13

# Authentication, Authorization, and Accounting Configuration on the Cisco PIX Firewall

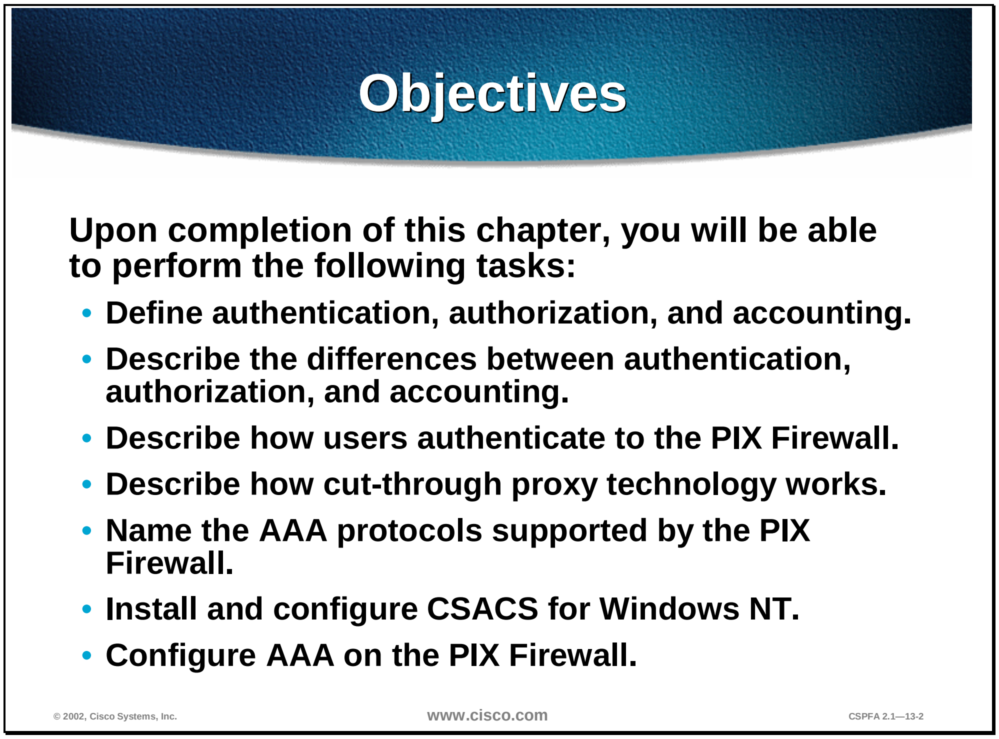
## Overview

This chapter includes the following topics:

- Objectives
- Introduction
- Installation of CSACS for Windows NT
- Authentication configuration
- Authorization configuration
- Accounting configuration
- Troubleshooting the AAA configuration
- Summary
- Lab exercise

# Objectives

This section lists the chapter's objectives.



## Objectives

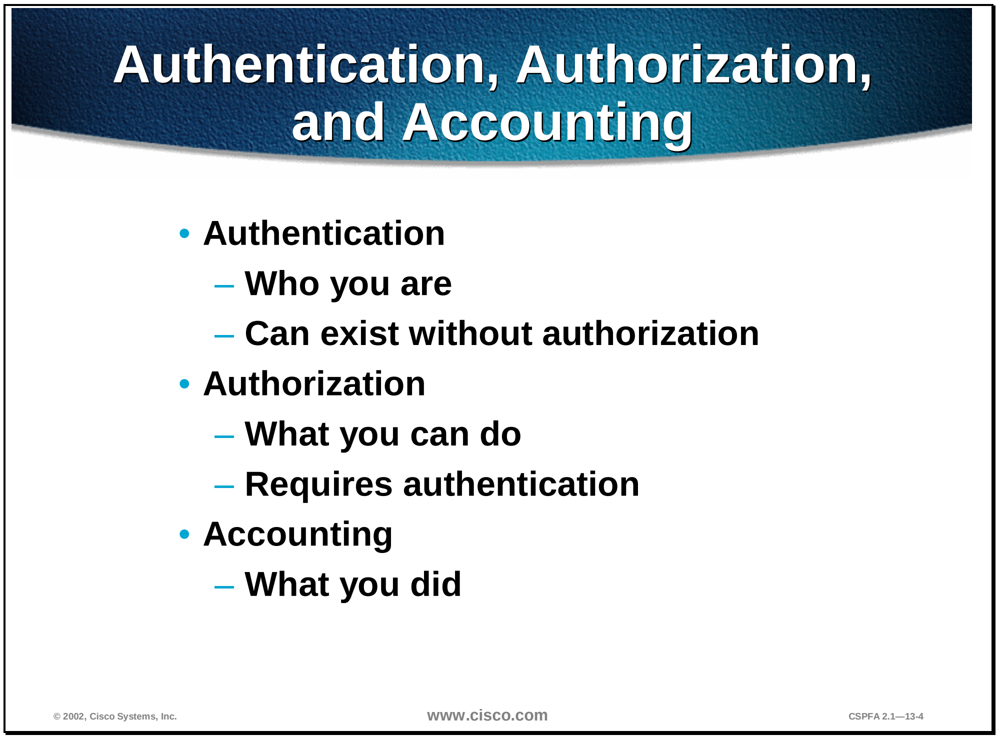
**Upon completion of this chapter, you will be able to perform the following tasks:**

- **Define authentication, authorization, and accounting.**
- **Describe the differences between authentication, authorization, and accounting.**
- **Describe how users authenticate to the PIX Firewall.**
- **Describe how cut-through proxy technology works.**
- **Name the AAA protocols supported by the PIX Firewall.**
- **Install and configure CSACS for Windows NT.**
- **Configure AAA on the PIX Firewall.**

© 2002, Cisco Systems, Inc. www.cisco.com CSPFA 2.1—13-2

# Introduction

This section introduces the authentication, authorization, and accounting concepts and how the Cisco PIX™ Firewall supports them.



## Authentication, Authorization, and Accounting

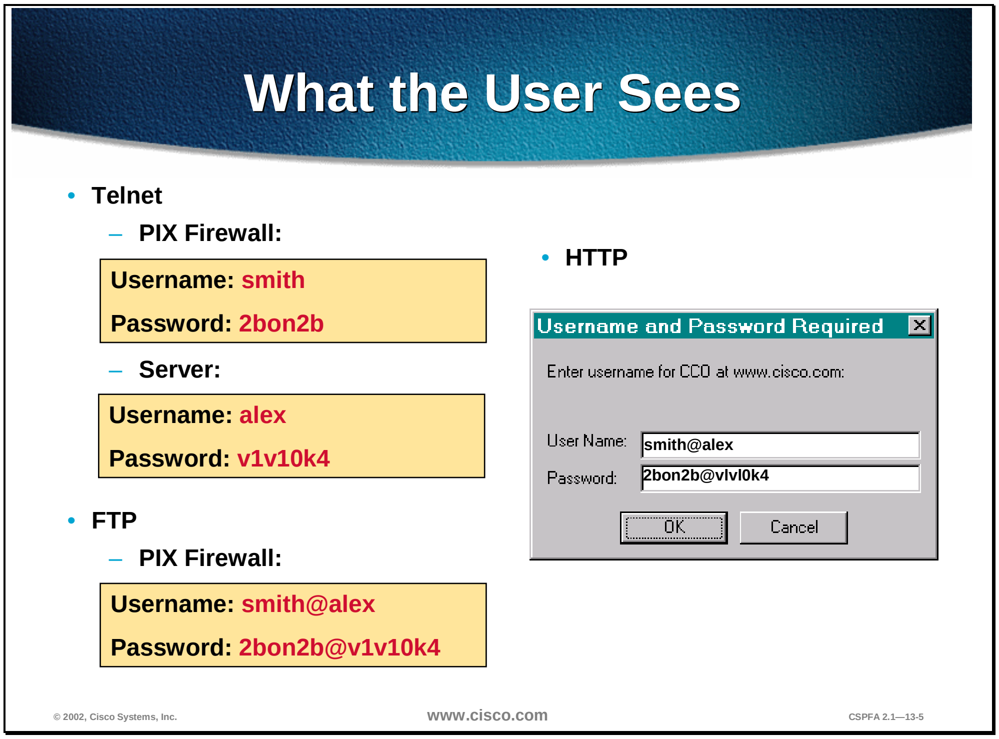
- Authentication
  - Who you are
  - Can exist without authorization
- Authorization
  - What you can do
  - Requires authentication
- Accounting
  - What you did

© 2002, Cisco Systems, Inc. www.cisco.com CSPFA 2.1—13-4

Authentication, Authorization, and Accounting (AAA) is used to tell the PIX Firewall who the user is, what the user can do, and what the user did. Authentication is valid without authorization. Authorization is never valid without authentication.

Suppose you have 100 users inside and you want only six of these users to perform FTP, Telnet, or HTTP outside the network. Tell the PIX Firewall to authenticate outbound traffic and give all 6 users identifications on the Terminal Access Controller Access Control System (TACACS+) or Remote Authentication Dial-In User Service (RADIUS) AAA server. With simple authentication, these six users are authenticated with a username and password, and then permitted outside the network. The other 94 users cannot go outside the network. The PIX Firewall prompts users for their username and password, and then passes their username and password to the TACACS+ or RADIUS AAA server. Depending on the response, the PIX Firewall opens or denies the connection.

Suppose one of these users, “baduser,” is not to be trusted. You want to allow “baduser” to perform FTP, but not HTTP or Telnet, to the outside network. This means you must add authorization, that is, authorize what users can do in addition to authenticating who they are. This is only valid with TACACS+. When you add authorization to the PIX Firewall, it first sends the untrusted user a username and password to the AAA server, then sends an authorization request telling the AAA server what command “baduser” is trying to do. With the server set up properly, “baduser” is allowed to perform FTP but is not allowed to perform HTTP or Telnet.



## What the User Sees

- Telnet
  - PIX Firewall:

    **Username: smith**
    **Password: 2bon2b**

  - Server:

    **Username: alex**
    **Password: v1v10k4**

- FTP
  - PIX Firewall:

    **Username: smith@alex**
    **Password: 2bon2b@v1v10k4**

- HTTP

Username and Password Required

Enter username for CCO at www.cisco.com:

User Name: smith@alex

Password: 2bon2b@vlvl0k4

© 2002, Cisco Systems, Inc. www.cisco.com CSPFA 2.1—13-5

You can authenticate with the PIX Firewall in one of three ways:

- Telnet—You get a prompt generated by the PIX Firewall. You have up to four chances to log in. If the username or password fails after the fourth attempt, the PIX Firewall drops the connection. If authentication and authorization are successful, you are prompted for a user name and password by the destination server.
- FTP—You get a prompt from the FTP program. If you enter an incorrect password, the connection is dropped immediately. If the username or password on the authentication database differs from the username or password on the remote host to which you are accessing via FTP, enter the username and password in the following formats:
  - aaa_username@remote_username
  - aaa_password@remote_password

  The PIX Firewall sends the aaa_username and aaa_password to the AAA server, and if authentication and authorization are successful, the remote_username and remote_password are passed to the destination FTP server.

**Note** Some FTP GUIs do not display challenge values.

- HTTP—You see a pop-up window generated by the web browser. If you enter an incorrect password, you are prompted again. If the username or password on the authentication database differs from the username or password on the remote host to which you are using HTTP to access, enter the username and password in the following formats:
  - aaa_username@remote_username

- aaa_password@remote_password

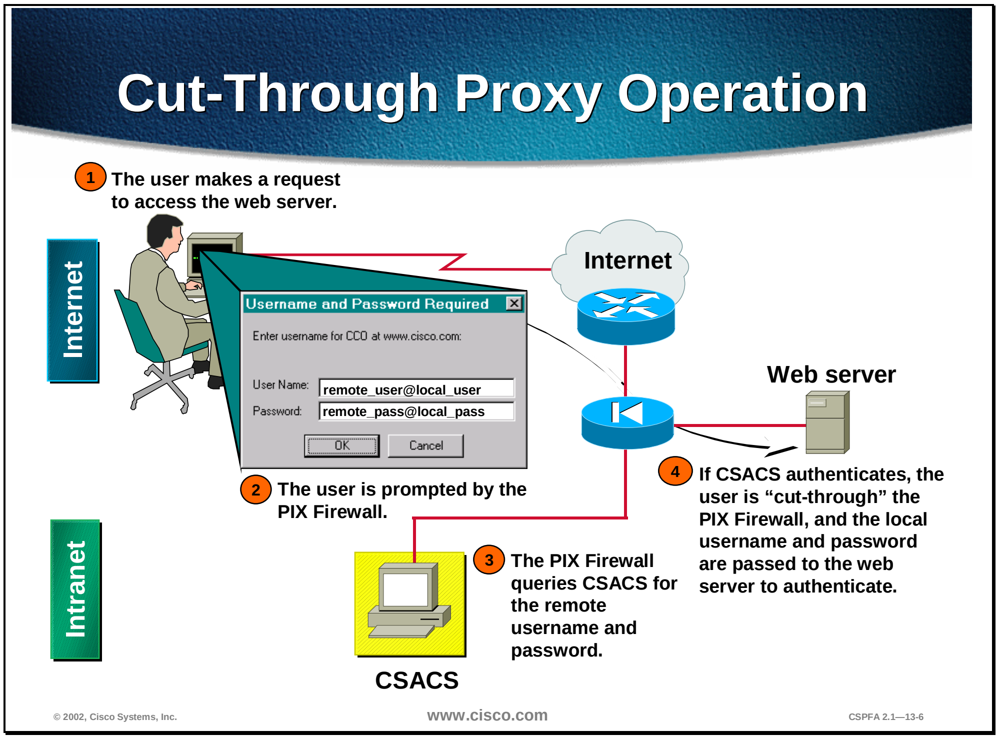
  The PIX Firewall sends the aaa_username and aaa_password to the AAA server, and if authentication and authorization are successful, the remote_username and remote_password are passed to the destination HTTP server.

  Keep in mind that browsers cache usernames and passwords. If you believe that the PIX Firewall should be timing out an HTTP connection but it is not, re-authentication may actually be taking place with the web browser sending the cached username and password back to the PIX Firewall. The Syslog service will show this phenomenon. If Telnet and FTP seem to work normally, but HTTP connections do not, this is usually why.

The PIX Firewall supports authentication usernames up to 127 characters and passwords of up to 63 characters. A password or username may not contain an at (@) character as part of the password or username string.

**Note** If PIX Firewalls are in tandem, Telnet authentication works in the same way as a single PIX Firewall, but FTP and HTTP authentication have additional complexity because you have to enter each password and username with an additional “at” (@) character and password or username for each in-tandem PIX Firewall.

**Note** Once authenticated with HTTP, a user never has to reauthenticate no matter how low the PIX Firewall uauth timeout is set. This is because the browser caches the "Authorization: Basic=Uuhjksdkfhk==" string in every subsequent connection to that particular site. This can only be cleared when the user exits all instances of the web browser and restarts. Flushing the cache is of no use.



## Cut-Through Proxy Operation

1 The user makes a request to access the web server.

2 The user is prompted by the PIX Firewall.

3 The PIX Firewall queries CSACS for the remote username and password.

4 If CSACS authenticates, the user is “cut-through” the PIX Firewall, and the local username and password are passed to the web server to authenticate.

Username and Password Required

Enter username for CCO at www.cisco.com:

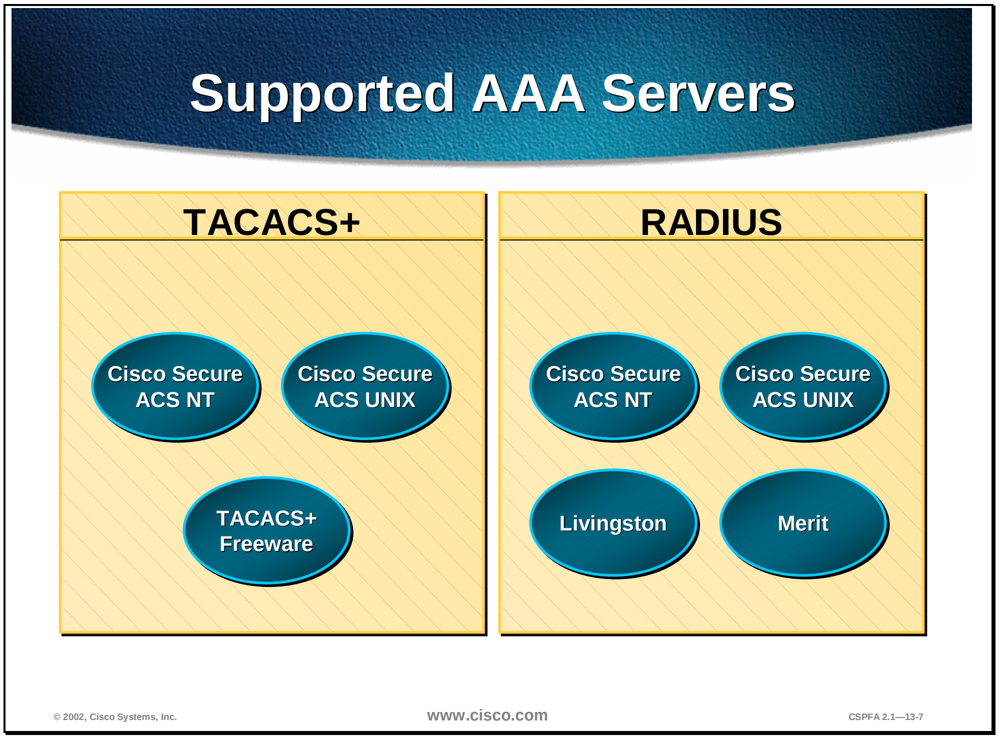
User Name: remote_user@local_user

Password: remote_pass@local_pass

© 2002, Cisco Systems, Inc. www.cisco.com CSPFA 2.1—13-6

The PIX Firewall gains dramatic performance advantages because of the cut-through proxy, a method of transparently verifying the identity of users at the firewall and permitting or denying access to any TCP- or UDP-based application. This method eliminates the price and performance impact that UNIX system-based firewalls impose in similar configurations, and leverages the authentication and authorization services of CSACS.

The PIX Firewall’s cut-through proxy challenges a user initially at the application layer, and then authenticates against standard TACACS or RADIUS+ databases. After the policy is checked, the PIX Firewall shifts the session flow and all traffic flows directly and quickly between the server and the client while maintaining session state information.



