone for display purposes only, enter the following command:

```
clock timezone zone hours [minutes]
```

The **clock timezone** command sets the time zone for display purposes (internally, the time is kept in UTC). The **no** form of the command is used to set the time zone to Coordinated Universal Time (UTC). The *zone* parameter is the name of the time zone to be displayed when standard time is in effect. The *hours* parameter is the hours offset from UTC. The *minutes* option is the minutes offset from UTC.

The **clear clock** command will remove the summer time setting and set the time zone to UTC.

## Using Telnet for Remote System Management

The serial console lets a single user configure the PIX Firewall, but often this is not convenient for a site with more than one administrator. PIX Firewall lets you access the console via Telnet from hosts on any internal interface. With IPsec configured, you can use Telnet to remotely administer the console of a PIX Firewall from lower security interfaces.



### Note

SSH provides another option for remote management of the PIX Firewall using a lower security interface. For further information, refer to “[Using SSH for Remote System Management](#).”

This section includes the following topics:

- [Configuring Telnet Console Access to the Inside Interface](#)
- [Allowing a Telnet Connection to the Outside Interface](#)
- [Using Telnet](#)
- [Trace Channel Feature](#)

## Configuring Telnet Console Access to the Inside Interface



### Note

See the **telnet** command page within the *Cisco PIX Firewall Command Reference* for more information about this command.

Follow these steps to configure Telnet console access:

**Step 1** Enter the PIX Firewall **telnet** command.

For example, to let a host on the internal interface with an address of 192.168.1.2 access the PIX Firewall, enter the following:

```
telnet 192.168.1.2 255.255.255.255 inside
```

To Telnet to a lower security interface, refer to “[Allowing a Telnet Connection to the Outside Interface](#).”

**Step 2** If required, set the duration for how long a Telnet session can be idle before PIX Firewall disconnects the session.

The default duration, 5 minutes, is too short in most cases and should be increased until all pre-production testing and troubleshooting has been completed. Set a longer idle time duration as shown in the following example.

```
telnet timeout 15
```

**Step 3** To protect access to the console with an authentication server, use the **aaa authentication telnet console** command.

This requires that you have a username and password on the authentication server. When you access the console, PIX Firewall prompts you for these login credentials. If the authentication server is off line, you can still access the console by using the username **pix** and the password set with the **enable password** command.

**Step 4** Save the commands in the configuration using the **write memory** command.

[Example 9-4](#) shows commands for using Telnet to permit host access to the PIX Firewall console.

#### **Example 9-4 Using Telnet**

```
telnet 10.1.1.11 255.255.255.255
telnet 192.168.3.0 255.255.255.0
```

The first **telnet** command permits a single host, 10.1.1.11 to access the PIX Firewall console with Telnet. The 255 value in the last octet of the netmask means that only the specified host can access the console.

The second **telnet** command permits PIX Firewall console access from all hosts on the 192.168.3.0 network. The 0 value in the last octet of the netmask permits all hosts in that network access. However, Telnet only permits 16 hosts simultaneous access to the PIX Firewall console over Telnet.

## Allowing a Telnet Connection to the Outside Interface

This section tells you how to configure a Telnet connection to a lower security interface of the PIX Firewall. It includes the following topics:

- [Overview](#)
- [Using Cisco Secure VPN Client](#)
- [Using Cisco VPN 3000 Client](#)

### Overview

This section also applies when using the Cisco Secure Policy Manager version 2.0 or higher. It is assumed you are using the Cisco VPN Client version 3.x, Cisco Secure VPN Client version 1.1, or the Cisco VPN 3000 Client version 2.5/2.6, to initiate the Telnet connection.



#### **Note**

Use the **auth-prompt** command for changing the login prompt for Telnet sessions *through* the PIX Firewall. It does not change the login prompt for Telnet sessions to the PIX Firewall.

Once you have configured Telnet access, refer to “[Using Telnet](#)” for more information about using this command.

**Note**

You must have two security policies set up on your VPN client. One security policy is used to secure your Telnet connection and another is used to secure your connection to the inside network.

## Using Cisco Secure VPN Client

This section applies only if you are using a Cisco Secure VPN Client. In the example, the IP address of the PIX Firewall's outside interface is 168.20.1.5, and the Cisco Secure VPN Client's IP address, derived from the virtual pool of addresses, is 10.1.2.0.

To encrypt your Telnet connection to a PIX Firewall lower interface, perform the following steps as part of your PIX Firewall configuration:

- 
- Step 1** Create an **access-list** command statement to define the traffic to protect from the PIX Firewall to the VPN client using a destination address from the virtual local pool of addresses:
- ```
access-list 80 permit ip host 168.20.1.5 10.1.2.0 255.255.255.0
```
- Step 2** Specify which host can access the PIX Firewall console with Telnet:
- ```
telnet 10.1.2.0 255.255.255.0 outside
```
- Specify the VPN client's address from the local pool and the outside interface.
- Step 3** Within the VPN client, create a security policy that specifies the Remote Party Identity IP address and gateway IP address as the same IP address—the IP address of the PIX Firewall's outside interface. In this example, the IP address of the PIX Firewall's outside is 168.20.1.5.
- Step 4** Configure the rest of the security policy on the VPN client to match the PIX Firewall's security policy.
- 

## Using Cisco VPN 3000 Client

This section applies only if you are using a Cisco VPN 3000 Client. To encrypt your Telnet connection to the PIX Firewall's outside interface, perform the following step as part of your PIX Firewall configuration. In the following example, the IP address of the PIX Firewall's outside interface is 168.20.1.5, and the Cisco VPN 3000 Client's IP address stemming from the virtual pool of addresses is 10.1.2.0.

Specify which host can access the PIX Firewall console with Telnet. Specify the VPN client's address from the local pool and the outside interface.

```
telnet 10.1.2.0 255.255.255.0 outside
```

**Note**

To complete the configuration of the VPN client, refer to the **vpngroup** command in the *Cisco PIX Firewall Command Reference*.

## Using Telnet

Perform the following steps to test Telnet access:

- 
- Step 1** From the host, start a Telnet session to a PIX Firewall interface IP address.
- If you are using Windows 95 or Windows NT, click **Start>Run** to start a Telnet session. For example, if the inside interface IP address is 192.168.1.1, enter the following command.
- ```
telnet 192.168.1.1
```
- Step 2** The PIX Firewall prompts you with a password:
- ```
PIX passwd:
```
- Enter **cisco** and press the **Enter** key. You are then logged into the PIX Firewall.
- The default password is **cisco**, which you can change with the **passwd** command.
- You can enter any command on the Telnet console that you can set from the serial console, but if you reboot the PIX Firewall, you will must log back into the PIX Firewall after it restarts.
- Some Telnet applications such as the Windows 95 or Windows NT Telnet sessions may not support access to the PIX Firewall's command history feature used with the arrow keys. However, you can access the last entered commands by pressing Ctrl-P.
- Step 3** Once you have Telnet access available, you may want to view ping information while debugging.
- You can view ping information from Telnet sessions with the **debug icmp trace** command. The Trace Channel feature also affects **debug** displays, which is explained in "[Trace Channel Feature](#)."
- Messages from a successful ping appear as follows:
- ```
Outbound ICMP echo request (len 32 id 1 seq 512) 209.165.201.2 > 209.165.201.1
Inbound ICMP echo reply (len 32 id 1 seq 256) 209.165.201.1 > 209.165.201.23
```
- Step 4** In addition, you can use the Telnet console session to view syslog messages:
- Start message displays with the **logging monitor 7** command. The "7" will cause all syslog message levels to display.
- If you are using the PIX Firewall in production mode, you may wish to use the **logging buffered 7** command to store messages in a buffer that you can view with the **show logging** command, and clear the buffer for easier viewing with the **clear logging** command. To stop buffering messages, use the **no logging buffered** command.
- You can also lower the number from **7** to a lesser value, such as **3**, to limit the number of messages that appear.
- If you entered the **logging monitor** command, then enter the **terminal monitor** command to cause the messages to display in your Telnet session. To disable message displays, use the **terminal no monitor** command.
- 

## Trace Channel Feature

The **debug packet** command sends its output to the Trace Channel. All other **debug** commands do not. Use of Trace Channel changes the way you can view output on your screen during a PIX Firewall console or Telnet session.