## Supported AAA Servers

TACACS+

- Cisco Secure ACS NT
- Cisco Secure ACS UNIX
- TACACS+ Freeware

RADIUS

- Cisco Secure ACS NT
- Cisco Secure ACS UNIX
- Livingston
- Merit

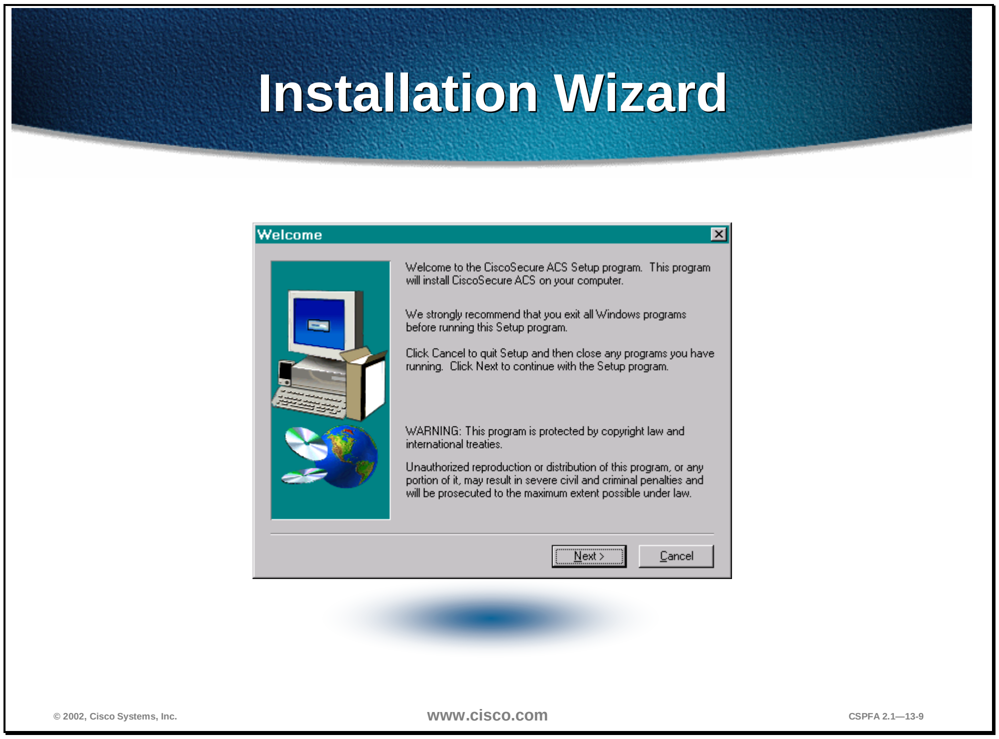
www.cisco.com CSPFA 2.1—13-7

The PIX Firewall supports the following AAA protocols and servers:

- Terminal Access Controller Access Control System Plus (TACACS+)
  - Cisco Secure Access Control Server for Windows NT (CSACS-NT)
  - CSACS for UNIX (CSACS-UNIX)
  - TACACS+ Freeware
- Remote Authentication Dial-In User Service (RADIUS)
  - CSACS for Windows NT (CSACS-NT)
  - CSACS for UNIX (CSACS-UNIX)
  - Livingston
  - Merit

# Installation of CSACS for Windows NT

This section explains how to install the Cisco Secure Access Control Server (CSACS) for Windows NT.



Installation Wizard

Welcome

Welcome to the CiscoSecure ACS Setup program. This program will install CiscoSecure ACS on your computer.

We strongly recommend that you exit all Windows programs before running this Setup program.

Click Cancel to quit Setup and then close any programs you have running. Click Next to continue with the Setup program.

WARNING: This program is protected by copyright law and international treaties.

Unauthorized reproduction or distribution of this program, or any portion of it, may result in severe civil and criminal penalties and will be prosecuted to the maximum extent possible under law.

© 2002, Cisco Systems, Inc. www.cisco.com CSPFA 2.1—13-9

**Note** Close all Windows programs before you run the setup program.

To start installation of CSACS for Windows NT, complete the following steps:

**Step 1** Log in as the local system administrator to the machine on which you are installing CSACS.

**Step 2** Insert the CSACS CD-ROM into your CD-ROM drive. The Installation window opens.

**Step 3** Click **Install**. The Software License Agreement window opens.

**Step 4** Read the Software License Agreement. Click **Accept** to agree to the licensing terms and conditions. The Welcome window opens.

**Step 5** Click **Next**. The Before You Begin window opens.

**Step 6** Verify that each condition is met, and then click the check box for each item. Click **Next**.

**Step 7** Click **Next**. (Click **Explain** for more information on the listed items. If any condition is not met, click **Cancel** to exit the program.)

**Step 8** If all conditions are met, click **Next** to continue.

**Note** If this is a new installation, skip to Step 11.

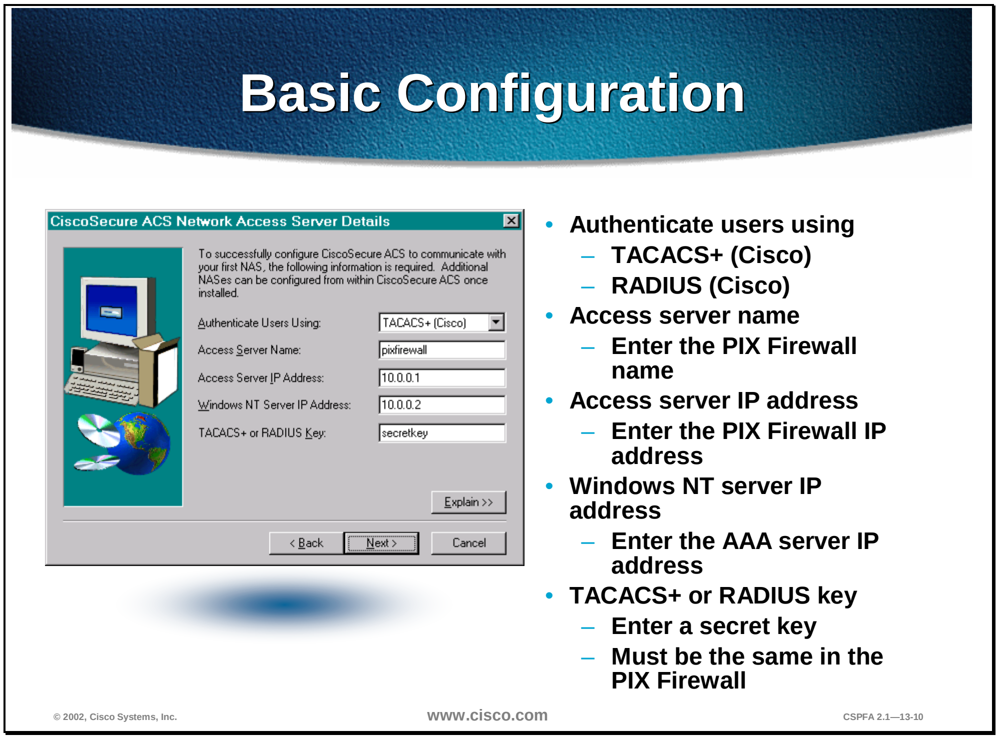
**Step 9** (Optional.) If CSACS is already installed, the Previous Installation window opens. You are prompted to remove the previous version and save the existing database information. To keep the existing data, click **Yes, keep existing database** and click **Next**. To use a new database, deselect the check box and click **Next**. If you checked the check box, the setup program backs up the existing database information and removes the old files. When the files are removed, click **OK**.

**Step 10** If Setup finds an existing configuration, you are prompted whether you want to import the configuration. To keep the existing configuration, click **Yes, import configuration** and click **Next**. To use a new configuration, deselect the check box and click **Next**. The Choose Destination Location window opens.

**Step 11** To install the software in the default directory, click **Next**. To use a different directory, click **Browse** and enter the directory to use. If the directory does not exist, you are prompted to create one. Click **Yes**. The Authentication Database Configuration window opens.

**Step 12** Click the option button for the authentication databases to be used by CSACS. Check the **CSACS Database only** option (the default). Also check the **Windows NT User Database** option. If you select the first option, Cisco Secure ACS will use only the CSACS database for authentication; if you select the second option, CSACS will check both databases.

**Step 13** (Optional.) To limit dial-in access to only those users you specified in the Windows NT User Manager, click the **Yes, reference "Grant dialin permission to user"** setting. Click **Next**. The Network Access Server Details window opens.



## Basic Configuration

CiscoSecure ACS Network Access Server Details

To successfully configure CiscoSecure ACS to communicate with your first NAS, the following information is required. Additional NASes can be configured from within CiscoSecure ACS once installed.

Authenticate Users Using: TACACS+ (Cisco)

Access Server Name: pixfirewall

Access Server IP Address: 10.0.0.1

Windows NT Server IP Address: 10.0.0.2

TACACS+ or RADIUS Key: secretkey

- Authenticate users using
  - TACACS+ (Cisco)
  - RADIUS (Cisco)
- Access server name
  - Enter the PIX Firewall name
- Access server IP address
  - Enter the PIX Firewall IP address
- Windows NT server IP address
  - Enter the AAA server IP address
- TACACS+ or RADIUS key
  - Enter a secret key
  - Must be the same in the PIX Firewall

www.cisco.com CSPFA 2.1—13-10

**Step 14** Complete the following information:

- Authenticate Users Using—Type of security protocol to be used. TACACS+ (Cisco) is the default.
- Access Server Name—Name of the network access server (NAS) that will be using the CSACS services.
- Access Server IP Address—IP address of the NAS that will be using the CSACS services.
- Windows NT Server IP Address—IP address of this Windows NT server.
- TACACS+ or RADIUS Key—Shared secret of the NAS and CSACS. These passwords must be identical to ensure proper function and communication between the NAS and CSACS. Shared secrets are case sensitive. Setup installs the CSACS files and updates the Registry. Click **Next**. The Interface Configuration window opens.

**Step 15** The Interface Configuration options are disabled by default. Click the check box to enable any or all of the options listed. Click **Next**. The Active Service Monitoring window opens.

**Note** Configuration options for these items are displayed in the CSACS interface only if they are enabled. You can disable or enable any or all of these and additional options after installation in the Interface Configuration: Advanced Options window.

**Step 16** To enable the CSACS monitoring service, CSMon, check the **Enable Log-in Monitoring** check box, then select a script to execute when the login process fails the test:

- No Remedial Action—Leave CSACS operating as is.
- Reboot—Reboot the system on which CSACS is running.

- Restart All—(Default.) Restart all CSACS services.
- Restart RADIUS/TACACS+—Restart only RADIUS, TACACS+, or both protocols.

You can also develop your own scripts to be executed if there is a system failure. See the online documentation for more information.

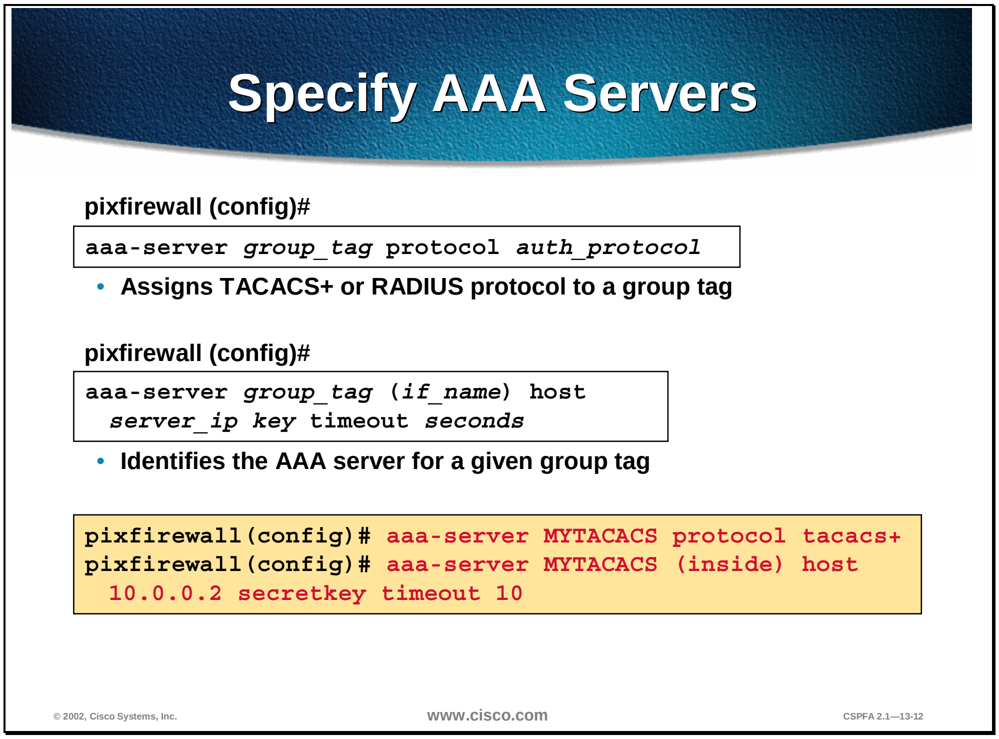
**Step 17** To have CSACS generate an e-mail message when administrator events occur, check the **Enable Mail Notifications** check box, then enter the following information:

- SMTP Mail Server—The name and domain of the sending mail server (for example, server1.company.com).
- Mail account to notify—The complete e-mail address of the intended recipient (for example, msmith@company.com).

**Step 18** Click **Next**. The CSACS Service Initiation window opens. If you do not want to configure a NAS from Setup, click **Next**. To configure a single NAS now, click **Yes, I want to configure Cisco IOS** now. Click **Next**.

# Authentication Configuration

This section discusses how to configure authentication on the PIX Firewall.



## Specify AAA Servers

pixfirewall (config)#

```
aaa-server group_tag protocol auth_protocol
```

- Assigns TACACS+ or RADIUS protocol to a group tag

pixfirewall (config)#

```
aaa-server group_tag (if_name) host
  server_ip key timeout seconds
```

- Identifies the AAA server for a given group tag

```
pixfirewall(config)# aaa-server MYTACACS protocol tacacs+
pixfirewall(config)# aaa-server MYTACACS (inside) host
  10.0.0.2 secretkey timeout 10
```

© 2002, Cisco Systems, Inc. www.cisco.com CSPFA 2.1—13-12

Use the **aaa-server** command to specify AAA server groups. The PIX Firewall lets you define separate groups of TACACS+ or RADIUS servers for specifying different types of traffic, such as a TACACS+ server for inbound traffic and another for outbound traffic. The **aaa** command references the group tag to direct authentication, authorization, or accounting traffic to the appropriate AAA server.

You can have up to 14 tag groups, and each group can have up to 16 AAA servers for a total of up to 256 TACACS+ or RADIUS servers. When a user logs in, the servers are accessed one at a time, starting with the first server you specify in the tag group, until a server responds.

The default configuration provides these following two aaa-server protocols:

- aaa-server MYTACACS protocol tacacs+
- aaa-server RADIUS protocol radius

**Note** If you are upgrading from a previous version of the PIX Firewall and have **aaa** command statements in your configuration, using the default server groups enables you to maintain backward compatibility with the aaa command statements in your configuration

**Note** The previous server type option at the end of the **aaa authentication** and **aaa accounting** commands has been replaced with the aaa-server group tag. Backward compatibility with previous versions is maintained by the inclusion of two default protocols for TACACS+ and RADIUS.

**Note** The PIX Firewall listens for RADIUS on ports 1645 and 1646. If your RADIUS server uses ports 1812 and 1813, you will need to reconfigure it to listen on ports 1645 and 1646.

The syntax for all forms of the **aaa-server** command is as follows:

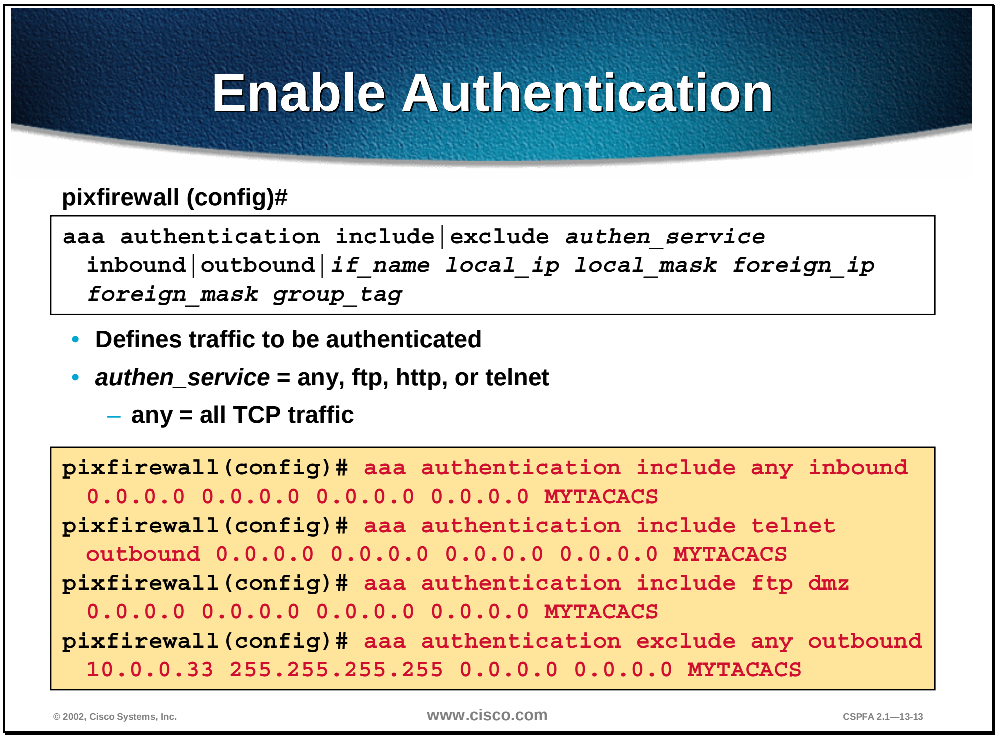
**aaa-server** ***group_tag*** **(*if_name*) host** ***server_ip key*** **timeout** ***seconds***

**no aaa-server** ***group_tag*** **(*if_name*) host** ***server_ip key*** **timeout** ***seconds***

**aaa-server** ***group_tag*** **protocol** ***auth_protocol***

**clear aaa-server [*group_tag*]**

| | |
|---|---|
| ***group_tag*** | An alphanumeric string that is the name of the server group. Use the group_tag in the **aaa** command to associate **aaa authentication, aaa authorization,** and **aaa accounting** command statements with an AAA server. |
| ***if_name*** | The interface name on the side where the AAA server resides. |
| **host *server_ip*** | The IP address of the TACACS+ or RADIUS server. |
| ***key*** | A case-sensitive, alphanumeric keyword of up to 127 characters that is the same value as the key on the TACACS+ server. Any characters entered past 127 are ignored. The key is used between the client and server for encrypting data between them. The key must be the same on both the client and server systems. Spaces are not permitted in the key, but other special characters are.<br>If a key is not specified, encryption does not occur. |
| **timeout *seconds*** | A retransmit timer that specifies the duration that the PIX Firewall retries access. Access to the AAA server is retried four times before choosing the next AAA server. The default is 5 seconds. The maximum time is 30 seconds.<br>For example, if the timeout value is 10 seconds, the PIX Firewall retransmits for 10 seconds and if no acknowledgment is received, tries three times more for a total of 40 seconds to retransmit data before the next AAA server is selected. |
| **protocol *auth_protocol*** | The type of AAA server, either TACACS+ or RADIUS. |



## Enable Authentication

pixfirewall (config)#

```
aaa authentication include|exclude authen_service
  inbound|outbound|if_name local_ip local_mask foreign_ip
  foreign_mask group_tag
```

- Defines traffic to be authenticated
- *authen_service* = any, ftp, http, or telnet
  - any = all TCP traffic

```
pixfirewall(config)# aaa authentication include any inbound
  0.0.0.0 0.0.0.0 0.0.0.0 0.0.0.0 MYTACACS
pixfirewall(config)# aaa authentication include telnet
  outbound 0.0.0.0 0.0.0.0 0.0.0.0 0.0.0.0 MYTACACS
pixfirewall(config)# aaa authentication include ftp dmz
  0.0.0.0 0.0.0.0 0.0.0.0 0.0.0.0 MYTACACS
pixfirewall(config)# aaa authentication exclude any outbound
  10.0.0.33 255.255.255.255 0.0.0.0 0.0.0.0 MYTACACS
```

www.cisco.com

CSPFA 2.1—13-13

The **aaa authentication** command enables or disables user authentication services. When you start a connection via Telnet, FTP, or HTTP, you are prompted for a username and password. A AAA server, designated previously with the **aaa-server** command, verifies whether the username and password are correct. If they are correct, the PIX Firewall's cut-through proxy permits further traffic between the initiating host and the target host.

The **aaa authentication** command is not intended to mandate your security policy. The AAA servers determine whether a user can or cannot access the system, what services can be accessed, and what IP addresses the user can access. The PIX Firewall interacts with Telnet, FTP, and HTTP to display the prompts for logging. You can specify that only a single service be authenticated, but this must agree with the AAA server to ensure that both the firewall and server agree.

For each IP address, one **aaa authentication** command is permitted for inbound connections and one for outbound connections. The PIX Firewall permits only one authentication type per network. For example, if one network connects through the PIX Firewall using TACACS+ for authentication, another network connecting through the PIX Firewall can authenticate with RADIUS, but one network cannot authenticate with both TACACS+ and RADIUS.

**Note** The new include and exclude options are not backward compatible with PIX Firewall versions 5.0 and earlier. If you downgrade to an earlier version, the **aaa authentication** command statements are removed from your configuration.

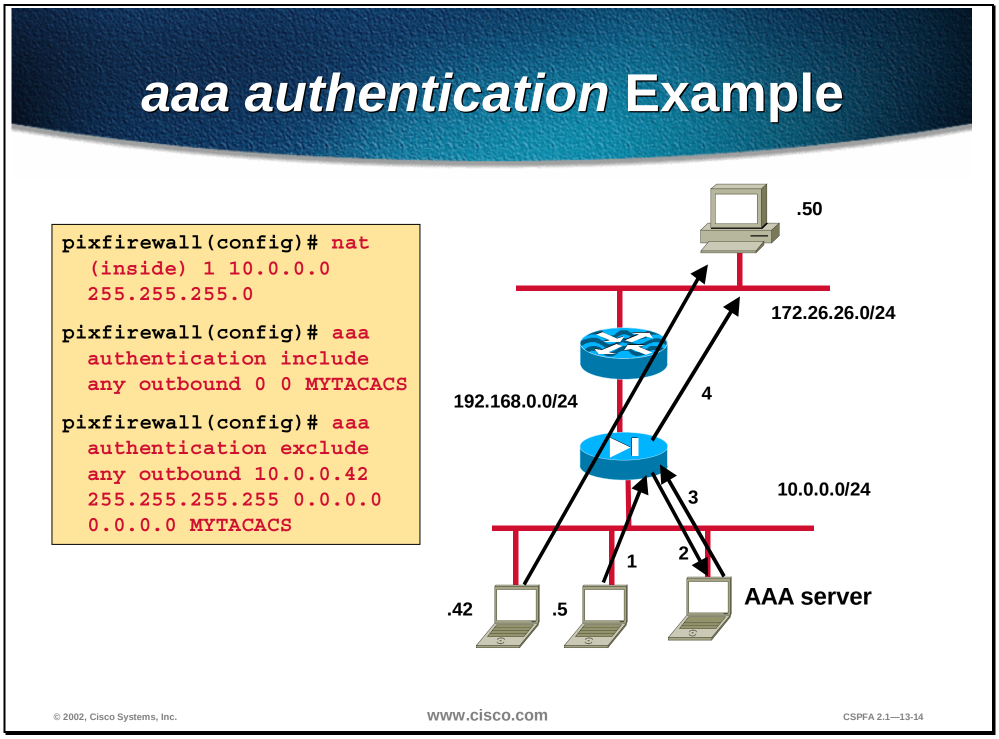
The syntax for all forms of the **aaa authentication** command is as follows:

**aaa authentication include | exclude** ***authen_service*** **inbound | outbound |** ***if_name local_ip local_mask foreign_ip foreign_mask group_tag***

**no aaa authentication [include | exclude** ***authen_service*** **inbound | outbound |** ***if_name local_ip local_mask foreign_ip foreign_mask group_tag*****]**

**clear aaa [authentication include | exclude *authen_service* inbound | outbound | *if_name local_ip local_mask foreign_ip foreign_mask group_tag*]**

| | |
|---|---|
| **include** | Creates a new rule with the specified service to include. |
| **exclude** | Creates an exception to a previously stated rule by excluding the specified service from authentication to the specified host. The exclude parameter improves the former except option by enabling the user to specify a port to exclude to a specific host or hosts. |
| ***authen_service*** | The services that require user authentication before they are let through the firewall. Use any, ftp, http, or telnet. The any value enables authentication for all TCP services. |
| **inbound** | Authenticates inbound connections. Inbound means the connection originates on the outside interface and is being directed to the inside or any other perimeter interface. |
| **outbound** | Authenticate outbound connections. Outbound means the connection originates on the inside and is being directed to the outside or any other perimeter interface. |
| ***if_name*** | Interface name from which users require authentication. Use if_name in combination with the local_ip address and the foreign_ip address to determine where access is sought and from whom. The local_ip address is always on the interface with the highest security level and foreign_ip is always on the lowest. |
| ***local_ip*** | The IP address of the host or network of hosts that you want to be authenticated. You can set this address to 0 to mean all hosts and to let the authentication server decide which hosts are authenticated. |
| ***local_mask*** | Network mask of local_ip. Always specify a specific mask value. Use 0 if the IP address is 0. Use 255.255.255.255 for a host. |
| ***foreign_ip*** | The IP address of the hosts you want to access the local_ip address. Use 0 to mean all hosts. |
| ***foreign_mask*** | Network mask of foreign_ip. Always specify a specific mask value. Use 0 if the IP address is 0. Use 255.255.255.255 for a host. |
| ***group_tag*** | The group tag set with the **aaa-server** command. |



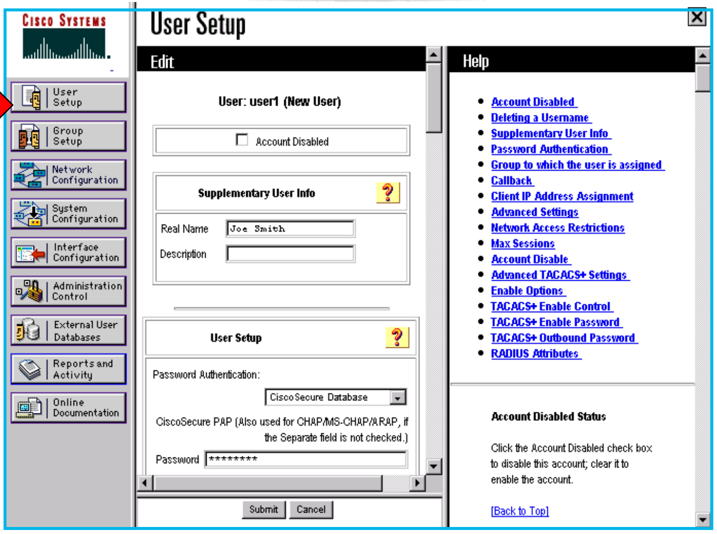
## *aaa authentication* Example

```
pixfirewall(config)# nat
   (inside) 1 10.0.0.0
  255.255.255.0

pixfirewall(config)# aaa
  authentication include
  any outbound 0 0 MYTACACS

pixfirewall(config)# aaa
  authentication exclude
  any outbound 10.0.0.42
  255.255.255.255 0.0.0.0
  0.0.0.0 MYTACACS
```

www.cisco.com
CSPFA 2.1—13-14

In the example above, workstations on the 10.0.0.0 network can originate outbound connections, but users must be authenticated. Host 10.0.0.42, however, is allowed to start outbound connections without being authenticated.

## How to Add Users to CSACS-NT



User Setup

Edit

User: user1 (New User)

□ Account Disabled

Supplementary User Info

Real Name: Joe Smith

Description:

User Setup

Password Authentication: CiscoSecure Database

CiscoSecure PAP (Also used for CHAP/MS-CHAP/ARAP, if the Separate field is not checked.)

Password: ********

Help

- Account Disabled
- Deleting a Username
- Supplementary User Info
- Password Authentication
- Group to which the user is assigned
- Callback
- Client IP Address Assignment
- Advanced Settings
- Network Access Restrictions
- Max Sessions
- Account Disable
- Advanced TACACS+ Settings
- Enable Options
- TACACS+ Enable Control
- TACACS+ Enable Password
- TACACS+ Outbound Password
- RADIUS Attributes

Account Disabled Status

Click the Account Disabled check box to disable this account; clear it to enable the account.

[Back to Top]

© 2002, Cisco Systems, Inc. www.cisco.com CSPFA 2.1—13-15

To add users to the CSACS, complete the following steps:

**Step 1** Click **User Setup** from the navigation bar. The Select window opens.

**Step 2** Enter a name in the User field.

**Note** The username can contain up to 32 characters. Names cannot contain the following special characters: #, ?, ", *, >, <. Leading and trailing spaces are not allowed.

**Step 3** Click **Add/Edit**. The Edit window opens. The username being added or edited appears at the top of the window.

The Edit window contains the following sections:

- Account Disabled
- Supplementary User Info
- User Setup
- Account Disable

## Account Disabled

If you need to disable an account, select the Account Disabled check box in the Account Disabled section to deny access for this user.

**Note** You must click **Submit** to have this action take effect.

## Supplementary User Info

In this section, you can enter supplemental information to appear in each user profile. The fields shown below are available by default; however, you can insert additional fields by clicking Interface Configuration in the navigation bar and then clicking User Data Configuration (configuring supplemental information is optional):

- Real Name—If the username is not the user’s real name, enter the real name here.
- Description—Enter a detailed description of the user.

## User Setup

In the User Setup group box, you can edit or enter the following information for the user as applicable:

- Password Authentication—From the drop-down menu, choose a database to use for username and password authentication. You can select the Windows NT user database or the Cisco Secure database. The Windows NT option authenticates a user with an existing account in the Windows NT User Database located on the same machine as the CSACS server. The Cisco Secure Database option authenticates a user from the local CSACS database. If you select this database, enter and confirm the Password Authentication Protocol (PAP) password to be used. The Separate CHAP/MS-CHAP/ARAP option is not used with the PIX Firewall.

**Note** The Password and Confirm Password fields are required for all authentication methods except for all third-party user databases.

- Group to which the user is assigned—From the Group to which the user is assigned drop-down menu, choose the group to which to assign the user. The user inherits the attributes and operations assigned to the group. By default, users are assigned to the Default Group. Users who authenticate via the Unknown User method who are not found in an existing group are also assigned to the Default Group.
- Callback—This is not used with the PIX Firewall.
- Client IP Address Assignment—This is not used with PIX Firewall.

## Account Disable

The Account Disable group box can be used to define the circumstances under which the user’s account will become disabled.

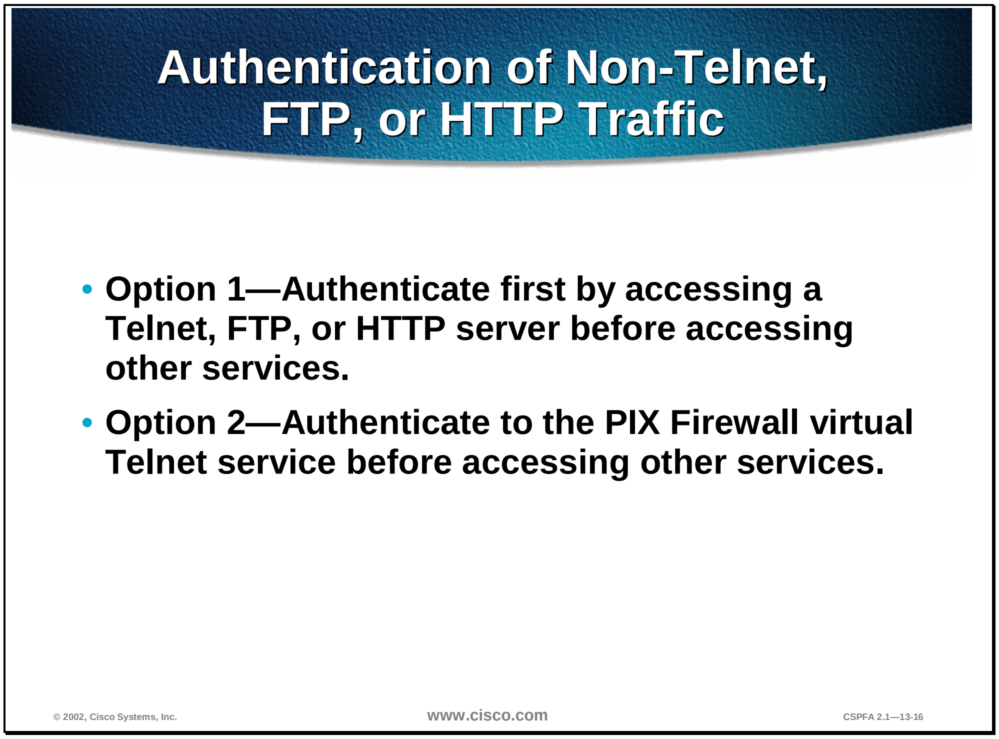
**Note** This is not to be confused with account expiration due to password aging. Password aging is defined for groups only, not for individual users.

- Never radio button—Select to keep the user’s account always enabled. This is the default.

- Disable account if radio button—Click to disable the account under the circumstances you specify in the following fields:
  - Date exceeds—From the drop-down menus, choose the month, date, and year on which to disable the account. The default is 30 days after the user is added.
  - Failed attempts exceed—Select the check box and enter the number of consecutive unsuccessful login attempts to allow before disabling the account. The default is 5.
  - Failed attempts since last successful login—This counter shows the number of unsuccessful login attempts since the last time this user logged in successfully.
- Reset current failed attempts count on submit—If an account is disabled because the failed attempts count has been exceeded, select this check box and click **Submit** to reset the failed attempts counter to 0 and reinstate the account.

If you are using the Windows NT user database, this expiration information is in addition to the information in the Windows NT user account. Changes here do not alter settings configured in Windows NT.

When you have finished configuring all user information, click **Submit**.



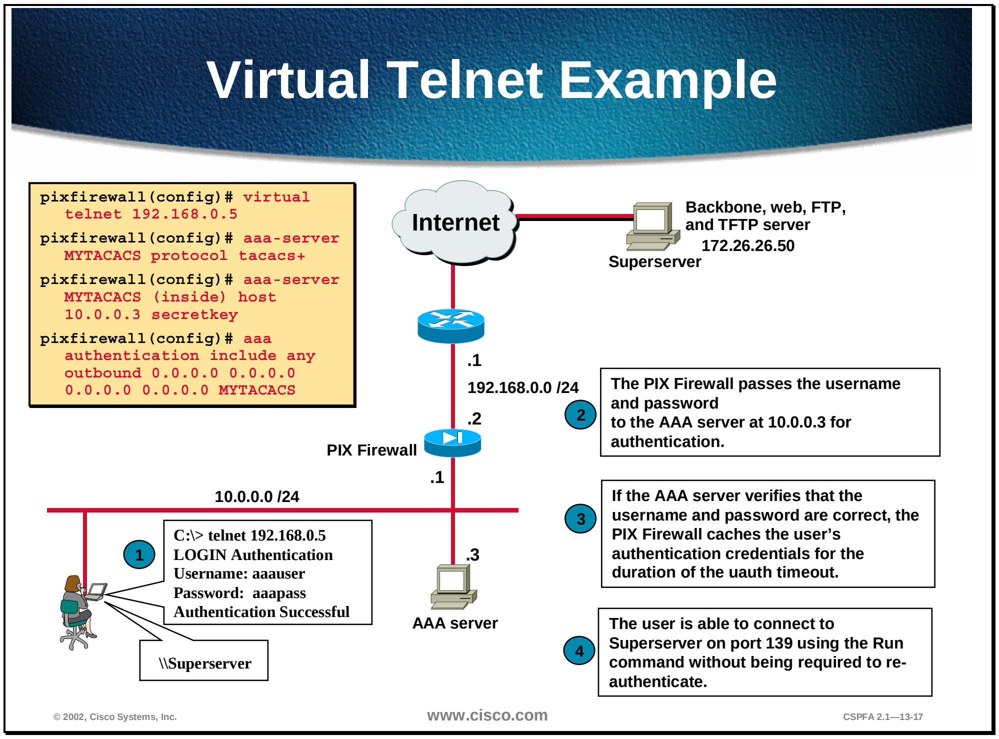
## Authentication of Non-Telnet, FTP, or HTTP Traffic

- Option 1—Authenticate first by accessing a Telnet, FTP, or HTTP server before accessing other services.
- Option 2—Authenticate to the PIX Firewall virtual Telnet service before accessing other services.