If a **debug** command does not use Trace Channel, each session operates independently, which means any commands started in the session only appear in the session. By default, a session not using Trace Channel has output disabled by default.

The location of the Trace Channel depends on whether you have a simultaneous Telnet console session running at the same time as the console session, or if you are using only the PIX Firewall serial console:

- If you are only using the PIX Firewall serial console, all **debug** commands display on the serial console.
- If you have both a serial console session and a Telnet console session accessing the console, then no matter where you enter the **debug** commands, the output displays on the Telnet console session.
- If you have two or more Telnet console sessions, the first session is the Trace Channel. If that session closes, the serial console session becomes the Trace Channel. The next Telnet console session that accesses the console then becomes the Trace Channel.

The **debug** commands are shared between all Telnet and serial console sessions.

**Note**

The downside of the Trace Channel feature is that if one administrator is using the serial console and another administrator starts a Telnet console session, the output from the **debug** commands on the serial console will suddenly stop without warning. In addition, the administrator on the Telnet console session will suddenly be viewing **debug** command output, which may be unexpected. If you are using the serial console and **debug** command output is not appearing, use the **who** command to see if a Telnet console session is running.

## Using SSH for Remote System Management

This section describes how to use Secure Shell (SSH) for remote access to the PIX Firewall console. It includes the following topics:

- [Overview](#)
- [Obtaining an SSH Client](#)
- [Identifying the Host Using an SSH Client](#)
- [Configuring Authentication for an SSH Client](#)
- [Connecting to the PIX Firewall with an SSH Client](#)
- [Viewing SSH Status](#)

### Overview

SSH (Secure Shell) is an application running on top of a reliable transport layer, such as TCP/IP that provides strong authentication and encryption capabilities. PIX Firewall supports the SSH remote shell functionality provided in SSH version 1. SSH version 1 also works with Cisco IOS software devices. Up to five SSH clients are allowed simultaneous access to the PIX Firewall console.

**Note**

Before trying to use SSH, generate an RSA key-pair for the PIX Firewall. To use SSH, your PIX Firewall requires a DES or 3DES activation key.

Another method of remotely configuring a PIX Firewall unit involves using a Telnet connection to the PIX Firewall to start a shell session and then entering configuration mode. This connection method can only provide as much security as Telnet provides, which is only provided as lower-layer encryption (for example, IPSec) and application security (username/password authentication at the remote host).

**Note**

The PIX Firewall SSH implementation provides a secure remote shell session without IPSec, and only functions as a server, which means that the PIX Firewall cannot initiate SSH connections.

## Obtaining an SSH Client

**Note**

SSH v1.x and v2 are entirely different protocols and are not compatible. Make sure that you download a client that supports SSH v1.x.

Download an SSH v1.x client from one of the following websites.

- 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**

To use Tera Term Pro with SSH, download TTSSH. TTSSH provides a Zip file that you copy to your system. Extract the zipped files into the same folder that you installed Tera Term Pro.

- 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/>

## Identifying the Host Using an SSH Client

Identify each host to be used to access the PIX Firewall console using SSH by entering the following command:

```
[no] ssh ip_address [netmask] [interface_name]
```

To use this command:

- Replace *ip\_address* with the IP address of the host or network authorized to initiate an SSH connection to the PIX Firewall.
- Replace *netmask* with the network mask for *ip\_address*.



**Note** The **netmask** parameter is optional if you omit the interface name and if you use the default subnet mask (255.255.255.255). The **netmask** parameter is required if you specify the interface name or if you do not use the default subnet mask.

- Replace *interface\_name* with the PIX Firewall interface name on which the host or network initiating the SSH connection resides.

To specify the duration in minutes that a session can be idle before being disconnected, enter the following command:

```
ssh timeout number
```

Replace *number* with a value from 1 to 60 (minutes). The default duration is 5 minutes.

To disconnect a specific session, enter the following command:

```
ssh disconnect session_id
```

Replace *session\_id* with the identifier for the specific session that you want to disconnect. To display the identifiers for the active sessions, use the **show ssh sessions** command.

To remove all **ssh** command statements from the configuration, enter the following command:

```
clear ssh
```

Use the **no** keyword to remove selected **ssh** command statements from the configuration.



**Note** To use SSH, your PIX Firewall must have a DES or 3DES activation key and you must generate an RSA key-pair for the PIX Firewall before clients can connect to the PIX Firewall console. Use the **ca generate rsa key 512** command to generate a key; change the modulus size from 512, as needed. After generating the RSA key, save the key using the **ca save all** command.

## Configuring Authentication for an SSH Client

To configure local authentication for an SSH client accessing the PIX Firewall from a Linux or UNIX command line, enter the following command:

```
ssh -c 3des -l pix -v ipaddress
```

Use the **-c** option to identify the cipher used. PIX Firewall accepts **3des** and **des**. Use the **-l** option to identify the password used for connecting to the PIX Firewall. If no authentication is enabled on the SSH connection, use the default user name **pix**. Use the **-v** option to enable verbose mode, and replace *ipaddress* with the address of the PIX Firewall.



**Note** Windows and Macintosh SSH clients typically have graphic interfaces where you enter the required information.

The password used to perform local authentication is the same as the one used for Telnet access. The default for this password is **cisco**. To change this password, enter the following command:

```
passwd string
```

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

To enable authentication using a AAA server, enter the following command:

```
aaa authenticate ssh console server_tag
```

Replace *server\_tag* with the identifier for the AAA server.

## Connecting to the PIX Firewall with an SSH Client

To gain access to the PIX Firewall console using SSH, at the SSH client, enter the username **pix** and enter the Telnet password.

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

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

The display of the dot does not affect the functionality of SSH. The dot appears 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.

## Viewing SSH Status

To view the status of SSH sessions, enter the following command:

```
show ssh [sessions [ip_address]]
```

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.

## Enabling Auto Update Support

Auto Update is a protocol specification introduced with PIX Firewall version 6.2. This section describes how to enable support for this specification on a PIX Firewall and includes the following topics:

- [Overview](#)
- [Identifying the Auto Update Server](#)
- [Managing Auto Update Support](#)
- [Viewing the Auto Update Configuration](#)

## Overview

The Auto Update specification provides the infrastructure necessary for remote management applications to download PIX Firewall configurations, software images, and to perform basic monitoring from a centralized location.