© 2002, Cisco Systems, Inc. www.cisco.com CSPFA 2.1—13-16

The PIX Firewall authenticates users via Telnet, FTP, or HTTP. But what if users need to access a Microsoft file server on port 139 or a Cisco IP/TV server for instance? How will they be authenticated? Whenever users are required to authenticate to access services other than Telnet, FTP, or HTTP, they need to do one of the following:

- Option 1—Authenticate first by accessing a Telnet, FTP, or HTTP server before accessing other services.
- Option 2—Authenticate to the PIX Firewall virtual Telnet service before accessing other services. When there are no Telnet, FTP, or HTTP servers with which to authenticate, or just to simplify authentication for the user, the PIX Firewall allows a virtual Telnet authentication option. This permits the user to authenticate directly with the PIX Firewall using the virtual Telnet IP address.



## Virtual Telnet Example

```
pixfirewall(config)# virtual
   telnet 192.168.0.5
pixfirewall(config)# aaa-server
   MYTACACS protocol tacacs+
pixfirewall(config)# aaa-server
   MYTACACS (inside) host
   10.0.0.3 secretkey
pixfirewall(config)# aaa
   authentication include any
   outbound 0.0.0.0 0.0.0.0
   0.0.0.0 0.0.0.0 MYTACACS
```

```
C:\> telnet 192.168.0.5
LOGIN Authentication
Username: aaauser
Password:  aaapass
Authentication Successful
```

```
\\Superserver
```

1

2 The PIX Firewall passes the username and password to the AAA server at 10.0.0.3 for authentication.

3 If the AAA server verifies that the username and password are correct, the PIX Firewall caches the user’s authentication credentials for the duration of the uauth timeout.

4 The user is able to connect to Superserver on port 139 using the Run command without being required to re-authenticate.

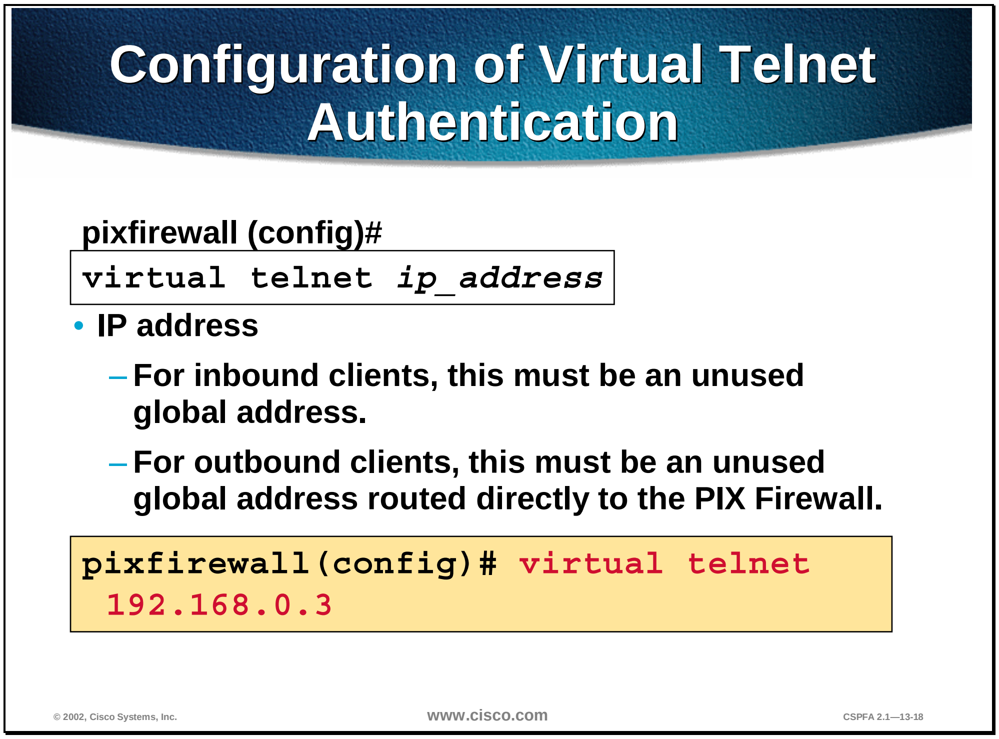
www.cisco.com CSPFA 2.1—13-17

The virtual Telnet option provides a way to pre-authenticate users who require connections through the PIX Firewall using services or protocols that do not support authentication. The virtual Telnet IP address is used both to authenticate in and authenticate out of the PIX Firewall.

When an unauthenticated user Telnets to the virtual IP address, the user is challenged for the username and password, and then authenticated with the TACACS+ or RADIUS server. Once authenticated, the user sees the message “Authentication Successful” and the authentication credentials are cached in the PIX Firewall for the duration of the user authentication (uauth) timeout.

If a user wishes to log out and clear the entry in the PIX Firewall uauth cache, the user can again Telnet to the virtual address. The user is prompted for a username and password, the PIX Firewall removes the associated credentials from the uauth cache, and the user receives a “Logout Successful” message.

In the previous figure, the user wants to establish a NetBIOS session on port 139 to access the file server named Superserver. The user telnets to the virtual Telnet address at 192.168.0.5, and is immediately challenged for a username and password before being authenticated with the TACACS+ AAA server. Once authenticated, the PIX Firewall allows the user to connect to the file server without re-authentication.



## Configuration of Virtual Telnet Authentication

pixfirewall (config)#

```
virtual telnet ip_address
```

- IP address
  - For inbound clients, this must be an unused global address.
  - For outbound clients, this must be an unused global address routed directly to the PIX Firewall.

```
pixfirewall(config)# virtual telnet
 192.168.0.3
```

www.cisco.com

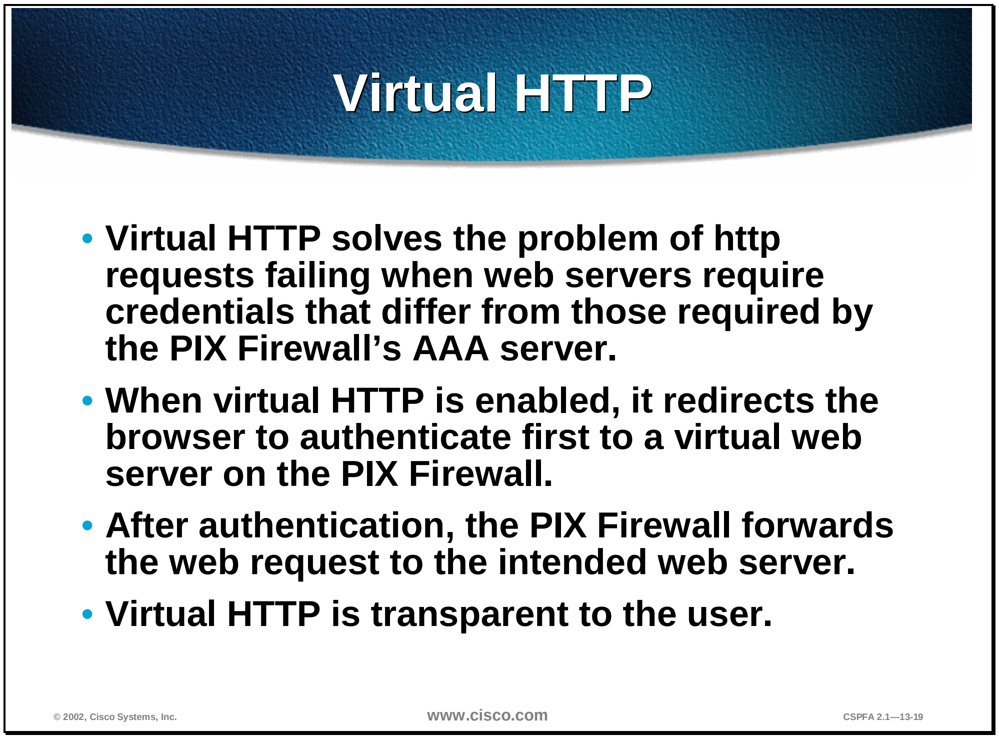
CSPFA 2.1—13-18

When using virtual Telnet to authenticate inbound clients, the IP address must be an unused global address. When using virtual Telnet to authenticate outbound clients, the IP address must be an unused global address routed directly to the PIX Firewall.

The syntax for the **virtual telnet** command is as follows:

**virtual telnet *ip_address***

| | |
|---|---|
| ***ip_address*** | Unused global IP address on PIX Firewall, used for Telnet for authentication. |



## Virtual HTTP

- Virtual HTTP solves the problem of http requests failing when web servers require credentials that differ from those required by the PIX Firewall’s AAA server.
- When virtual HTTP is enabled, it redirects the browser to authenticate first to a virtual web server on the PIX Firewall.
- After authentication, the PIX Firewall forwards the web request to the intended web server.
- Virtual HTTP is transparent to the user.

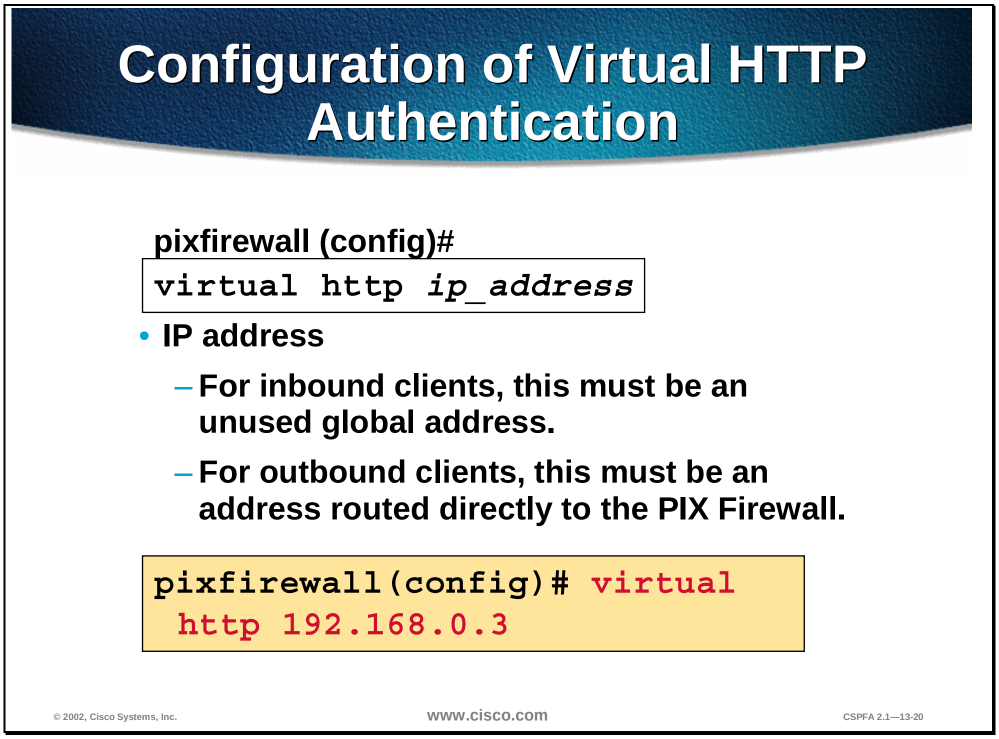
www.cisco.com
CSPFA 2.1—13-19

With the virtual HTTP option, web browsers work correctly with the PIX Firewall’s HTTP authentication. The PIX Firewall assumes that the AAA server database is shared with a web server and automatically provides the AAA server and web server with the same information. The virtual HTTP option works with the PIX Firewall to authenticate the user, separate the AAA server information from the web client’s URL request, and direct the web client to the web server. The virtual HTTP option works by redirecting the web browser’s initial connection to an IP address, which resides in the PIX Firewall, authenticating the user, then redirecting the browser back to the URL that the user originally requested. This option is so named because it accesses a virtual HTTP server on the PIX Firewall, which in reality does not exist.

This option is especially useful for PIX Firewall interoperability with Microsoft Internet Information Server (IIS), but is useful for other authentication servers. When using HTTP authentication to a site running Microsoft IIS that has “Basic text authentication” or “NT Challenge” enabled, users may be denied access from the Microsoft IIS server because the browser appends the string: “Authorization: Basic=Uuhjksdkfhk==” to the HTTP GET commands. This string contains the PIX Firewall authentication credentials. Windows NT Microsoft IIS servers respond to the credentials and assume that a Windows NT user is trying to access privileged pages on the server. Unless the PIX Firewall username and password combination is exactly the same as a valid Windows NT username and password combination on the Microsoft IIS server, the HTTP GET command is denied.

To solve this problem, the PIX Firewall redirects the browser’s initial connection to its virtual HTTP IP address, authenticates the user, and then redirects the browser back to the URL that the user originally requested. Virtual HTTP is transparent to the user; therefore, users enter actual destination URLs in their browsers as they normally would.

**Note** Do not set the timeout uauth duration to 0 seconds when using the virtual HTTP option. This will prevent HTTP connections to the real web server.



## Configuration of Virtual HTTP Authentication

pixfirewall (config)#

```
virtual http ip_address
```

- IP address
  - For inbound clients, this must be an unused global address.
  - For outbound clients, this must be an address routed directly to the PIX Firewall.

```
pixfirewall(config)# virtual
 http 192.168.0.3
```

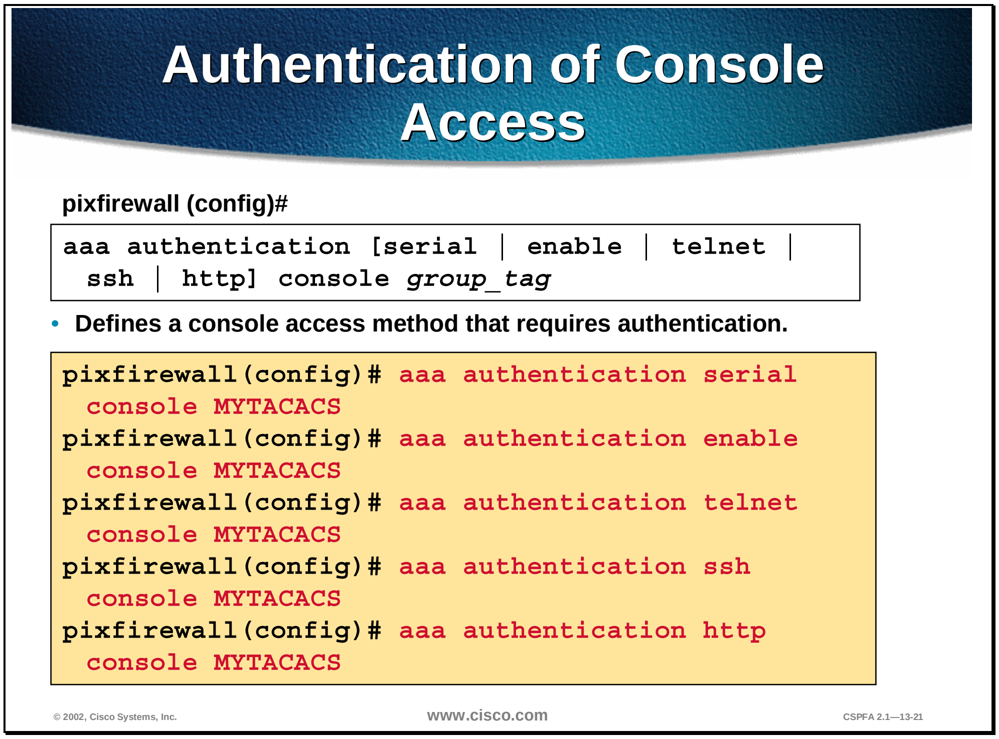
© 2002, Cisco Systems, Inc. www.cisco.com CSPFA 2.1—13-20

The virtual address identifies the IP address of the virtual HTTP server on the PIX Firewall. For inbound use, **ip_ address** can be any unused global address. Access to this address must be provided by an **access-list** and **static** command pair. For outbound use, **ip_address** must be an address routed directly to the PIX Firewall. The syntax for the **virtual http** command is as follows:

**virtual http *ip_address* [warn]**

**no virtual http *ip_address***

| | |
|---|---|
| ***ip_address*** | The PIX Firewall's network interface IP address. |
| **warn** | Informs **virtual http** command users that the command was redirected. This option is only applicable for text-based browsers where the redirect cannot happen automatically. |



## Authentication of Console Access

pixfirewall (config)#

```
aaa authentication [serial | enable | telnet |
  ssh | http] console group_tag
```

- Defines a console access method that requires authentication.

```
pixfirewall(config)# aaa authentication serial
  console MYTACACS
pixfirewall(config)# aaa authentication enable
  console MYTACACS
pixfirewall(config)# aaa authentication telnet
  console MYTACACS
pixfirewall(config)# aaa authentication ssh
  console MYTACACS
pixfirewall(config)# aaa authentication http
  console MYTACACS
```

© 2002, Cisco Systems, Inc. www.cisco.com CSPFA 2.1—13-21

Use the **aaa authentication console** command to require authentication verification to access the PIX Firewall's console. Authenticated access to the PIX Firewall console has different types of prompts depending on the option you choose. While the enable and ssh options allow three tries before stopping with an access denied message, both the serial and telnet options cause you to be prompted continually until you have successfully logged in.

The serial option requests a username and password before the first command-line prompt on the serial console connection. The telnet option forces you to specify a username and password before the first command-line prompt of a Telnet console connection. The **ssh** option requests a username and password before the first command-line prompt on the SSH console connection. The enable option requests a username and password before accessing privileged mode for serial, Telnet, or SSH connections.

Telnet access to the PIX Firewall console is available from any internal interface (and from the outside interface with IPSec configured) and requires previous use of the **telnet** command. SSH access to the PIX Firewall console is available from any interface without IPSec configured and requires previous use of the **ssh** command.

Authentication of the serial console creates a potential deadlock situation if the authentication server requests are not answered and you need access to the console to attempt diagnosis. If the console login request times out, you can gain access to the PIX Firewall from the serial console by entering the PIX Firewall username and the enable password. The maximum password length for accessing the console is 16 characters.

The **aaa authentication** command also supports PIX Device Manager (PDM) authentication. By using the **aaa authentication http console** command, you can configure the PIX Firewall to require authentication before allowing PDM to access it. If an **aaa authentication http console** command statement is not defined, you can gain access to the PIX Firewall (via PDM) with no username and the PIX Firewall enable password (set with the **password** command). If the **aaa** command is defined but the HTTP authentication request times out, which implies the AAA server may be down or not available, you can gain access to the PIX Firewall using the username *pix* and the enable password.

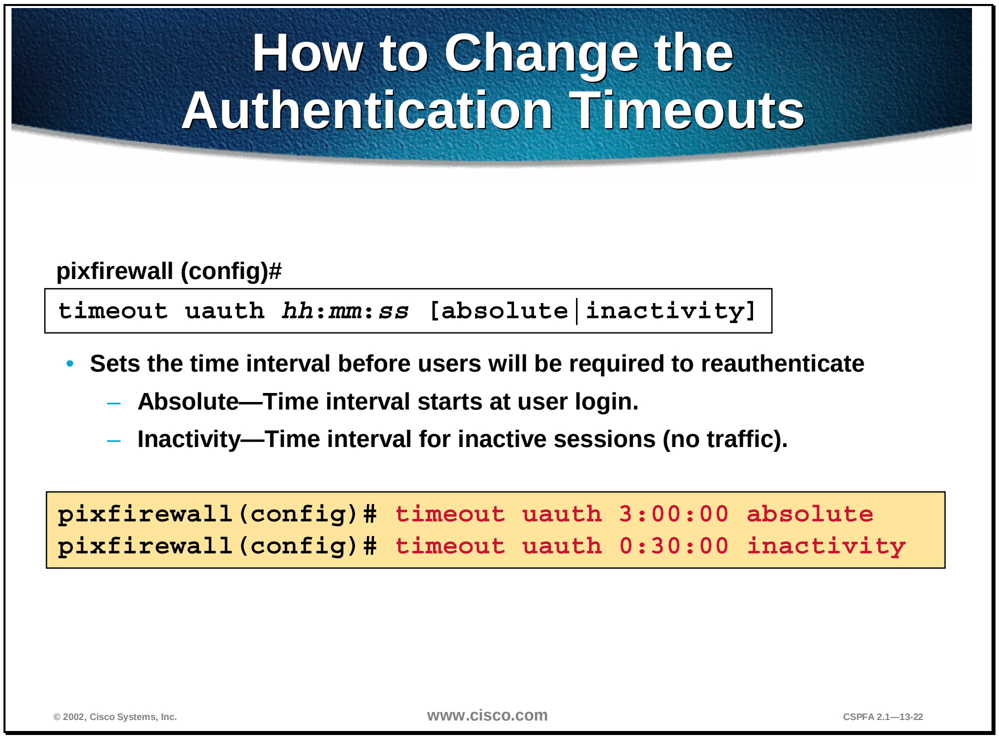
**Note** The serial console option also logs to a Syslog server changes made to the configuration from the serial console.

The syntax for the **aaa authentication console** command is as follows:

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

**no aaa authentication [serial | enable | telnet | ssh | http] console** ***group_tag***

| | |
|---|---|
| **serial** | Access verification for the PIX Firewall's serial console. |
| **enable** | Access verification for the PIX Firewall's privilege mode. |
| **telnet** | Access verification for the Telnet access to the PIX Firewall console. |
| **ssh** | Access verification for the SSH access to the PIX Firewall console. |
| **http** | Access verification for the HTTP access to the PIX Firewall (via PDM). |
| **console** | Specifies that access to the PIX Firewall console requires authentication. |
| ***group_tag*** | The group tag set with the **aaa-server** command. |



## How to Change the Authentication Timeouts

pixfirewall (config)#

```
timeout uauth hh:mm:ss [absolute|inactivity]
```

- Sets the time interval before users will be required to reauthenticate
  - Absolute—Time interval starts at user login.
  - Inactivity—Time interval for inactive sessions (no traffic).

```
pixfirewall(config)# timeout uauth 3:00:00 absolute
pixfirewall(config)# timeout uauth 0:30:00 inactivity
```

© 2002, Cisco Systems, Inc. www.cisco.com CSPFA 2.1—13-22

Use the **timeout uauth** command to specify how long the cache should be kept after the user connections become idle. The timeout command value must be at least 2 minutes. Use the **clear uauth** command to delete all authorization caches for all users, which will cause them to reauthenticate the next time they create a connection.

The inactivity and absolute qualifiers cause users to reauthenticate after either a period of inactivity or an absolute duration. The inactivity timer starts after a connection becomes idle. If a user establishes a new connection before the duration of the inactivity timer, the user is not required to reauthenticate. If a user establishes a new connection after the inactivity timer expires, the user must reauthenticate.

The absolute timer runs continuously, but waits to reprompt the user when the user starts a new connection, such as clicking a link after the absolute timer has elapsed. The user is then prompted to reauthenticate. The absolute timer must be shorter than the xlate timer, otherwise a user could be reprompted after their session already ended.

The inactivity timer gives users the best Internet access because they are not prompted to regularly reauthenticate. Absolute timers provide security and manage the PIX Firewall connections better. By being prompted to reauthenticate regularly, users manage their use of the resources more efficiently. Also by being reprompted, you minimize the risk that someone will attempt to use another user’s access after they leave their workstation, such as in a college computer lab. You may want to set an absolute timer during peak hours and an inactivity timer during other times.

Both an inactivity timer and an absolute timer can operate at the same time, but you should set the absolute timer duration for a longer time than the inactivity timer. If the absolute timer is set less than the inactivity timer, the inactivity timer never occurs. For example, if you set the absolute timer to 10 minutes and the inactivity timer to an hour, the absolute timer reprompts the user every 10 minutes, and the inactivity timer will never be started.

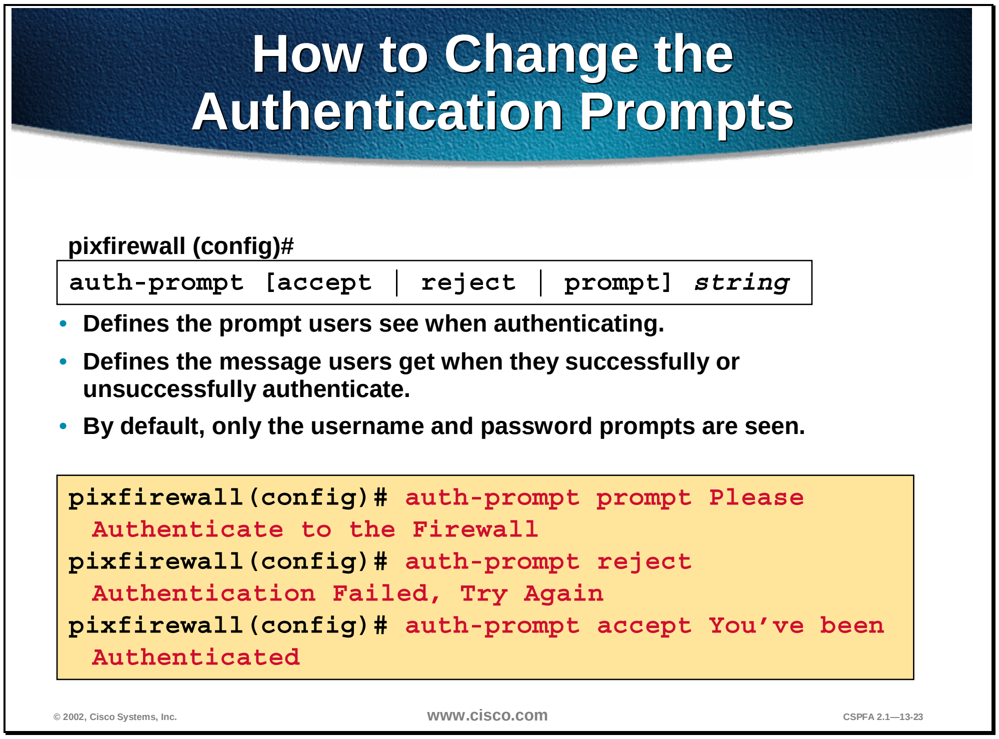
If you set the inactivity timer to some duration, but the absolute timer to zero, users are only reauthenticated after the inactivity timer elapses. If you set both timers to zero, users have to reauthenticate on every new connection.

**Note** Do not set the timeout uauth duration to 0 seconds when using the virtual HTTP option or passive FTP.

The syntax for the **timeout uauth** command is as follows:

```
timeout uauth hh:mm:ss [absolute | inactivity]

show timeout

clear uauth
```

| | |
|---|---|
| **uauth *hh:mm:ss*** | Duration before the authentication and authorization cache times out and the user has to reauthenticate the next connection. This duration must be shorter than the xlate values. Set to 0 to disable caching. |
| **absolute** | Runs the uauth timer continuously, but after timer elapses, waits to reprompt the user until the user starts a new connection (for example, clicking a link in a web browser). To disable absolute, set it to zero (0). The default is 5 minutes. |
| **inactivity** | Starts the uauth timer after a connection becomes idle. The default is 0. |



## How to Change the Authentication Prompts

pixfirewall (config)#

```
auth-prompt [accept | reject | prompt] string
```

- Defines the prompt users see when authenticating.
- Defines the message users get when they successfully or unsuccessfully authenticate.
- By default, only the username and password prompts are seen.

```
pixfirewall(config)# auth-prompt prompt Please
  Authenticate to the Firewall
pixfirewall(config)# auth-prompt reject
  Authentication Failed, Try Again
pixfirewall(config)# auth-prompt accept You've been
  Authenticated
```

© 2002, Cisco Systems, Inc. www.cisco.com CSPFA 2.1—13-23

Use the **auth-prompt** command to change the AAA challenge text for HTTP, FTP, and Telnet access. This is text that appears above the username and password prompts that you view when logging in.

**Note** Microsoft Internet Explorer only displays up to 37 characters in an authentication prompt, Netscape Navigator displays up to 120 characters, and Telnet and FTP display up to 235 characters in an authentication prompt.

The syntax for the **auth-prompt** command is as follows:

**auth-prompt [accept | reject | prompt]** ***string***

**no auth-prompt [accept | reject | prompt]** ***string***

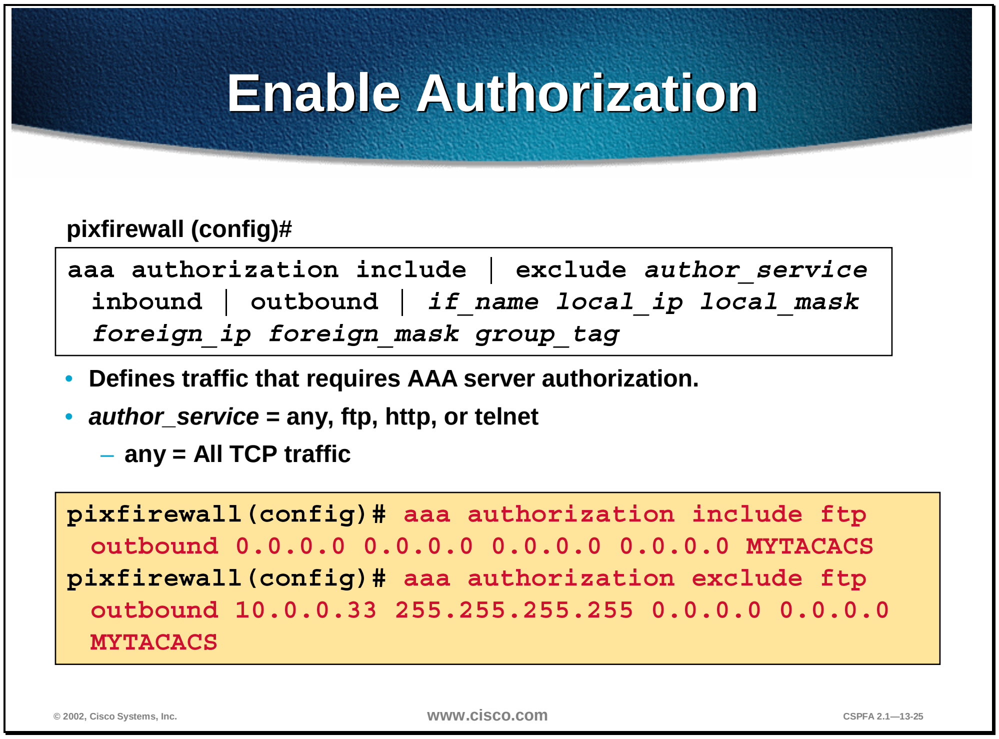
**show auth-prompt**

**clear auth-prompt**

| | |
|---|---|
| **accept** | If a user authentication via Telnet is accepted, the accept message is displayed. |
| **reject** | If a user authentication via Telnet is rejected, the reject message is displayed. |
| **prompt** | The AAA challenge prompt string follows this keyword. This keyword is optional for backward compatibility. |
| ***string*** | A string of up to 235 alphanumeric characters. Special characters should not be used; however, spaces and punctuation characters are permitted. Entering a question mark or pressing the Enter key ends the string. (The question mark appears in the string.) |

# Authorization Configuration

This section discusses the configuration of the PIX Firewall for authorization.



## Enable Authorization

pixfirewall (config)#

```
aaa authorization include | exclude author_service
  inbound | outbound | if_name local_ip local_mask
  foreign_ip foreign_mask group_tag
```

- Defines traffic that requires AAA server authorization.
- *author_service* = any, ftp, http, or telnet
  - any = All TCP traffic

```
pixfirewall(config)# aaa authorization include ftp
  outbound 0.0.0.0 0.0.0.0 0.0.0.0 0.0.0.0 MYTACACS
pixfirewall(config)# aaa authorization exclude ftp
  outbound 10.0.0.33 255.255.255.255 0.0.0.0 0.0.0.0
  MYTACACS
```

© 2002, Cisco Systems, Inc. www.cisco.com CSPFA 2.1—13-25

The PIX Firewall uses authorization services with TACACS+ AAA servers that determine which services an authenticated user can access.

**Note** The PIX Firewall does not support RADIUS authorization.

The syntax for the **aaa authorization** command is as follows:

**aaa authorization include | exclude** ***author_service*** **inbound | outbound |** ***if_name local_ip local_mask foreign_ip foreign_mask group_tag***

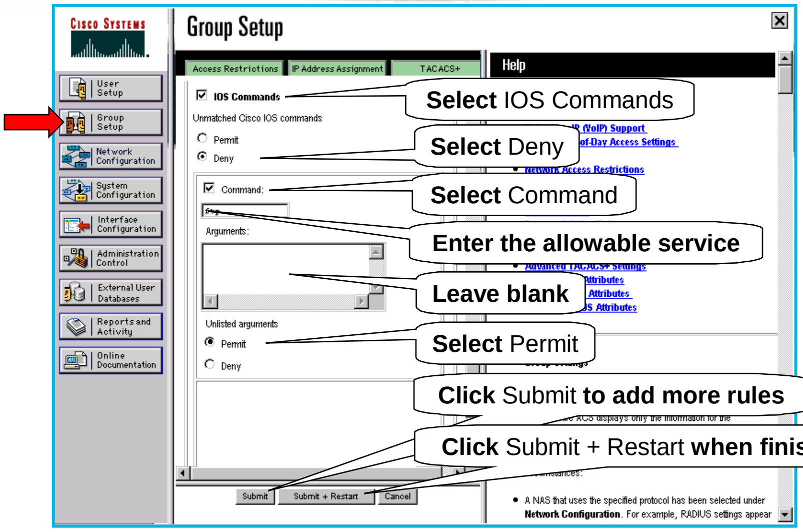
**no aaa authorization include | exclude** ***author_service*** **inbound | outbound |** ***if_name local_ip local_mask foreign_ip foreign_mask group_tag***

**clear aaa authorization [include | exclude** ***author_service*** **inbound | outbound |** ***if_name local_ip local_mask foreign_ip foreign_mask group_tag*****]**

| | |
|---|---|
| **include *author_service*** | The services that require authorization. Use any, ftp, http, or telnet. Services not specified are authorized implicitly. Services specified in the **aaa authentication** command do not affect the services, which require authorization. |

| | |
|---|---|
| **exclude *author_service*** | Creates an exception to a previously stated rule by excluding the specified service from authorization to the specified host. The exclude parameter improves the former except option by allowing the user to specify a port to exclude for a specific host or hosts. |
| **inbound** | Authenticates or authorizes inbound connections. Inbound means the connection originates on the outside interface and is being directed to the inside or any other perimeter interface. |
| **outbound** | Authenticates or authorizes outbound connections. Outbound means the connection originates on the inside and is being directed to the outside or any other perimeter interface. |
| ***if_name*** | Interface name from which users require authentication. Use if_name in combination with the local_ip address and the foreign_ip address to determine where access is sought and from whom. |
| ***local_ip*** | The IP address of the host or network of hosts that you want to be authenticated or authorized. You can set this address to 0 to mean all hosts and to let the authentication server decide which hosts are authenticated. |
| ***local_mask*** | Network mask of local_ip. Always specify a specific mask value. Use 0 if the IP address is 0. Use 255.255.255.255 for a host. |
| ***foreign_ip*** | The IP address of the hosts you want to access the local_ip address. Use 0 to mean all hosts. |
| ***foreign_mask*** | Network mask of foreign_ip. Always specify a specific mask value. Use 0 if the IP address is 0. Use 255.255.255.255 for a host. |
| ***group_tag*** | The group tag set with the **aaa-server** command. |

# Authorization Rules Allowing Specific Services



☑ IOS Commands

Unmatched Cisco IOS commands

○ Permit

● Deny

☑ Command: ftp

Arguments:

Unlisted arguments

● Permit

○ Deny

Select IOS Commands

Select Deny

Select Command

Enter the allowable service

Leave blank

Select Permit

Click Submit to add more rules

Click Submit + Restart when finished

www.cisco.com

Complete the following steps to add authorization rules for specific services in CSACS:

**Step 1** Click **Group Setup** from the navigation bar. The Group Setup window opens.

**Step 2** Scroll down in Group Setup until you find IOS Commands, and select the **IOS Commands** check box.

**Step 3** Select **Deny**, which is found under Unmatched Cisco IOS commands.

**Step 4** Select the **Command** check box.

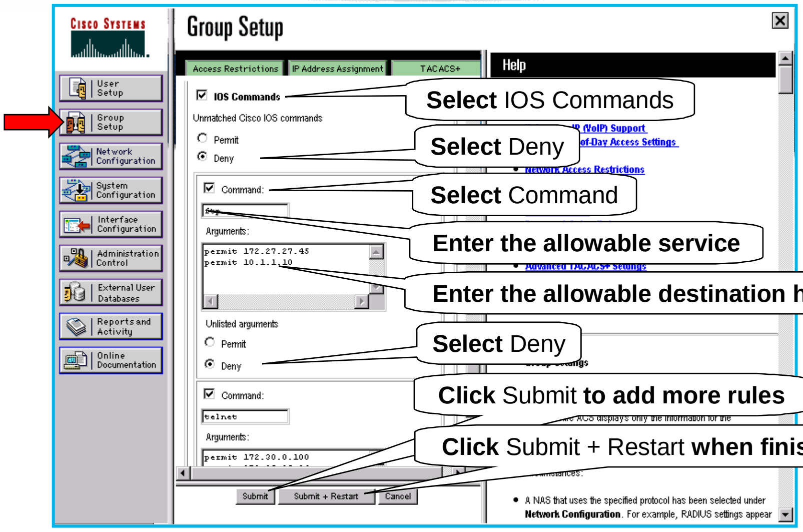
**Step 5** In the command field, enter one of the following allowable services: **ftp**, **telnet**, or **http**.