The Auto Update specification allows the Auto Update Server to either push configuration information and send requests for information to the PIX Firewall, or to cause the PIX Firewall to periodically poll the Auto Update Server. The Auto Update Server can also send a command to the PIX Firewall to send an immediate polling request at any time. Communication between the Auto Update Server and the PIX Firewall requires a communications path and local CLI configuration on each PIX Firewall.

## Identifying the Auto Update Server

To specify the URL of the Auto Update Server, use the following command:

```
[no] auto-update server url [verify-certificate]
```

Only one server can be configured. Replace *url* with a URL using the following syntax:

```
[http[s]://][user:password@]location[:port]/pathname
```

SSL will be used when **https** is specified. The *user* and *password* segment is used for Basic Authentication when logging in to the server. The user and password are replaced with ‘\*\*\*\*\*’ when the configuration is viewed with either the **write terminal**, **show configuration** or **show tech-support** commands.

Replace *location* with the IP address (or a DNS host name that resolves to the IP address) of the server. The *port* segment specifies the port to contact on the server. The default is 80 for HTTP and 443 for HTTPS. The *pathname* segment is the name of the resource.

The **verify-certificate** option specifies that the certificate returned by the server should be verified.

The **no auto-update server** command disables polling for updates by terminating the Auto Update daemon running on the PIX Firewall.

## Managing Auto Update Support

To enable the PIX Firewall for polling a Auto Update Server, use the following command:

```
[no] auto-update device-id hardware-serial | hostname | ipaddress [if-name] | mac-address  
[if-name] | string text
```

The **auto-update device-id** command is used to identify the device ID to send when communicating with the Auto Update Server. The identifier used is determined by using one of the following parameters:

- **hardware-serial**—Use the PIX Firewall serial number
- **hostname** option—Use the PIX Firewall host name
- **ipaddress** option—Use the IP address of the interface with the name *if-name*. If the interface name is not specified, it will use the IP address of the interface used to communicate with the Auto Update Server.
- **mac-address** option—Use the MAC address of the interface with the name *if-name*. If the interface name is not specified, it will use the MAC address of the interface used to communicate with the Auto Update Server.

- **string**—Use the specified text identifier, which cannot contain white space or the characters ‘, “, , >, & and ?.

Use the **no auto-update device-id** command to reset the device ID to the default of host name.

To specify how often to poll the Auto Update server for configuration or image updates, enter the following command:

```
[no] auto-update poll-period poll-period [retry-count [retry-period]]
```

The *poll-period* parameter specifies how often (in minutes) to check for an update. The default is 720 minutes (12 hours). The *retry-count* option specifies how many times to try re-connecting to the server if the first attempt fails. The default is 0. The *retry-period* option specifies how long to wait (in minutes) between retries. The default is 5.

Use the **no auto-update poll-period** command to reset the poll period to the default.

To cause the PIX Firewall to stop all new connections if the Auto Update server has not been contacted for a period of time, enter the following command:

```
[no] auto-update timeout period
```

Use this command to ensure that the PIX Firewall has the most recent image and configuration. This condition will be reported with the existing syslog %PIX-3-201008.

To remove the entire Auto Update configuration, enter the following command:

```
clear auto-update
```

## Viewing the Auto Update Configuration

To display the Auto Update Server, poll time, timeout period, device ID, poll statistics and update statistics, enter the following command:

```
show auto-update
```

The following is sample output from the **show auto-update** command:

```
show auto-update
Server: https://\*\*\*\*\*@172.23.58.115:1742/management.cgi?1276
Certificate will be verified
Poll period: 720 minutes, retry count: 2, retry period: 5 minutes
Timeout: none
Device ID: host name [jeffryp-pix-pri]
Next poll in 4.93 minutes
Last poll: 11:36:46 PST Tue Nov 13 2001
Last PDM update: 23:36:46 PST Tue Nov 12 2001
```

## Capturing Packets

This section describes the packet capture utility introduced with PIX Firewall version 6.2. It includes the following topics:

- [Overview](#)
- [Configuration Procedure](#)
- [Packet Capture Output Formats](#)
- [Packet Capture Examples](#)

## Overview

Capturing packets is sometimes useful when troubleshooting connectivity problems or monitoring suspicious activity. You can use the PIX Firewall packet capture utility to capture specific types of traffic on any PIX Firewall interface.

The packet capture utility provides the following features:

- Capture of packets to a linear buffer
- Capture of ARP and other Layer 2 packets
- Timestamp of captured packets based from the PIX Firewall clock (in milliseconds)
- Selective packet capture and display based on access lists
- Display of captured buffer on any console or using a web browser
- Brief and expanded view of capture data
- Export of captured packets in libpcap format

## Configuration Procedure

To capture and display packets on a PIX Firewall interface, perform the following steps:

- Step 1** To define a packet capture and begin capturing packets on a specific interface, enter the following command:

```
capture capture-name [access-list acl_id] [buffer bytes] [ethernet-type type] [interface name] [packet-length bytes]
```

Replace *capture-name* with an alphanumeric identifier that you will use to display or copy the captured packets. The PIX Firewall captures packets on the interface specified by *name* until the packet capture buffer is full.

Replace *acl\_id* with the name of any existing access list, which can limit the capture based on one or more of the following selection criteria:

- IP protocol type
- Source or destination addresses
- TCP or UDP port
- ICMP type

For information about configuring an access control list, refer to “[Controlling Outbound Connectivity](#)” in [Chapter 3](#), “[Controlling Network Access and Use](#).”

To use the **buffer** option, replace *bytes* with the number of bytes you want to assign to the packet capture buffer, subject to the memory available on the PIX Firewall. The default buffer size is 512 K. You can run multiple packet captures on different interfaces concurrently if the PIX Firewall has sufficient memory.

To use the **ethernet** option, replace *type* with one of the following packet types: ip, arp, rarp, vlan, 802.1Q, ipx, ip6, pppoed, pppoes, or any number in the range from 1 to 65536 (corresponding to the protocol type specified in the Ethernet packet). When using 802.1Q (VLAN), the 802.1Q tag is automatically skipped and the inner ethernet-type is used for matching. If you enter **ethernet-type 0**, all packet types are captured.

To use the **packet-length** option, replace *bytes* with the maximum number of bytes from each packet that you want copied to the capture buffer. By default, the limit is 68 bytes.

**Step 2** To view the contents of the packet capture buffer, enter the following command:

```
show capture [capture-name] [access-list acl_id] [count count] [detail] [dump]
```

Replace *capture-name* with the identifier you assigned to the packet capture. Replace *acl\_id* with the name of an access control list to restrict the display of the captured packets. Replace *count* with the number of packets to display.

The fields included when you use the **detail** option are listed within square brackets ([]) in [Table 9-4](#).

The **dump** option displays a hexadecimal display of the packet transported over the data link transport. Note that Media Access Control (MAC) information is not shown. A dump is also displayed if no protocol is available.

Use the **show capture** command without any parameters to display the current runtime configuration for packet captures.