**Step 6** Leave the Arguments field blank.

**Step 7** Select **Permit**, which is found under Unlisted arguments.

**Step 8** Click **Submit** to add more rules, or click **Submit + Restart** when finished.

## Authorization Rules Allowing Services Only to Specific Hosts



☑ IOS Commands

Unmatched Cisco IOS commands

○ Permit
● Deny

☑ Command: ftp

Arguments:

```
permit 172.27.27.45
permit 10.1.1.10
```

Unlisted arguments

○ Permit
● Deny

☑ Command: telnet

Arguments:

```
permit 172.30.0.100
```

Submit | Submit + Restart | Cancel

Select IOS Commands

Select Deny

Select Command

Enter the allowable service

Enter the allowable destination hosts

Select Deny

Click Submit to add more rules

Click Submit + Restart when finished

© 2002, Cisco Systems, Inc. www.cisco.com CSPFA 2.1—13-27

Complete the following steps to add authorization rules for services to specific hosts in CSACS:

**Step 1** Click **Group Setup** from the navigation bar. The Group Setup window opens.

**Step 2** Scroll down in Group Setup until you find IOS Commands and select **the IOS Command** check box.

**Step 3** Select **Deny**, which is found under Unmatched Cisco IOS commands, select Deny.

**Step 4** Select the **Command** check box.

**Step 5** In the command field, enter one of the following allowable services: ftp, telnet, or http.

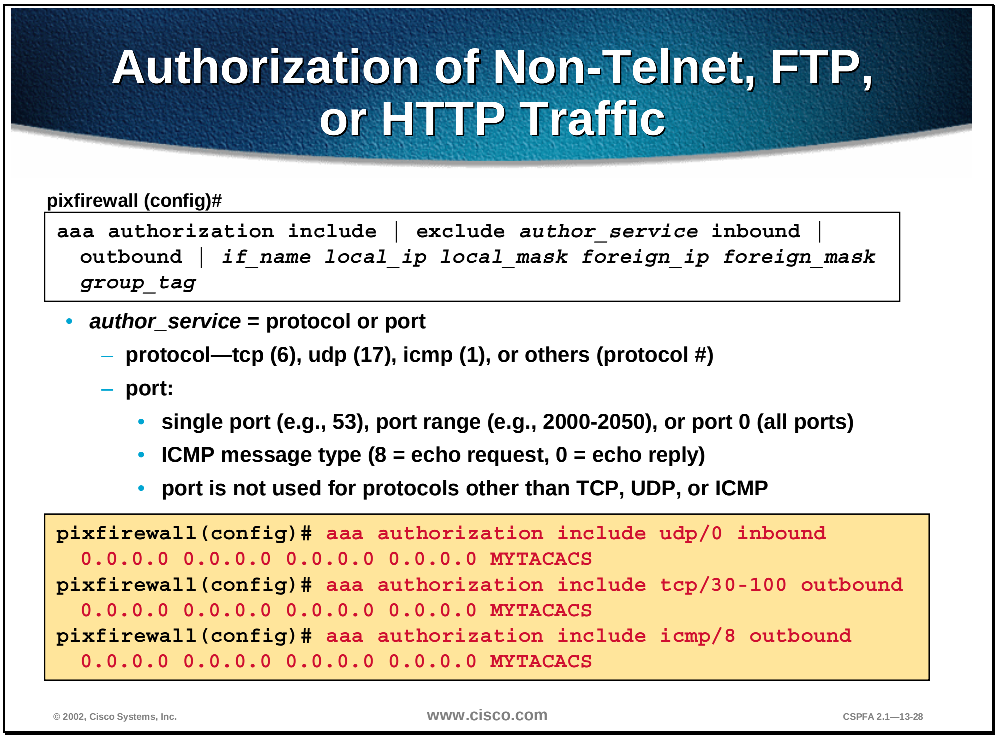
**Step 6** In the Arguments field, enter the IP addresses of the host that users are authorized to go to. Use the following format:

```
permit ip_addr
```

(where *ip_addr* = the IP address of the host)

**Step 7** Select **Deny**, which is found under Unlisted arguments.

**Step 8** Click **Submit** to add more rules, or click **Submit + Restart** when finished.



## Authorization of Non-Telnet, FTP, or HTTP Traffic

pixfirewall (config)#

```
aaa authorization include | exclude author_service inbound |
  outbound | if_name local_ip local_mask foreign_ip foreign_mask
  group_tag
```

- *author_service* = protocol or port
  - protocol—tcp (6), udp (17), icmp (1), or others (protocol #)
  - port:
    - single port (e.g., 53), port range (e.g., 2000-2050), or port 0 (all ports)
    - ICMP message type (8 = echo request, 0 = echo reply)
    - port is not used for protocols other than TCP, UDP, or ICMP

```
pixfirewall(config)# aaa authorization include udp/0 inbound
  0.0.0.0 0.0.0.0 0.0.0.0 0.0.0.0 MYTACACS
pixfirewall(config)# aaa authorization include tcp/30-100 outbound
  0.0.0.0 0.0.0.0 0.0.0.0 0.0.0.0 MYTACACS
pixfirewall(config)# aaa authorization include icmp/8 outbound
  0.0.0.0 0.0.0.0 0.0.0.0 0.0.0.0 MYTACACS
```

www.cisco.com CSPFA 2.1—13-28

The syntax of the **aaa authorization** of non-Telnet, FTP, or HTTP command is as follows:

**aaa authorization include | exclude *author_service* inbound | outbound | *if_name local_ip local_mask foreign_ip foreign_mask group_tag***

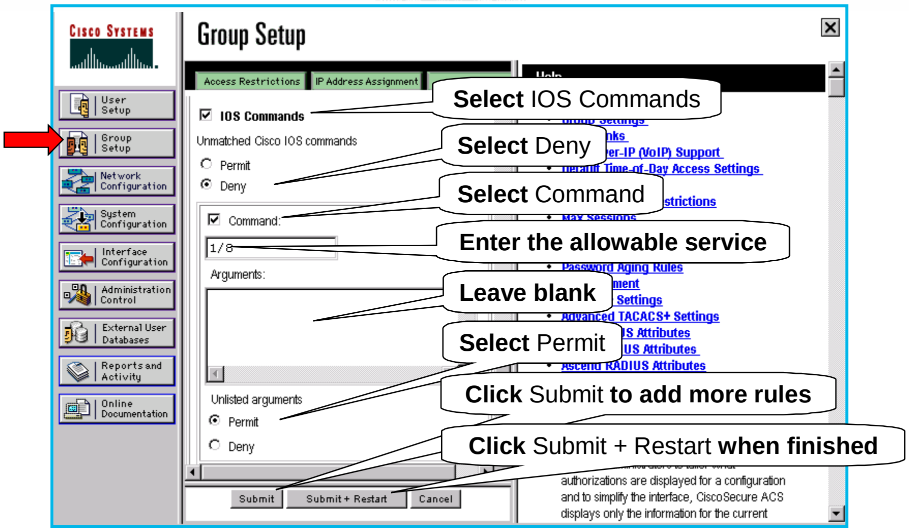
**no aaa authorization [include | exclude *author_service* inbound | outbound | *if_name local_ip local_mask foreign_ip foreign_mask group_tag***

**clear aaa [authorization [include | exclude *author_service* inbound | outbound | *if_name local_ip local_mask foreign_ip foreign_mask group_tag*]]**

| | |
|---|---|
| **include *author_service*** | The services that require authorization. Use a protocol or port number. Services not specified are authorized implicitly. Services specified in the **aaa authentication** command do not affect the services that require authorization. |
| **exclude *author_service*** | Creates an exception to a previously stated rule by excluding the specified service from authorization to the specified host or networks. |
| **inbound** | Authenticates or authorizes inbound connections. Inbound means the connection originates on the outside interface and is being directed to the inside or any other perimeter interface. |
| **outbound** | Authenticates or authorizes outbound connections. Outbound means the connection originates on the inside and is being directed to the outside or any other perimeter interface. |

| | |
|---|---|
| ***if_name*** | Interface name from which users require authentication. Use if_name in combination with the local_ip address and the foreign_ip address to determine where access is sought and from whom. |
| ***local_ip*** | The IP address of the host or network of hosts that you want to be authenticated or authorized. You can set this address to 0 to mean all hosts and to let the authentication server decide which hosts are authenticated. |
| ***local_mask*** | Network mask of local_ip. Always specify a specific mask value. Use 0 if the IP address is 0. Use 255.255.255.255 for a host. |
| ***foreign_ip*** | The IP address of the hosts you want to access the local_ip address. Use 0 to mean all hosts. |
| ***foreign_mask*** | Network mask of foreign_ip. Always specify a specific mask value. Use 0 if the IP address is 0. Use 255.255.255.255 for a host. |
| ***group_tag*** | The group tag set with the **aaa-server** command. |

# Authorization of Non-Telnet, FTP, or HTTP Traffic on CSACS-NT



Group Setup

☑ IOS Commands

Unmatched Cisco IOS commands

○ Permit

● Deny

☑ Command: 1/8

Arguments:

Unlisted arguments

● Permit

○ Deny

Submit | Submit + Restart | Cancel

authorizations are displayed for a configuration and to simplify the interface, CiscoSecure ACS displays only the information for the current

Select IOS Commands

Select Deny

Select Command

Enter the allowable service

Leave blank

Select Permit

Click Submit to add more rules

Click Submit + Restart when finished

© 2002, Cisco Systems, Inc. www.cisco.com CSPFA 2.1—13-29

Complete the following steps to add authorization rules for specific non-telnet, FTP, or HTTP services in CSACS:

**Step 1** Click **Group Setup** from the navigation bar. The Group Setup window opens.

**Step 2** Scroll down in Group Setup until you find IOS Commands, and select the **IOS Command** check box.

**Step 3** Select **Deny**, which is found under Unmatched Cisco IOS commands.

**Step 4** Select the **Command** check box.

**Step 5** In the command field, enter an allowable service using the following format: ***protocol*** or ***port*** (where protocol is the protocol number and port is the port number).

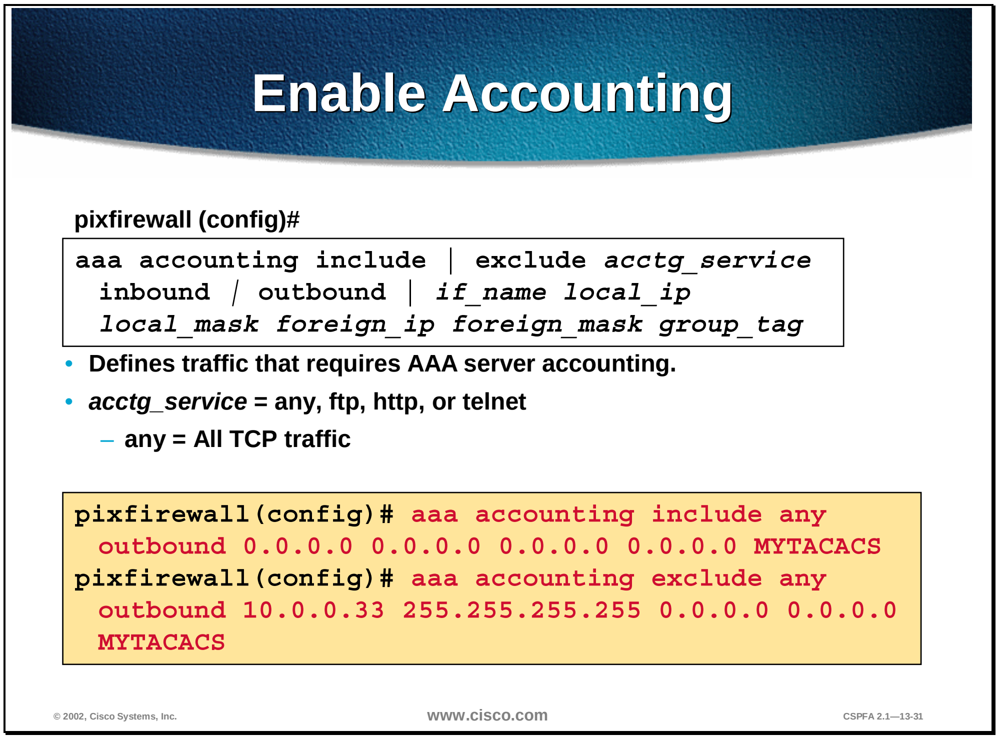
**Step 6** Leave the Arguments field blank.

**Step 7** Select **Permit**, which is found under Unlisted arguments.

**Step 8** Click **Submit** to add more rules, or click **Submit** + **Restart** when finished.

# Accounting Configuration

This section demonstrates how to enable and configure accounting for all services, select services, or no services.



## Enable Accounting

pixfirewall (config)#

```
aaa accounting include | exclude acctg_service
  inbound | outbound | if_name local_ip
  local_mask foreign_ip foreign_mask group_tag
```

- Defines traffic that requires AAA server accounting.
- *acctg_service* = any, ftp, http, or telnet
  - any = All TCP traffic

```
pixfirewall(config)# aaa accounting include any
  outbound 0.0.0.0 0.0.0.0 0.0.0.0 0.0.0.0 MYTACACS
pixfirewall(config)# aaa accounting exclude any
  outbound 10.0.0.33 255.255.255.255 0.0.0.0 0.0.0.0
  MYTACACS
```

© 2002, Cisco Systems, Inc. www.cisco.com CSPFA 2.1—13-31

The syntax for the **aaa accounting** command is as follows:

**aaa accounting include** | **exclude** ***acctg_service*** **inbound** | **outbound** | ***if_name local_ip local_mask foreign_ip foreign_mask group_tag***

**no aaa accounting include** | **exclude** ***authen_service*** **inbound** | **outbound** | ***if_name group_tag***

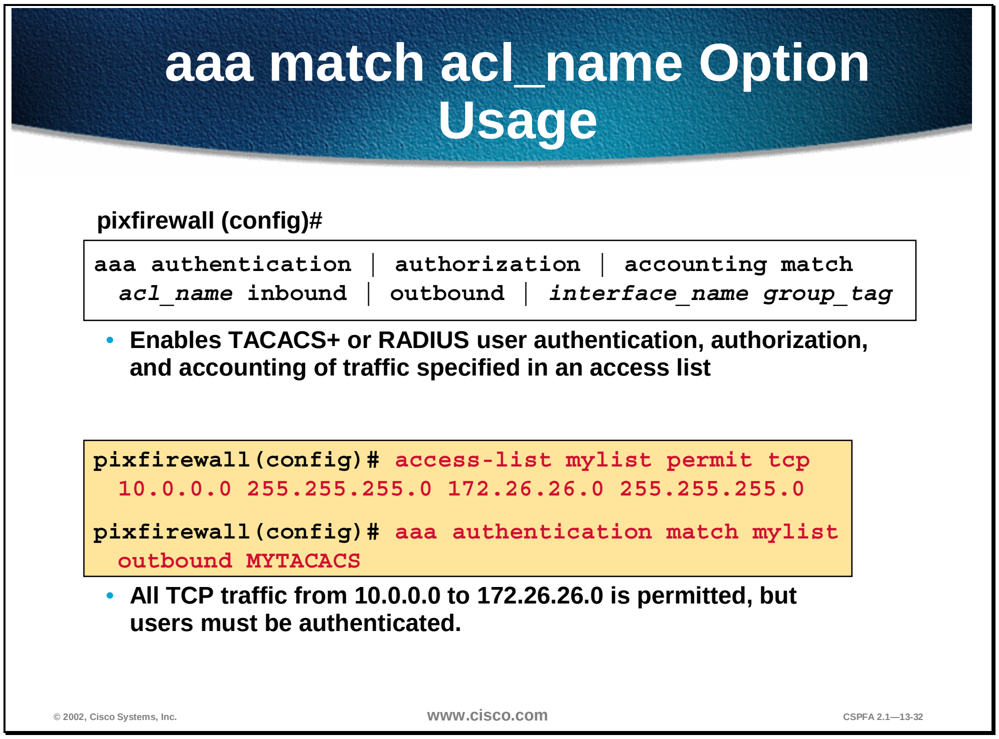
**clear aaa [accounting [include** | **exclude** ***authen_service*** **inbound** | **outbound** | ***if_name group_tag*]]**

| | |
|---|---|
| **include *acctg_service*** | The accounting service. Accounting is provided for all services, or you can limit it to one or more services. Possible values are any, ftp, http, telnet, or protocol/port. Use any to provide accounting for all TCP services. To provide accounting for UDP services, use the *protocol*/*port* form. |
| **exclude *acctg_service*** | Create an exception to a previously stated rule by excluding the specified service from authentication, authorization, or accounting to the specified host. The exclude parameter improves the former except option by allowing the user to specify a port to exclude to a specific host or hosts. |

| | |
|---|---|
| **inbound** | Authenticates or authorizes inbound connections. Inbound means the connection originates on the outside interface and is being directed to the inside or any other perimeter interface. |
| **outbound** | Authenticates or authorizes outbound connections. Outbound means the connection originates on the inside and is being directed to the outside or any other perimeter interface. |
| ***if_name*** | Interface name from which users require authentication. Use if_name in combination with the local_ip address and the foreign_ip_address to determine where access is sought and from whom. |
| ***local_ip*** | The IP address of the host or network of hosts that you want to be authenticated or authorized. You can set this address to 0 to mean all hosts and to let the authentication server decide which hosts are authenticated. |
| ***local_mask*** | Network mask of local_ip. Always specify a specific mask value. Use 0 if the IP address is 0. Use 255.255.255.255 for a host. |
| ***foreign_ip*** | The IP address of the hosts you want to access the local_ip address. Use 0 to mean all hosts. |
| ***foreign_mask*** | Network mask of foreign_ip. Always specify a specific mask value. Use 0 if the IP address is 0. Use 255.255.255.255 for a host. |
| ***group_tag*** | The group tag set with the **aaa-server** command. |

To specify the value of the acctg.service argument using the protocol/port form, enter the protocol as a number (6 for TCP, 17 for UDP, 1 for ICMP, and so on). The port is the TCP or UDP destination port. A port value of 0 (zero) means all ports. If the protocol specified is ICMP, the port is the ICMP type, such as 8 for ICMP echo and 0 for ICMP echo-reply. Examples of the **aaa accounting** command using protocol/port form follow:

- **aaa accounting include udp/53 inbound 0 0 0 0 MYTACACS**—Enables accounting for DNS lookups from the outside interface
- **aaa accounting include 1/0 outbound 0 0 0 0 MYTACACS**—Enables accounting of ICMP echo-reply packets arriving at the inside interface from inside hosts
- **aaa accounting include 1/8 outbound 0 0 0 0 MYTACACS**—Enables accounting only for ICMP echoes (pings) that arrive at the inside interface from an inside host



## aaa match acl_name Option Usage

pixfirewall (config)#

```
aaa authentication | authorization | accounting match
  acl_name inbound | outbound | interface_name group_tag
```

- Enables TACACS+ or RADIUS user authentication, authorization, and accounting of traffic specified in an access list

```
pixfirewall(config)# access-list mylist permit tcp
  10.0.0.0 255.255.255.0 172.26.26.0 255.255.255.0

pixfirewall(config)# aaa authentication match mylist
  outbound MYTACACS
```

- All TCP traffic from 10.0.0.0 to 172.26.26.0 is permitted, but users must be authenticated.

www.cisco.com CSPFA 2.1—13-32

In the PIX Firewall software versions 5.2 and higher, the match acl_name option is available in the **aaa** command. The **aaa** command can take part of its input from an access control list (ACL).