**Step 3** To view a packet capture using a web browser, enter the following command:

```
https://pix-host/capture/capture-name[/pcap]
```

Replace *pix-host* with the IP address or host name of the PIX Firewall where the packet capture occurred. Replace *capture-name* with the name of the packet capture you want to view.

The **pcap** option causes the packet capture to be downloaded to the web browser in libpcap format. After you save the packet capture from the browser, you can view a libpcap file with **tcpdump** or other applications.

**Step 4** To copy the contents of the packet capture buffer to a TFTP server, enter the following command:

```
copy capture:capture-name tftp://location/path [pcap]
```

Replace *capture-name* with the name of the packet capture you want to view. Replace *location* and *path* with the host name, path name, and file name of the file where you want to store the captured packets. Some TFTP servers may require that the file already exists with write permission assigned to “world.” The **pcap** option causes the file to be created in libpcap format, which can be viewed with **tcpdump** or other applications.

**Step 5** To clear the packet capture buffer, enter the following command:

```
clear capture capture-name
```

**Step 6** To clear the packet capture definition and release the resources allocated for it, enter the following command:

```
no capture capture-name
```

Replace *capture-name* with the name of the packet capture you want to clear.

**Step 7** To stop the packet capture and save the current contents of the packet capture buffer, enter the following command:

```
no capture capture-name [interface name]
```

Replace *capture-name* with the name of the packet capture you want to stop. When you use the **interface** option to identify a specific interface, replace *name* with the name assigned to the interface.

**Step 8** To remove the access list from a running packet capture, enter the following command:

```
no capture capture-name access-list acl_id
```

Replace *capture-name* with the name of the packet capture and replace *acl\_id* with the name of the access list.

## Packet Capture Output Formats

Table 9-4 shows the output formats for packet captures of different protocol types. The decoded output of the packets is dependent on the protocol of the packet. The output in square brackets is displayed when you use the **capture** command with the **detail** option.

**Table 9-4 Packet Capture Formats**

| Capture Type     | Syntax                                                                                                                                                               |
|------------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| ICMP packet      | HH:MM:SS.ms [ether-hdr] ip-source ip-destination: icmp: icmp-type icmp-code [checksum-failure]                                                                       |
| UDP packet       | HH:MM:SS.ms [ether-hdr] src-addr.src-port dest-addr.dst-port:[checksum-info] udp payload-len                                                                         |
| TCP packet       | HH:MM:SS.ms [ether-hdr] src-addr.src-port dest-addr.dst-port: tcp-flags [header-check] [checksum-info] sequence-number ack-number tcp-window urgent-info tcp-options |
| Other IP packets | HH:MM:SS.ms [ether-hdr] src-addr dest-addr: ip-protocol ip-length                                                                                                    |
| ARP packets      | HH:MM:SS.ms [ether-hdr] arp-type arp-info                                                                                                                            |
| Other packets    | HH:MM:SS.ms ether-hdr: hex-dump                                                                                                                                      |

## Packet Capture Examples

This section includes examples of different types of packet captures.

[Example 9-5](#) illustrates an HTTP packet capture.

### **Example 9-5 Capturing an HTTP Session**

In the following example, traffic is captured from an outside client at 209.165.200.225 to an inside HTTP server:

```
access-list http permit tcp host 10.120.56.15 eq http host 209.165.200.225
access-list http permit tcp host 209.165.200.225 host 10.120.56.15 eq http
capture capweb access-list http packet-length 74 interface inside
```

[Example 9-6](#) illustrates how to display a packet capture using a web browser.

### **Example 9-6 Displaying a libpcap File with a Web Browser**

The following command downloads a libpcap file to a local machine, using a web browser such as Internet Explorer or Netscape Communicator:

```
https://209.165.200.226/capture/http/pcap
```

[Example 9-7](#) copies an FTP trace to the file “ftp-dump” on the TFTP server 209.165.200.226.

**Example 9-7 Saving to a Remote TFTP Server**

```
pixfirewall# copy capture:ftp tftp://209.165.200.226/ftp-dump
Writing to file '/tftpboot/ftp-dump' at 209.165.200.226 on outside
```

[Example 9-8](#) illustrates a packet capture of ARP packets:

**Example 9-8 ARP Packet Capture**

```
+-----+
| pixfirewall# capture arp ethernet-type arp interface outside
| pixfirewall# show capture
| capture arp ethernet-type arp interface outside
| pixfirewall# show capture arp
| 6 packets captured, 6 packets to be shown
| 10:46:25.452369 arp who-has 209.165.200.225 (ff:ff:ff:ff:ff:ff)
| tell 209.165.200.235
| 10:46:26.312850 arp who-has 209.165.201.2 tell 209.165.200.227
| 10:46:26.392283 arp who-has 209.165.200.225 (ff:ff:ff:ff:ff:ff)
| tell 209.165.200.235
| 10:46:28.923368 arp who-has 209.165.200.226 (ff:ff:ff:ff:ff:ff)
| tell 209.165.200.235
| 10:46:29.255998 arp who-has 209.165.202.129
| tell 209.165.202.130 (0:2:b9:45:bf:7b)
| 10:46:29.256136 arp reply 209.165.202.129 is-at 0:a0:c9:86:8e:9c
+-----+
```

[Example 9-9](#) illustrates a packet capture of PPPoE discovery packets:

**Example 9-9 Capturing PPPoE Discovery**

```
+-----+
| pixfirewall# capture pppoe ethernet-type pppoe interface outside
| pixfirewall(config)# show capture
| capture pppoe ethernet-type pppoe interface outside
| pixfirewall(config)# show capture pppoe
| 3 packets captured, 3 packets to be shown
| 02:13:21.844408 ffff.ffff.2ac5 ffff.ffff.ffff 0x8863 32:
| 1109 0000 000c 0101 0000 0103 0004 386c
| f280
| 02:13:25.841738 ffff.ffff.3cc0 ffff.ffff.ffff 0x8863 32:
| 1109 0000 000c 0101 0000 0103 0004 386c
| f280
| 02:13:33.841875 ffff.ffff.76c0 ffff.ffff.ffff 0x8863 32:
| 1109 0000 000c 0101 0000 0103 0004 386c
| f280
+-----+
```

[Example 9-9](#) illustrates a packet capture on multiple interfaces. The example captures an FTP session to an FTP server at host 209.165.202.129.

**Example 9-10 Capturing On Multiple Interfaces**

```

-----
| pixfirewall(config)# access-list ftp tcp any host 209.165.202.129 eq ftp |
| pixfirewall(config)# access-list ftp tcp host 209.165.202.129 eq ftp any |
| pixfirewall# capture ftp access-list ftp |
| pixfirewall# capture ftp interface inside interface outside |
| pixfirewall# show capture |
| capture ftp access-list ftp interface outside interface inside |
| pixfirewall# |
| pixfirewall# show capture ftp |
| 5 packets captured, 5 packets to be shown |
| 11:21:17.705041 10.1.1.15.2158 > 10.1.1.15.2158: |
|           S 3027585165:3027585165(0) win 512 <mss 1460> |
| 11:21:17.705133 209.165.202.130.2158 > 209.165.202.130.2158: |
|           S 4192390209:4192390209(0) win 512 <mss 1380> |
| 11:21:17.705651 10.1.1.15.2158 > 10.1.1.15.2158: |
|           . ack 3463843411 win 32120 |
| 11:21:17.705667 209.165.202.130.2158 > 209.165.202.130.2158: |
|           . ack 3463843411 win 32120 |
| 11:21:20.784337 10.1.1.15.2158 > 10.1.1.15.2158: |
. ack 3463843521 win 32120

```

## IDS Syslog Messages

PIX Firewall lists single-packet (*atomic*) Cisco Intrusion Detection System (IDS) signature messages via syslog. Refer to *Cisco PIX Firewall System Log Messages* for a list of the supported messages. You can view this document online at the following website:

[http://www.cisco.com/univercd/cc/td/doc/product/iaabu/pix/pix\\_61/syslog/index.htm](http://www.cisco.com/univercd/cc/td/doc/product/iaabu/pix/pix_61/syslog/index.htm)

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 ip_addr to ip_addr 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
```

**Note**

Cisco IDS signature number 1101 is not supported by PIX Firewall. When an unsupported signature number is entered, PIX Firewall returns an error message.

Table 9-5 lists the values and the meaning of each syslog output parameter.

**Table 9-5 Syslog Output Values**

| Syslog Value | Meaning                                                                                                                                                                                                                                                                                                                                                                                                            |
|--------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| sig_num      | The signature number. Refer to the <i>Cisco Secure Intrusion Detection System Version 2.2.1 User Guide</i> for more information. You can view the “NSDB and Signatures” chapter from this guide at the following website:<br><a href="http://www.cisco.com/univercd/cc/td/doc/product/iaabu/csids/csids1/csidsug/sigs.htm">http://www.cisco.com/univercd/cc/td/doc/product/iaabu/csids/csids1/csidsug/sigs.htm</a> |
| sig_msg      | The signature message—approximately the same as the NetRanger signature message.                                                                                                                                                                                                                                                                                                                                   |
| ip_addr      | The local to remote address to which the signature applies.                                                                                                                                                                                                                                                                                                                                                        |
| int_name     | The name of the interface on which the signature originated.                                                                                                                                                                                                                                                                                                                                                       |

Table 9-6 summarizes the commands that you can use to determine the messages that are displayed.

**Table 9-6 Commands to Control Syslog Messages**

| Command                                                      | Effect                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                            |
|--------------------------------------------------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <code>ip audit signature signature_number disable</code>     | Attaches a global policy to a signature. Used to disable or exclude a signature from auditing.                                                                                                                                                                                                                                                                                                                                                                                                                                                                    |
| <code>no ip audit signature signature_number</code>          | Removes the policy from a signature. Used to reenableView a signature.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                            |
| <code>show ip audit signature [signature_number]</code>      | Displays disabled signatures.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                     |
| <code>ip audit info [action [alarm] [drop] [reset]]</code>   | Specifies the default action to be taken for signatures classified as informational signatures. The <b>alarm</b> option indicates that when a signature match is detected in a packet, PIX Firewall reports the event to all configured syslog servers. The <b>drop</b> option drops the offending packet. The <b>reset</b> option drops the offending packet and closes the connection if it is part of an active connection. The default is <b>alarm</b> . To cancel event reactions, specify the <b>ip audit info</b> command without an <b>action</b> option. |
| <code>no ip audit info</code>                                | Sets the action to be taken for signatures classified as informational and reconnaissance to the default action.                                                                                                                                                                                                                                                                                                                                                                                                                                                  |
| <code>show ip audit info</code>                              | Displays the default informational actions.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                       |
| <code>ip audit attack [action [alarm] [drop] [reset]]</code> | Specifies the default actions to be taken for attack signatures. The <b>action</b> options are as previously described.                                                                                                                                                                                                                                                                                                                                                                                                                                           |

Table 9-6 Commands to Control Syslog Messages (continued)

| Command                                                                             | Effect                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                       |
|-------------------------------------------------------------------------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <code>no ip audit attack</code>                                                     | Sets the action to be taken for attack signatures to the default action.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                     |
| <code>show ip audit attack</code>                                                   | Displays the default attack actions. 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 the user 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. |
| <code>ip audit name <i>audit_name</i> info [action [alarm] [drop] [reset]]</code>   | All informational signatures except those disabled or excluded by the <b>ip audit signature</b> command are considered part of the policy. The actions are the same as described previously.                                                                                                                                                                                                                                                                                                                                                                                                                                                                 |
| <code>no ip audit name <i>audit_name</i> [info]</code>                              | Remove the audit policy <i>audit_name</i> .                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                  |
| <code>ip audit name <i>audit_name</i> attack [action [alarm] [drop] [reset]]</code> | All attack signatures except those disabled or excluded by the <b>ip audit signature</b> command are considered part of the policy. The actions are the same as described previously.                                                                                                                                                                                                                                                                                                                                                                                                                                                                        |
| <code>no ip audit name <i>audit_name</i> [attack]</code>                            | Removes the audit specification <i>audit_name</i> .                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                          |
| <code>show ip audit name [name [info   attack]]</code>                              | Displays all audit policies or specific policies referenced by name and possibly type.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                       |
| <code>ip audit interface <i>if_name</i> <i>audit_name</i></code>                    | Applies an audit specification or policy (via the <b>ip audit name</b> command) to an interface.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                             |
| <code>no ip audit interface [<i>if_name</i>]</code>                                 | Removes a policy from an interface.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                          |
| <code>show ip audit interface</code>                                                | Displays the interface configuration.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                        |

## Using SNMP

This section describes how to enable SNMP for monitoring the PIX Firewall with a network management system (NMS). It includes the following topics:

- [Overview](#)
- [MIB Support](#)
- [SNMP CPU Utilization](#)
- [SNMP Usage Notes](#)
- [SNMP Traps](#)
- [Compiling Cisco Syslog MIB Files](#)

- [Using the Firewall and Memory Pool MIBs](#)

## Overview

The **snmp-server** command causes the PIX Firewall to send SNMP traps so that the PIX Firewall can be monitored remotely. Use **snmp-server host** command to specify which systems receive the SNMP traps.

The PIX Firewall SNMP MIB-II groups available are System and Interfaces. The Cisco Firewall MIB and Cisco Memory Pool MIB are also available.

All SNMP values are read only (RO).

Using SNMP, you can monitor system events on the PIX Firewall. SNMP events can be read, but information on the PIX Firewall cannot be changed with SNMP.

The PIX Firewall SNMP traps available to an SNMP management station are as follows:

- Generic traps:
  - Link up and link down (cable connected to the interface or not; cable connected to an interface working or not working)
  - Cold start
  - Authentication failure (mismatched community string)
- Security-related events sent via the Cisco Syslog MIB:
  - Global access denied
  - Failover syslog messages
  - syslog messages

Use CiscoWorks for Windows or any other SNMP V1, MIB-II compliant browser to receive SNMP traps and browse an MIB. SNMP traps occur at UDP port 162.

## MIB Support



### Note

---

The PIX Firewall does not support browsing of the Cisco syslog MIB.

---

You can browse the System and Interface groups of MIB-II. Browsing an MIB is different from sending traps. Browsing means doing an **snmpget** or **snmpwalk** of the MIB tree from the management station to determine values.

The Cisco Firewall MIB and Cisco Memory Pool MIB are available.

PIX Firewall does not support the following in the Cisco Firewall MIB:

- cfwSecurityNotification NOTIFICATION-TYPE
- cfwContentInspectNotification NOTIFICATION-TYPE
- cfwConnNotification NOTIFICATION-TYPE
- cfwAccessNotification NOTIFICATION-TYPE
- cfwAuthNotification NOTIFICATION-TYPE
- cfwGenericNotification NOTIFICATION-TYPE

## SNMP CPU Utilization

PIX Firewall version 6.2 introduces support for monitoring CPU utilization through SNMP. This feature allows network administrators to monitor PIX Firewall CPU usage using SNMP management software, (such as HP OpenView) for capacity planning.

This functionality is implemented through support for the `cpmCPUTotalTable` of the Cisco Process MIB (CISCO-PROCESS-MIB.my.) The other two tables in the MIB, `cpmProcessTable` and `cpmProcessExtTable` are not supported in this release.

Each row of the `cpmCPUTotalTable` consists of the following five elements:

- Index of each CPU



**Note** Because all current PIX Firewall hardware platforms support a single CPU, PIX Firewall returns only one row from `cpmCPUTotalTable` and the index is always 1.

- `entPhysicalIndex` of the physical entity for which the CPU statistics in this entry are maintained (the value of this object will be zero because the `entPhysicalTable` of Entity MIB is not supported on the PIX SNMP agent)
- Overall CPU busy percentage in the last five-second period
- Overall CPU busy percentage in the last one-minute period
- Overall CPU busy percentage in the last five-minute period

The values of the last three elements will be the same as the output of the `show cpu usage` command.

**Table 9-7 CPU Utilization MIB Variables**

| MIB object name                       | Description                                                                                                                                    |
|---------------------------------------|------------------------------------------------------------------------------------------------------------------------------------------------|
| <code>cpmCPUTotalIndex</code>         | The value of this object will be zero because the <code>entPhysicalTable</code> of Entity MIB is not supported on the PIX Firewall SNMP agent. |
| <code>cpmCPUTotalPhysicalIndex</code> | The value of this object will be zero because the <code>entPhysicalTable</code> of Entity MIB is not supported on the PIX Firewall SNMP agent. |
| <code>cpmCPUTotal5sec</code>          | Overall CPU busy percentage in the last five-second period.                                                                                    |
| <code>cpmCPUTotal1min</code>          | Overall CPU busy percentage in the last one-minute period.                                                                                     |
| <code>cpmCPUTotal5min</code>          | Overall CPU busy percentage in the last five-minute period.                                                                                    |

PIX Firewall does not support the following new MIB objects in the `cpmCPUTotalTable`:

- `cpmCPUTotal5secRev`
- `cpmCPUTotal1minRev`
- `cpmCPUTotal5minRev`

## SNMP Usage Notes

- The MIB-II ifEntry.ifAdminStatus object returns 1 if the interface is accessible and 2 if you administratively shut down the interface using the **shutdown** option of the **interface** command.
- The SNMP “ifOutUcastPkts” object now correctly returns the outbound packet count.
- Syslog messages generated by the SNMP module now specify the interface name instead of an interface number.

## SNMP Traps

Traps are different than browsing; they are unsolicited “comments” from the managed device to the management station for certain events, such as link up, link down, and syslog event generated.

An SNMP object ID (OID) for PIX Firewall displays in SNMP event traps sent from the PIX Firewall. PIX Firewall provides system OID in SNMP event traps & SNMP mib-2.system.sysObjectID variable based on the hardware platform.

Table 9-8 lists the system OID in PIX Firewall platforms:

**Table 9-8 System OID in PIX Firewall Platforms**

| PIX Firewall Platform | System OID                                       |
|-----------------------|--------------------------------------------------|
| 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 (original PIX Firewall OID) |

The SNMP service running on the PIX Firewall performs two different functions:

- Replies to SNMP requests from management stations (also known as an SNMP client)
- Sends traps (event notifications) to management stations or other devices that are registered to receive them from the PIX Firewall. PIX Firewall supports two types of traps: generic traps and syslog traps.

## Receiving Requests and Sending Syslog Traps

Follow these steps to receive requests and send traps from the PIX Firewall to an SNMP management station:

- 
- Step 1** Identify the IP address of the SNMP management station with the **snmp-server host** command.
- Step 2** Set the **snmp-server** options for **location**, **contact**, and the **community** password as required.

If you only want to send the cold start, link up, and link down generic traps, no further configuration is required.

If you only want to receive SNMP requests, no further configuration is required.

**Step 3** Add an **snmp-server enable traps** command statement.

**Step 4** Set the logging level with the **logging history** command:

```
logging history debugging
```

We recommend that you use the **debugging** level during initial set up and during testing. Thereafter, set the level from **debugging** to a lower value for production use.

(The **logging history** command sets the severity level for SNMP syslog messages.)

**Step 5** Start sending syslog traps to the management station with the **logging on** command.

**Step 6** To disable sending syslog traps, use the **no logging on** command or the **no snmp-server enable traps** command.

The commands in [Example 9-11](#) specify that PIX Firewall can receive the SNMP requests from host 192.168.3.2 on the inside interface but does not send SNMP syslog traps to any host.

#### **Example 9-11 Enabling SNMP**

```
snmp-server host 192.168.3.2
snmp-server location building 42
snmp-server contact polly hedra
snmp-server community ohwhatakeyisthee
```

The **location** and **contact** commands identify where the host is and who administers it. The **community** command specifies the password in use at the PIX Firewall SNMP agent and the SNMP management station for verifying network access between the two systems.

## Compiling Cisco Syslog MIB Files

To receive security and failover SNMP traps from the PIX Firewall, compile the Cisco SMI MIB and the Cisco syslog MIB into your SNMP management application. If you do not compile the Cisco syslog MIB into your application, you only receive traps for link up or down, firewall cold start and authentication failure.

You can select Cisco MIB files for PIX Firewall and other Cisco products from the following website:

<http://www.cisco.com/public/sw-center/netmgmt/cmtk/mibs.shtml>

From this page, select **PIX Firewall** from the Cisco Secure & VPN selection list.

Follow these steps to compile Cisco syslog MIB files into your browser using CiscoWorks for Windows (SNMPc):

**Step 1** Get the Cisco syslog MIB files.

**Step 2** Start SNMPc.

**Step 3** Click **Config>Compile MIB**.

**Step 4** Scroll to the bottom of the list, and click the last entry.

**Step 5** Click **Add**.

**Step 6** Find the Cisco syslog MIB files.



**Note** With certain applications, only files with a .mib extension may show in the file selection window of the SNMPc. The Cisco syslog MIB files with the .my extension will not be shown. In this case, you should manually change the .my extension to a .mib extension.

**Step 7** Click CISCO-FIREWALL-MIB.my (CISCO-FIREWALL-MIB.mib) and click **OK**.

**Step 8** Scroll to the bottom of the list, and click the last entry.

**Step 9** Click **Add**.

**Step 10** Find the file CISCO-MEMORY-POOL-MIB.my (CISCO-MEMORY-POOL-MIB.mib) and click **OK**.

**Step 11** Scroll to the bottom of the list, and click the last entry.

**Step 12** Click **Add**.

**Step 13** Find the file CISCO-SMI.my (CISCO-SMI.mib) and click **OK**.

**Step 14** Scroll to the bottom of the list, and click the last entry.

**Step 15** Click **Add**.

**Step 16** Find the file CISCO-SYSLOG-MIB.my (CISCO-SYSLOG-MIB.mib) and click **OK**.

**Step 17** Click **Load All**.

**Step 18** If there are no errors, restart SNMPc.



**Note** These instructions are only for SNMPc (CiscoWorks for Windows).

## Using the Firewall and Memory Pool MIBs

The Cisco Firewall and Memory Pool MIBs let you poll failover and system status.

This section contains the following topics:

- [ipAddrTable Notes](#)
- [Viewing Failover Status](#)
- [Verifying Memory Usage](#)
- [Viewing The Connection Count](#)
- [Viewing System Buffer Usage](#)

In the tables that follow in each section, the meaning of each returned value is shown in parentheses.

### ipAddrTable Notes

Use of the SNMP ip.ipAddrTable entry requires that all interfaces have unique addresses. If interfaces have not been assigned IP addresses, by default, their IP addresses are all set to 127.0.0.1. Having duplicate IP addresses causes the SNMP management station to loop indefinitely. The workaround is to assign each interface a different address. For example, you can set one address to 127.0.0.1, another to 127.0.0.2, and so on.

SNMP uses a sequence of GetNext operations to traverse the MIB tree. Each GetNext request is based on the result of the previous request. Therefore, if two consecutive interfaces have the same IP 127.0.0.1 (table index), the GetNext function returns 127.0.0.1, which is correct; however, when SNMP generates the next GetNext request using the same result (127.0.0.1), the request is identical to the previous one, which causes the management station to loop infinitely.

For example:

```
GetNext(ip.ipAddrTable.ipAddrEntry.ipAdEntAddr.127.0.0.1)
```

In SNMP protocol, the MIB table index should be unique for the agent to identify a row from the MIB table. The table index for ip.ipAddrTable is the PIX Firewall interface IP address, so the IP address should be unique; otherwise, the SNMP agent will get confused and may return information of another interface (row), which has the same IP (index).

## Viewing Failover Status

The Cisco Firewall MIB's cfwHardwareStatusTable lets you determine whether failover is enabled and which unit is active. The Cisco Firewall MIB indicates failover status by two rows in the cfwHardwareStatusTable object. From the PIX Firewall command line, you can view failover status with the **show failover** command. You can access the object table from the following path:

```
.iso.org.dod.internet.private.enterprises.cisco.ciscoMgmt.ciscoFirewallMIB.  
ciscoFirewallMIBObjects.cfwSystem.cfwStatus.cfwHardwareStatusTable
```

Table 9-9 lists which objects provide failover information.

**Table 9-9 Failover Status Objects**

| Object                           | Object Type     | Row 1: Returned if Failover is Disabled | Row 1: Returned if Failover is Enabled                          | Row 2: Returned if Failover is Enabled                          |
|----------------------------------|-----------------|-----------------------------------------|-----------------------------------------------------------------|-----------------------------------------------------------------|
| cfwHardwareType<br>(table index) | Hardware        | 6 (If primary unit)                     | 6 (If primary unit)                                             | 7 (If secondary unit)                                           |
| cfwHardwareInformation           | SnmpAdminString | blank                                   | blank                                                           | blank                                                           |
| cfwHardwareStatusValue           | HardwareStatus  | 0 (Not used)                            | active or 9 (If active unit) or standby or 10 (If standby unit) | active or 9 (If active unit) or standby or 10 (If standby unit) |
| cfwHardwareStatusDetail          | SnmpAdminString | Failover Off                            | blank                                                           | blank                                                           |

In the HP OpenView Browse MIB application's "MIB values" window, if failover is disabled, a sample MIB query yields the following information:

```
cfwHardwareInformation.6 :  
cfwHardwareInformation.7 :  
cfwHardwareStatusValue.6 :0  
cfwHardwareStatusValue.7 :0  
cfwHardwareStatusDetail.6 :Failover Off  
cfwHardwareStatusDetail.7 :Failover Off
```

From this listing, the table index, cfwHardwareType, appears as either .6 or .7 appended to the end of each of the subsequent objects. The cfwHardwareInformation field is blank, the cfwHardwareStatusValue is 0, and the cfwHardwareStatusDetail contains **Failover Off**, which indicates the failover status.

When failover is enabled, a sample MIB query yields the following information:

```

cfwHardwareInformation.6 :
cfwHardwareInformation.7 :
cfwHardwareStatusValue.6 : active
cfwHardwareStatusValue.7 : standby
cfwHardwareStatusDetail.6 :
cfwHardwareStatusDetail.7 :

```

In this listing, only the `cfwHardwareStatusValue` contains values, either **active** or **standby** to indicate the status of each unit.

## Verifying Memory Usage

You can determine how much free memory is available with the Cisco Memory Pool MIB. From the PIX Firewall command line, memory usage is viewed with the **show memory** command. The following is sample output from the **show memory** command.

```

show memory
16777216 bytes total, 5595136 bytes free

```

You can access the MIB objects from the following path:

```

.iso.org.dod.internet.private.enterprises.cisco.ciscoMgmt.ciscoMemoryPoolMIB.
ciscoMemoryPoolObjects.ciscoMemoryPoolTable

```

[Table 9-10](#) lists which objects provide memory usage information.

**Table 9-10** Memory Usage Objects

| Object                               | Object Type          | Returned Value                                                                         |
|--------------------------------------|----------------------|----------------------------------------------------------------------------------------|
| ciscoMemoryPoolType<br>(table index) | CiscoMemoryPoolTypes | <b>1</b> (Processor memory)                                                            |
| ciscoMemoryPoolName                  | DisplayString        | PIX Firewall system memory                                                             |
| ciscoMemoryPoolAlternate             | Integer32            | <b>0</b> (No alternate memory pool)                                                    |
| ciscoMemoryPoolValid                 | TruthValue           | <b>true</b> (Means that the values of the remaining objects are valid)                 |
| ciscoMemoryPoolUsed                  | Gauge32              | <i>integer</i> (Number of bytes currently in use—the total bytes minus the free bytes) |
| ciscoMemoryPoolFree                  | Gauge32              | <i>integer</i> (Number of bytes currently free)                                        |
| ciscoMemoryPoolLargestFree           | Gauge32              | <b>0</b> (Information not available)                                                   |

In the HP OpenView Browse MIB application's "MIB values" window a sample MIB query yields the following information:

```

ciscoMemoryPoolName.1 :PIX system memory
ciscoMemoryPoolAlternate.1 :0
ciscoMemoryPoolValid.1 :true
ciscoMemoryPoolUsed.1 :12312576
ciscoMemoryPoolFree.1 :54796288
ciscoMemoryPoolLargestFree.1 :0

```

From this listing, the table index, `ciscoMemoryPoolName`, appears as the **.1** value at the end of each subsequent object value. The `ciscoMemoryPoolUsed` object lists the number of bytes currently in use, **12312576**, and the `ciscoMemoryPoolFree` object lists the number of bytes currently free **54796288**. The other objects always list the values described in [Table 9-10](#).

## Viewing The Connection Count

You can view the number of connections in use from the `cfwConnectionStatTable` in the Cisco Firewall MIB. From the PIX Firewall command line, you can view the connection count with the **show conn** command. The following is sample output from the **show conn** command to demonstrate where the information in `cfwConnectionStatTable` originates.