In the previous example, the acl mylist permits all TCP traffic from network 10.0.0.0 to network 172.26.26.0. The match acl_name option in the **aaa** command instructs the PIX Firewall to require authentication when the action the user is trying to perform matches the actions specified in **mylist**. Therefore, any time a user on the 10.0.0.0 internal network uses any TCP application to access network 172.26.26.0, he will be required to authenticate. In other words, the command **aaa authentication match mylist outbound MYTACACS** is equal to **aaa authentication include any outbound 10.0.0.0 255.255.255.0 172.26.26.0 255.255.255.0 MYTACACS**.

Traditional **aaa** command configuration and functionality continue to work as in previous versions and are not converted to the ACL format. Hybrid configurations, which are traditional configurations combined with the new ACL configurations, are not recommended.

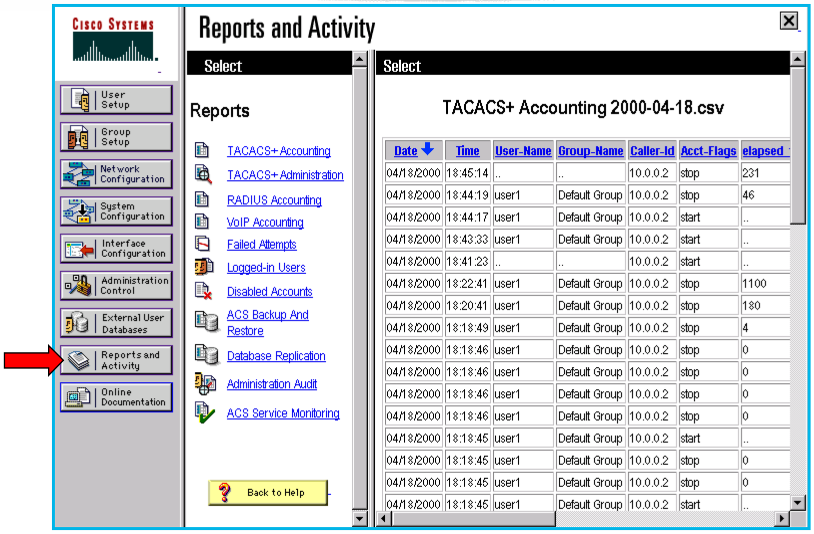
The syntax for the **aaa authentication | authorization | accounting** command is as follows:

**aaa authentication** | **authorization** | **accounting match** ***acl_name*** **inbound** | **outbound** | ***if_name*** ***group_tag***

| | |
|---|---|
| **match *acl_name*** | Specifies an access-list command statement name. |
| **inbound** | Authenticates or authorizes inbound connections. Inbound means the connection originates on the outside interface and is being directed to the inside interface. |
| **outbound** | Authenticates or authorizes outbound connections. Outbound means the connection originates on the inside and is |

| | |
|---|---|
| | being directed to the outside interface. |
| ***if_name*** | Interface name from which users require authentication. Use if_name in combination with the local_ip address and the foreign_ip address to determine where access is sought and from whom. |
| ***group_tag*** | The group tab set with the aaa-server command. |

## How to View Accounting Information in CSACS-NT



TACACS+ Accounting 2000-04-18.csv

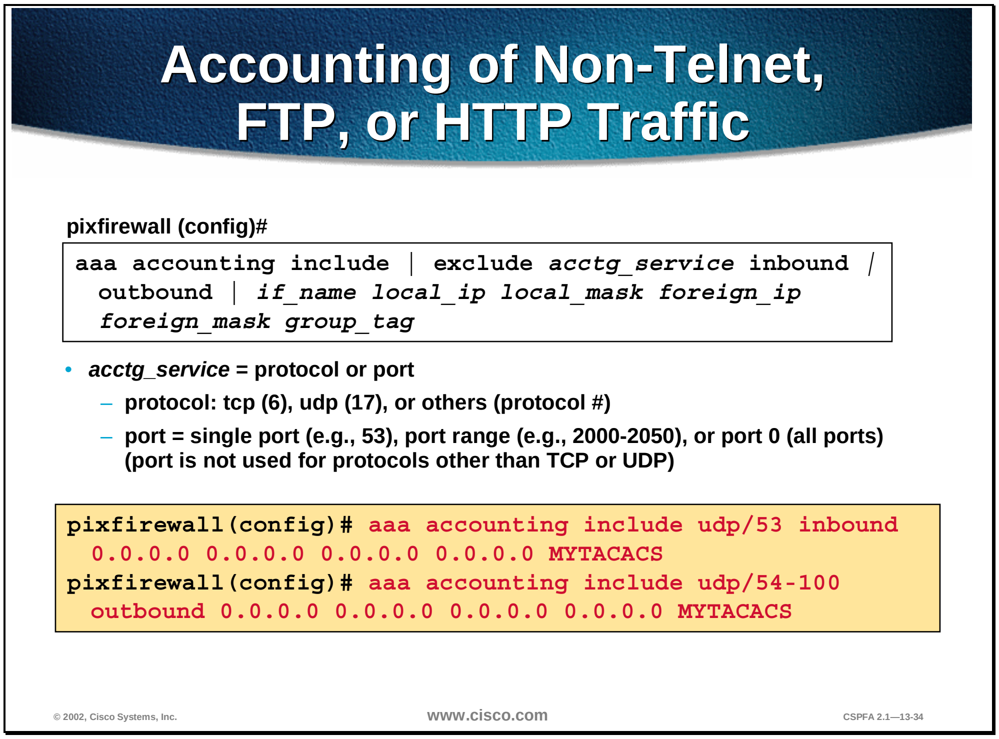
| Date | Time | User-Name | Group-Name | Caller-Id | Acct-Flags | elapsed |
|---|---|---|---|---|---|---|
| 04/18/2000 | 18:45:14 | .. | .. | 10.0.0.2 | stop | 231 |
| 04/18/2000 | 18:44:19 | user1 | Default Group | 10.0.0.2 | stop | 46 |
| 04/18/2000 | 18:44:17 | user1 | Default Group | 10.0.0.2 | start | .. |
| 04/18/2000 | 18:43:33 | user1 | Default Group | 10.0.0.2 | start | .. |
| 04/18/2000 | 18:41:23 | .. | .. | 10.0.0.2 | start | .. |
| 04/18/2000 | 18:22:41 | user1 | Default Group | 10.0.0.2 | stop | 1100 |
| 04/18/2000 | 18:20:41 | user1 | Default Group | 10.0.0.2 | stop | 180 |
| 04/18/2000 | 18:18:49 | user1 | Default Group | 10.0.0.2 | stop | 4 |
| 04/18/2000 | 18:18:46 | user1 | Default Group | 10.0.0.2 | stop | 0 |
| 04/18/2000 | 18:18:46 | user1 | Default Group | 10.0.0.2 | stop | 0 |
| 04/18/2000 | 18:18:46 | user1 | Default Group | 10.0.0.2 | stop | 0 |
| 04/18/2000 | 18:18:46 | user1 | Default Group | 10.0.0.2 | stop | 0 |
| 04/18/2000 | 18:18:45 | user1 | Default Group | 10.0.0.2 | start | .. |
| 04/18/2000 | 18:18:45 | user1 | Default Group | 10.0.0.2 | stop | 0 |
| 04/18/2000 | 18:18:45 | user1 | Default Group | 10.0.0.2 | stop | 0 |
| 04/18/2000 | 18:18:45 | user1 | Default Group | 10.0.0.2 | start | .. |

© 2002, Cisco Systems, Inc. www.cisco.com CSPFA 2.1—13-33

Complete the following steps to add authorization rules for specific non-telnet, FTP, or HTTP services in CSACS:

**Step 1** Click **Reports and Activity** from the navigation bar. The Report and Activity window opens.

**Step 2** Click TACACS+ Accounting from the Reports to display the accounting records.



## Accounting of Non-Telnet, FTP, or HTTP Traffic

pixfirewall (config)#

```
aaa accounting include | exclude acctg_service inbound |
  outbound | if_name local_ip local_mask foreign_ip
  foreign_mask group_tag
```

- *acctg_service* = protocol or port
  - protocol: tcp (6), udp (17), or others (protocol #)
  - port = single port (e.g., 53), port range (e.g., 2000-2050), or port 0 (all ports) (port is not used for protocols other than TCP or UDP)

```
pixfirewall(config)# aaa accounting include udp/53 inbound
  0.0.0.0 0.0.0.0 0.0.0.0 0.0.0.0 MYTACACS
pixfirewall(config)# aaa accounting include udp/54-100
  outbound 0.0.0.0 0.0.0.0 0.0.0.0 0.0.0.0 MYTACACS
```

© 2002, Cisco Systems, Inc. www.cisco.com CSPFA 2.1—13-34

The syntax for the **aaa accounting** of non-Telnet, FTP, or HTTP traffic command is as follows:

**aaa accounting include | exclude** ***acctg_service*** **inbound | outbound |** ***if_name local_ip local_mask foreign_ip foreign_mask group_tag***

**no aaa accounting include | exclude** ***authen_service*** **inbound | outbound |** ***if_name group_tag***

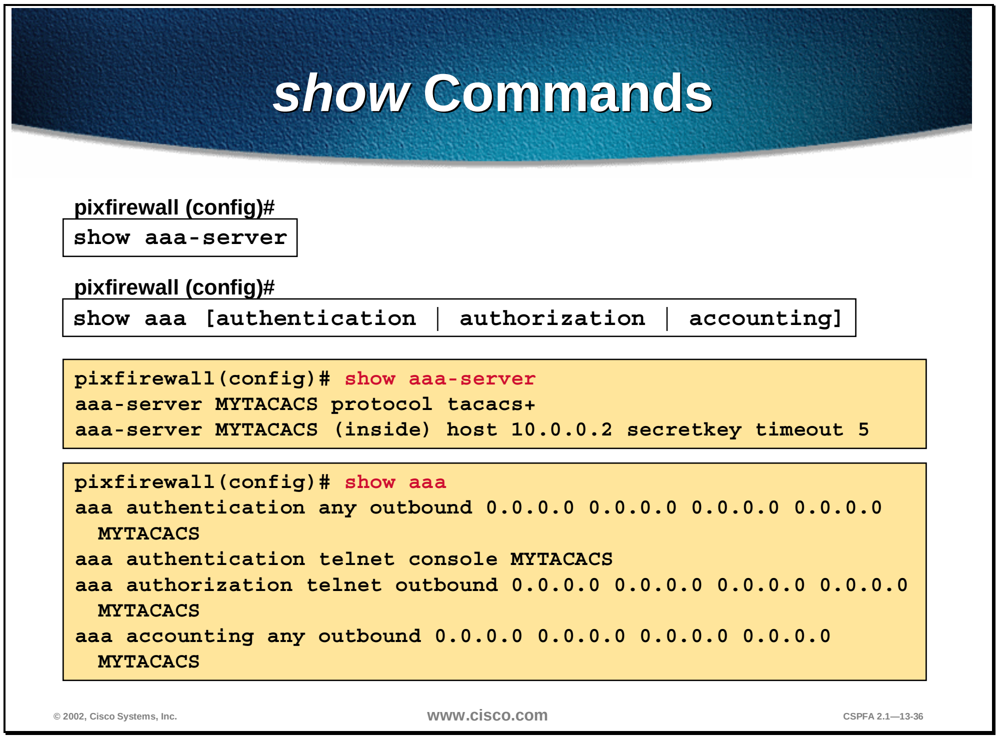
**clear aaa [accounting [include | exclude** ***authen_service*** **inbound | outbound |** ***if_name group_tag*****]]**

| | |
|---|---|
| **include *acctg_service*** | The accounting service. Accounting is provided for all services, or you can limit it to one or more services. Possible values are any, ftp, http, or telnet. Use any to provide accounting for all TCP services. To provide accounting for UDP services, use the protocol/port form. |
| **exclude *acctg_service*** | Creates an exception to a previously stated rule by excluding the specified service from authentication, authorization, or accounting to the specified host. The exclude parameter improves the former except option by enabling the user to specify a port to exclude to a specific host or hosts. |
| **inbound** | Authenticates or authorizes inbound connections. Inbound means the connection originates on the outside interface and is being directed to the inside or any other perimeter interface. |
| **outbound** | Authenticates or authorizes outbound connections. Outbound means the connection originates on the inside and is being directed to the outside or any other perimeter interface. |

| | |
|---|---|
| ***if_name*** | Interface name from which users require authentication. Use if_name in combination with the local_ip address and the foreign_ip address to determine where access is sought and from whom. |
| ***local_ip*** | The IP address of the host or network of hosts that you want to be authenticated or authorized. You can set this address to 0 to mean all hosts and to let the authentication server decide which hosts are authenticated. |
| ***local_mask*** | Network mask of local_ip. Always specify a specific mask value. Use 0 if the IP address is 0. Use 255.255.255.255 for a host. |
| ***foreign_ip*** | The IP address of the hosts you want to access the local_ip address. Use 0 to mean all hosts. |
| ***foreign_mask*** | Network mask of foreign_ip. Always specify a specific mask value. Use 0 if the IP address is 0. Use 255.255.255.255 for a host. |
| ***group_tag*** | The group tag set with the **aaa-server** command. |

# Troubleshooting the AAA Configuration

This section discusses the procedure for verifying the authentication, authorization, and accounting (AAA) configuration.



## *show* Commands

pixfirewall (config)#

```
show aaa-server
```

pixfirewall (config)#

```
show aaa [authentication | authorization | accounting]
```

```
pixfirewall(config)# show aaa-server
aaa-server MYTACACS protocol tacacs+
aaa-server MYTACACS (inside) host 10.0.0.2 secretkey timeout 5
```

```
pixfirewall(config)# show aaa
aaa authentication any outbound 0.0.0.0 0.0.0.0 0.0.0.0 0.0.0.0
  MYTACACS
aaa authentication telnet console MYTACACS
aaa authorization telnet outbound 0.0.0.0 0.0.0.0 0.0.0.0 0.0.0.0
  MYTACACS
aaa accounting any outbound 0.0.0.0 0.0.0.0 0.0.0.0 0.0.0.0
  MYTACACS
```

© 2002, Cisco Systems, Inc. www.cisco.com CSPFA 2.1—13-36

The syntax for the **show aaa-server** and **show aaa** commands are as follows:

**show aaa-server**

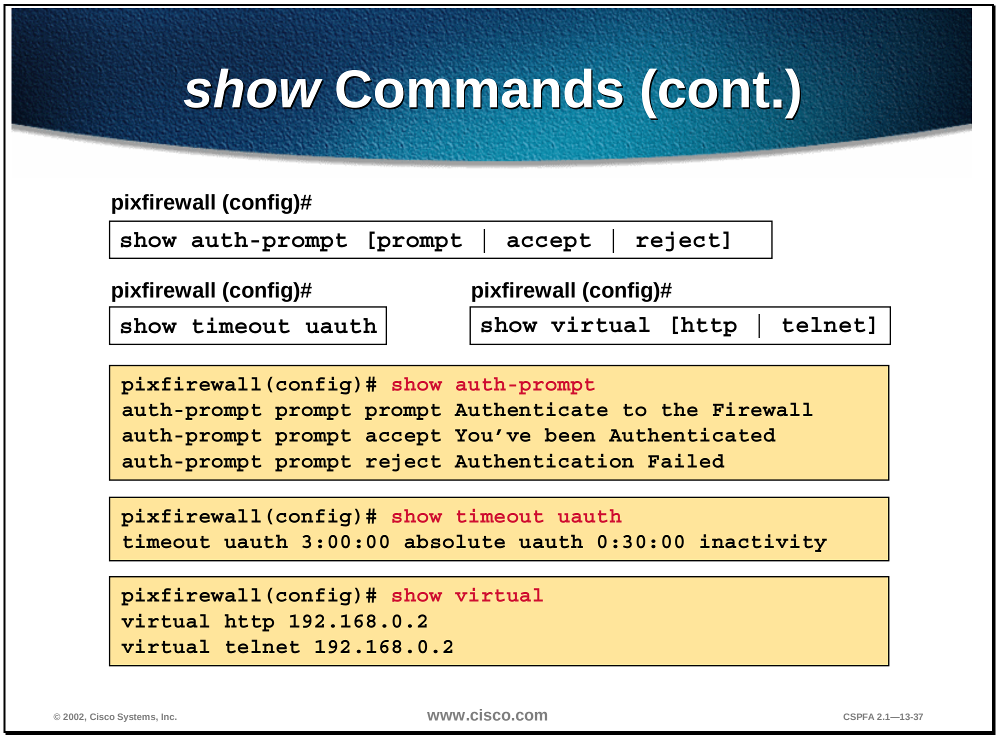
**clear aaa-server [group_tag]**

**no aaa-server** ***group_tag*** **(*if_name*) host** ***server_ip key*** **timeout** ***seconds***

**show aaa [authentication** | **authorization** | **accounting]**

| | |
|---|---|
| ***group tag*** | An alphanumeric string that is the name of the server group. |
| ***if_name*** | The interface name on which the server resides. |
| **host *server_ip*** | The IP address of the TACACS+ or RADIUS server. |
| ***key*** | A case-sensitive, alphanumeric keyword of up to 127 characters that is the same value as the key on the TACACS+ server. Any characters entered past 127 are ignored. The key is used between the client and server for encrypting data between them. The key must be the same on both the client and server systems. Spaces are not permitted in the key, but other special characters are. |

| | |
|---|---|
| **timeout *seconds*** | A retransmit timer that specifies the duration that the PIX Firewall retries access. Access to the AAA server is retried four times before choosing the next AAA server. The default is 5 seconds. The maximum time is 30 seconds. |
| **authentication** | Displays user authentication, prompts user for username and password, and verifies information with the authentication server. |
| **authorization** | Displays TACACS+ user authorization for services. (The PIX Firewall does not support RADIUS authorization.) The authentication server determines what services the user is authorized to access. |
| **accounting** | Displays accounting services with the authentication server. Use of this command requires that you previously used the **aaa-server** command to designate an authentication server. |