```
show conn
15 in use, 88 most used
```

The `cfwConnectionStatTable` object table can be accessed from the following path:

```
.iso.org.dod.internet.private.enterprises.cisco.ciscoMgmt.ciscoFirewallMIB.
ciscoFirewallMIBObjects.cfwSystem.cfwStatistics.cfwConnectionStatTable
```

Table 9-11 lists which objects provide connection count information.

**Table 9-11 Connection Count Objects**

| Object                                                 | Object Type     | Row 1: Returned Value                                                | Row 2: Returned Value                                                            |
|--------------------------------------------------------|-----------------|----------------------------------------------------------------------|----------------------------------------------------------------------------------|
| <code>cfwConnectionStatService</code><br>(Table index) | Services        | <b>40</b> (IP protocol)                                              | <b>40</b> (IP protocol)                                                          |
| <code>cfwConnectionStatType</code><br>(Table index)    | ConnectionStat  | <b>6</b> (Current connections in use)                                | <b>7</b> (High)                                                                  |
| <code>cfwConnectionStatDescription</code>              | SnmpAdminString | <b>number of connections currently in use by the entire firewall</b> | <b>highest number of connections in use at any one time since system startup</b> |
| <code>cfwConnectionStatCount</code>                    | Counter32       | <b>0</b> (Not used)                                                  | <b>0</b> (Not used)                                                              |
| <code>cfwConnectionStatValue</code>                    | Gauge32         | <i>integer</i> (In use number)                                       | <i>integer</i> (Most used number)                                                |

In the HP OpenView Browse MIB application's "MIB values" window a sample MIB query yields the following information:

```
cfwConnectionStatDescription.40.6 :number of connections currently in use by the entire firewall
cfwConnectionStatDescription.40.7 :highest number of connections in use at any one time since system startup
cfwConnectionStatCount.40.6 :0
cfwConnectionStatCount.40.7 :0
cfwConnectionStatValue.40.6 :15
cfwConnectionStatValue.40.7 :88
```

From this listing, the table index, `cfwConnectionStatService`, appears as the **.40** appended to each subsequent object and the table index, `cfwConnectionStatType`, appears as either **.6** to indicate the number of connections in use or **.7** to indicate the most used number of connections. The `cfwConnectionStatValue` object then lists the connection count. The `cfwConnectionStatCount` object always returns **0** (zero).

## Viewing System Buffer Usage

You can view the system buffer usage from the Cisco Firewall MIB in multiple rows of the `cfwBufferStatsTable`. The system buffer usage provides an early warning of the PIX Firewall reaching the limit of its capacity. On the command line, you can view this information with the **show blocks** command. The following is sample output from the **show blocks** command to demonstrate how `cfwBufferStatsTable` is populated.

```
show blocks
SIZE    MAX    LOW    CNT
  4     1600  1600  1600
  80     100   97     97
  256    80    79     79
 1550   780   402   404
65536   8     8     8
```

You can view `cfwBufferStatsTable` at the following path:

```
.iso.org.dod.internet.private.enterprises.cisco.ciscoMgmt.ciscoFirewallMIB.
ciscoFirewallMIBObjects.cfwSystem.cfwStatistics.cfwBufferStatsTable
```

Table 9-12 lists the objects required to view the system block usage.

**Table 9-12 System Block Usage Objects**

| Object                                          | Object Type        | First Row: Returned Value                                                                                         | Next Row: Returned Value                                                                                                    | Next Row: Returned Value                                                                                          |
|-------------------------------------------------|--------------------|-------------------------------------------------------------------------------------------------------------------|-----------------------------------------------------------------------------------------------------------------------------|-------------------------------------------------------------------------------------------------------------------|
| <code>cfwBufferStatSize</code><br>(Table index) | Unsigned32         | <i>integer</i> (SIZE value; for example, 4 for a 4-byte block)                                                    | <i>integer</i> (SIZE value; for example, 4 for a 4-byte block)                                                              | <i>integer</i> (SIZE value; for example, 4 for a 4-byte block)                                                    |
| <code>cfwBufferStatType</code><br>(Table index) | ResourceStatistics | 3 (MAX)                                                                                                           | 5 (LOW)                                                                                                                     | 8 (CNT)                                                                                                           |
| <code>cfwBufferStatInformation</code>           | SnmpAdminString    | <b>maximum number of allocated <i>integer</i> byte blocks</b> ( <i>integer</i> is the number of bytes in a block) | <b>fewest <i>integer</i> byte blocks available since system startup</b> ( <i>integer</i> is the number of bytes in a block) | <b>current number of available <i>integer</i> byte blocks</b> ( <i>integer</i> is the number of bytes in a block) |
| <code>cfwBufferStatValue</code>                 | Gauge32            | <i>integer</i> (MAX number)                                                                                       | <i>integer</i> (LOW number)                                                                                                 | <i>integer</i> (CNT number)                                                                                       |



### Note

The three rows repeat for every block size listed in the output of the **show blocks** command.

In the HP OpenView Browse MIB application's "MIB values" window a sample MIB query yields the following information:

```
cfwBufferStatInformation.4.3 :maximum number of allocated 4 byte blocks
cfwBufferStatInformation.4.5 :fewest 4 byte blocks available since system startup
cfwBufferStatInformation.4.8 :current number of available 4 byte blocks
cfwBufferStatInformation.80.3 :maximum number of allocated 80 byte blocks
cfwBufferStatInformation.80.5 :fewest 80 byte blocks available since system startup
cfwBufferStatInformation.80.8 :current number of available 80 byte blocks
cfwBufferStatInformation.256.3 :maximum number of allocated 256 byte blocks
cfwBufferStatInformation.256.5 :fewest 256 byte blocks available since system startup
cfwBufferStatInformation.256.8 :current number of available 256 byte blocks
```

```
cfwBufferStatInformation.1550.3 :maximum number of allocated 1550 byte blocks
cfwBufferStatInformation.1550.5 :fewest 1550 byte blocks available since system startup
cfwBufferStatInformation.1550.8 :current number of available 1550 byte blocks
cfwBufferStatValue.4.3: 1600
cfwBufferStatValue.4.5: 1600
cfwBufferStatValue.4.8: 1600
cfwBufferStatValue.80.3: 400
cfwBufferStatValue.80.5: 396
cfwBufferStatValue.80.8: 400
cfwBufferStatValue.256.3: 1000
cfwBufferStatValue.256.5: 997
cfwBufferStatValue.256.8: 999
cfwBufferStatValue.1550.3: 1444
cfwBufferStatValue.1550.5: 928
cfwBufferStatValue.1550.8: 932
```

From this listing, the first table index, `cfwBufferStatSize`, appears as first number appended to the end of each object, such as `.4` or `.256`. The other table index, `cfwBufferStatType`, appears as `.3`, `.5`, or `.8` after the first index. For each block size, the `cfwBufferStatInformation` object identifies the type of value and the `cfwBufferStatValue` object identifies the number of bytes for each value.

---