## *show* Commands (cont.)

pixfirewall (config)#

```
show auth-prompt [prompt | accept | reject]
```

pixfirewall (config)#

```
show timeout uauth
```

pixfirewall (config)#

```
show virtual [http | telnet]
```

```
pixfirewall(config)# show auth-prompt
auth-prompt prompt prompt Authenticate to the Firewall
auth-prompt prompt accept You've been Authenticated
auth-prompt prompt reject Authentication Failed
```

```
pixfirewall(config)# show timeout uauth
timeout uauth 3:00:00 absolute uauth 0:30:00 inactivity
```

```
pixfirewall(config)# show virtual
virtual http 192.168.0.2
virtual telnet 192.168.0.2
```

www.cisco.com CSPFA 2.1—13-37

The syntax for the **show auth-prompt**, **show timeout uauth**, and the **show virtual** commands are as follows:

**show auth-prompt** [**prompt** | **accept** | **reject**]

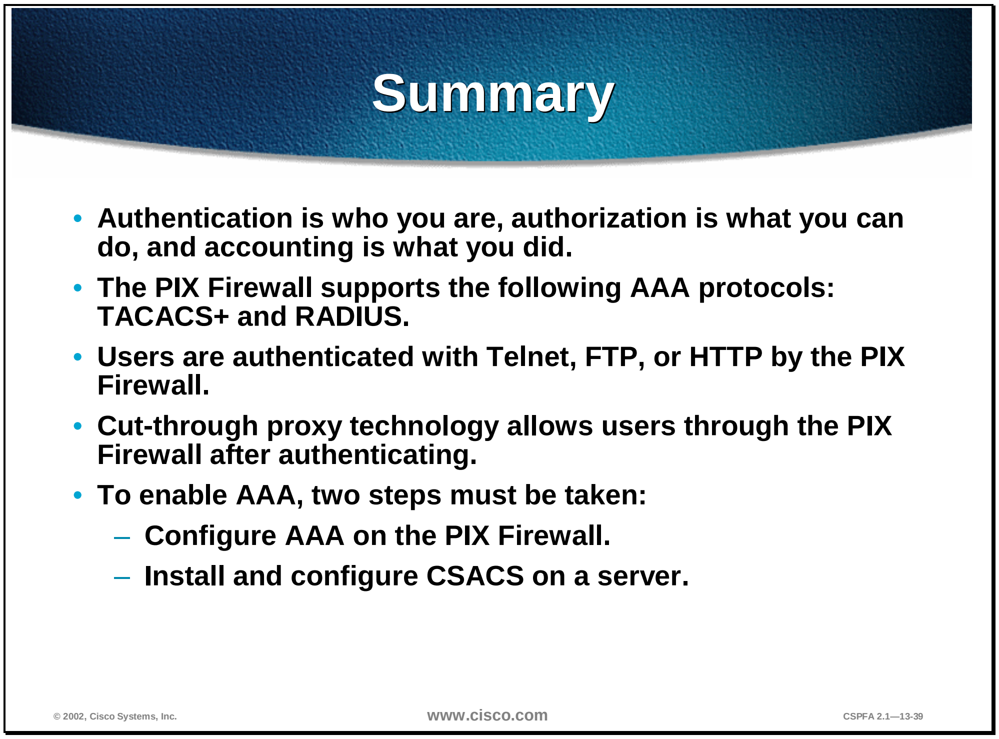
**show timeout uauth**

**show virtual** [**http** | **telnet**]

| | |
|---|---|
| **prompt** | Displays the prompt users get when authenticating. |
| **accept** | Displays the message users get when successfully authenticating. |
| **reject** | Displays the message users get when unsuccessfully authenticating. |
| **timeout uauth** | Displays the current uauth timer values for all authenticated users. |
| **http** | Displays the virtual HTTP configuration. |
| **telnet** | Displays the virtual Telnet configuration. |

# Summary

This section summarizes what you have learned in this chapter.



## Summary

- Authentication is who you are, authorization is what you can do, and accounting is what you did.
- The PIX Firewall supports the following AAA protocols: TACACS+ and RADIUS.
- Users are authenticated with Telnet, FTP, or HTTP by the PIX Firewall.
- Cut-through proxy technology allows users through the PIX Firewall after authenticating.
- To enable AAA, two steps must be taken:
  - Configure AAA on the PIX Firewall.
  - Install and configure CSACS on a server.

© 2002, Cisco Systems, Inc. www.cisco.com CSPFA 2.1—13-39

# Lab Exercise—Configure AAA on the PIX Firewall Using CSACS for Windows NT

Complete the following lab exercise to practice what you have learned in this chapter.

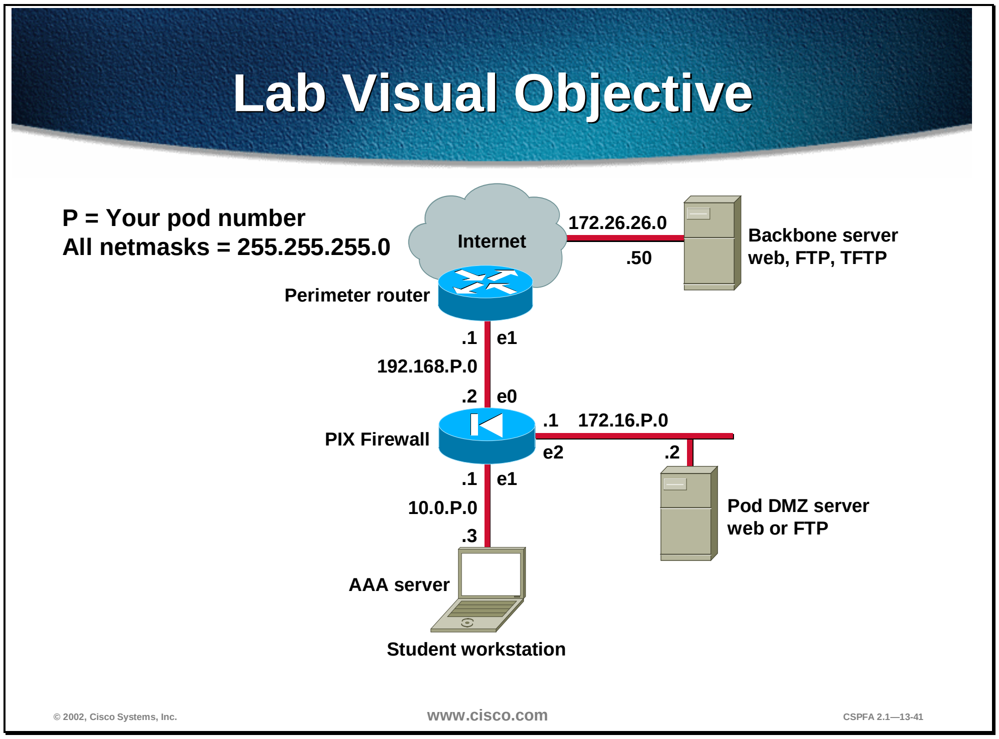
## Objectives

In this lab exercise you will complete the following tasks:

- Install CSACS for a Windows NT server.
- Add a user to the CSACS database.
- Identify a AAA server and protocol.
- Configure and test inbound authentication.
- Configure and test outbound authentication.
- Configure and test console access authentication.
- Configure and test Virtual Telnet authentication.
- Change and test authentication timeouts and prompts.
- Configure and test authorization.
- Configure and test accounting.

## Visual Objective

The following figure displays the configuration you will complete in this lab exercise.



## Lab Visual Objective

P = Your pod number
All netmasks = 255.255.255.0

www.cisco.com

CSPFA 2.1—13-41

## Task 1—Install CSACS for a Windows NT Server

Perform the following steps to install CSACS on your Windows NT server:

**Step 1** Install CSACS on your Windows NT server from the CD-ROM or from the files on your hard drive, as indicated by the instructor.

- When installing from the CD-ROM, complete the following:
  - Windows NT automatically starts the autorun.exe program and you are prompted to install CSACS.
  - Click **Install** to start the installation process.
- When installing from files in your hard drive, complete the following:
  - Open the folder where the installation files are located and double-click the **setup.exe** program to start installation.
  - Or choose **Start**>**Run** and enter **setup.exe** with a full path to the file.

**Step 2** Click **OK** in the Warning window.

**Step 3** Click **Accept** to accept the Software License Agreement. The Welcome window opens.

**Step 4** Read the Welcome panel. Click **Next** to continue. The Before You Begin window opens.

**Step 5** Read and then select all four check boxes for the items in the Before You Begin panel. This is a reminder of things you should do prior to installation. Click **Next** to continue. The Choose Destination Location window opens.

**Step 6** Use the default installation folder indicated in the Choose Destination Location windows by clicking **Next** to continue. The Authentication Database Configuration windows open.

**Step 7** Verify that Check the Cisco Secure ACS database only is already selected in the Authentication Database Configuration panel. Click **Next** to continue.

**Step 8** Enter the following information in the Cisco Secure ACS Network Access Server Details panel:

- Authenticate users: **TACACS+ (Cisco IOS)**
- Access server name: **pixP**

(where P = pod number)

- Access server IP address: **10.0.P.1**

(where P = pod number)

- Windows NT Server IP address: **10.0.P.3**

(where P = pod number)

- TACACS+ or RADIUS key: **secretkey**

**Step 9** Click **Next** to start the file installation process.

**Step 10** Select all six items displayed in the Advanced Options panel. Click **Next** to continue.

**Step 11** Verify that Enable Log-in Monitoring is already selected in the Active Service Monitoring panel. Click **Next** to continue.

***CAUTION*** *Do not select "Yes, I want to configure Cisco IOS software now" in the "Network Access Server Configuration" panel; this only applies to Cisco IOS routers.*

**Step 12** Click **Next** to continue.

**Step 13** Verify that the following are already selected in the Cisco Secure ACS Service Initiation panel:

- Yes, I want to start the Cisco Secure ACS Service now
- Yes, I want Setup to launch the Cisco Secure ACS Administrator from my browser following installation

**Note** Do not select "Yes, I want to review the Readme file."

**Step 14** Click **Next** to start the CSACS service.

**Step 15** Read the Setup Complete panel and then click **Finish** to end the installation wizard and start your web browser with CSACS.

## Task 2—Add a User to the CSACS Database

Perform the following steps to add a user to the CSACS database in your Windows NT server:

**Step 1** The CSACS interface should now be displayed in your web browser. Click **User Setup** to open the User Setup interface.

**Step 2** Add a user by entering **aaauser** in the user field.

**Step 3** Click **Add/Edit** to go into the user information edit window.

**Step 4** Give the user a password by entering **aaapass** in both the Password and Confirm Password fields.

**Step 5** Click **Submit** to add the new user to the CSACS database. Wait for the interface to return to the User Setup main window.

## Task 3—Identify a AAA Server and Protocol

Perform the following steps to identify a AAA server and a AAA protocol on the PIX Firewall:

**Step 1** Create a group tag called MYTACACS and assign the TACACS+ protocol to it:

```
pixP(config)# aaa-server MYTACACS protocol tacacs+
```

**Step 2** Assign the CSACS IP address and the encryption key secretkey.

```
pixP(config)# aaa-server MYTACACS (inside) host 10.0.P.3 secretkey
```

(where P = pod number)

**Step 3** Verify your configuration:

```
pixP(config)# show aaa-server
aaa-server TACACS+ protocol tacacs+
aaa-server RADIUS protocol radius
aaa-server MYTACACS protocol tacacs+
aaa-server MYTACACS (inside) host 10.0.P.3 secretkey timeout 5
```

## Task 4—Configure and Test Inbound Authentication

Perform the following steps to enable the use of inbound authentication on the PIX Firewall:

**Step 1** Configure the PIX Firewall to require authentication for all inbound traffic:

```
pixP(config)# aaa authentication include any inbound 0.0.0.0 0.0.0.0 0.0.0.0
0.0.0.0 MYTACACS
```

**Step 2** Verify your configuration:

```
pixP(config)# show aaa authentication
aaa authentication include any outside 0.0.0.0 0.0.0.0 0.0.0.0 0.0.0.0 MYTACACS
```

**Step 3** Enable console logging of all messages:

```
pixP(config)# logging console debug
```

**Note** If your web browser is open, close it. Choose **File>Close** from the web browser's menu.

**Step 4** You must now test a peer pod inbound web authentication. Open your web browser, and go to a peer's DMZ web server:

```
http://192.168.Q.11
```

(where Q = peer pod number)

**Step 5** When the web browser prompts you, enter **aaauser** for the username and **aaapass** for the password. On your PIX Firewall console, you should see the following:

```
109001: Auth start for user '???' from 192.168.Q.10/1726 to 172.16.P.2/80
109011: Authen Session Start: user 'aaauser', sid 0
109005: Authentication succeeded for user 'aaauser' from 172.16.P.2/80 to
192.168.Q.10/1921 on interface outside
302001: Built inbound TCP connection 3928 for faddr 192.168.Q.10/1921 gaddr
192.168.P.11/80 laddr 172.16.P.2/80 (aaauser)
```

**Step 6** After a peer successfully authenticates to your PIX Firewall, display your PIX Firewall authentication statistics:

```
pixP(config)# show uauth
                        Current    Most Seen
Authenticated Users       1          1
Authen In Progress        0          1
user 'pixuser' at 192.168.Q.10, authenticated
   absolute   timeout: 0:05:00
   inactivity timeout: 0:00:00
```

## Task 5—Configure and Test Outbound Authentication

Perform the following steps to enable the use of outbound authentication on the PIX Firewall:

**Step 1** Configure the PIX Firewall to require authentication for all outbound traffic:

```
pixP(config)# aaa authentication include any outbound 0.0.0.0 0.0.0.0 0.0.0.0
0.0.0.0 MYTACACS
```

**Step 2** Verify your configuration:

```
pixP(config)# show aaa authentication
aaa authentication include any outside 0.0.0.0 0.0.0.0 0.0.0.0 0.0.0.0 MYTACACS
aaa authentication include any inside 0.0.0.0 0.0.0.0 0.0.0.0 0.0.0.0 MYTACACS
```

**Step 3** Test FTP outbound authentication from your Windows NT server:

```
C:\> ftp 172.26.26.50
Connected to 172.26.26.50
220-FTP authentication :
220
User (172.26.26.50:(none)): aaauser@ftpuser
331-Password:
331
Password: aaapass@ftppass
230-220 172.26.26.50 FTP server ready.
331-Password required for ftpuser
230-User ftpuser logged in.
230
ftp>
```

On your PIX Firewall console, you should see the following:

```
109001: Auth start for user '???' from 10.0.P.3/1726 to 172.26.26.50/21
109011: Authen Session Start: user 'aaauser', sid 11
109005: Authentication succeeded for user 'aaauser' from 10.0.P.3/1726 to
172.26.26.50/21 on interface inside
302001: Built outbound TCP connection 3928 for faddr 172.26.26.50/21 gaddr
192.168.P.10/1726 laddr 10.0.P.3/1726 (aaauser)
```

(where P = pod number)

**Step 4** Display authentication statistics on the PIX Firewall:

```
pixP(config)# show uauth
                        Current    Most Seen
Authenticated Users       1          1
Authen In Progress        0          1
user 'aaauser' at 10.0.P.3, authenticated (P = your pod number)
   absolute   timeout: 0:05:00
   inactivity timeout: 0:00:00
```

**Step 5** Clear the uauth timer:

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

**Note** If your web browser is open, close it. Choose **File>Exit** from the web browser's menu.

**Step 6** Test web outbound authentication. Open your web browser and go to the following URL:

```
http://172.26.26.50
```

**Step 7** When you are prompted for a username and password, enter **aaauser** as the username and **aaapass** as the password:

```
User Name: aaauser
Password: aaapass
```

**Step 8** Display authentication statistics on the PIX Firewall:

```
pixP(config)# show uauth
                        Current    Most Seen
Authenticated Users       1          1
Authen In Progress        0          1
user 'pixuser' at 10.0.P.2, authenticated
   absolute   timeout: 0:05:00
   inactivity timeout: 0:00:00
```

(where P = pod number)

## Task 6—Configure and Test Console Access Authentication

Perform the following steps to enable console Telnet authentication at the PIX Firewall:

**Step 1** Configure the PIX Firewall to require authentication for Telnet console connections:

```
pixP(config)# aaa authentication telnet console MYTACACS
```

**Step 2** Verify your configuration:

```
pixP(config)# show aaa authentication
aaa authentication include any outside 0.0.0.0 0.0.0.0 0.0.0.0 0.0.0.0 MYTACACS
aaa authentication include any inside 0.0.0.0 0.0.0.0 0.0.0.0 0.0.0.0 MYTACACS
aaa authentication include any 0.0.0.0 0.0.0.0 0.0.0.0 0.0.0.0 MYTACACS
```

**Step 3** Configure the PIX Firewall to allow console Telnet logins:

```
pixP(config)# telnet 10.0.P.3 255.255.255.255 inside
```

(where P = pod number)

**Step 4** Verify your configuration:

```
pixP(config)# show telnet
10.0.P.3 255.255.255.255 inside
```

**Step 5** Clear the uauth timer:

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

**Step 6** Save your configuration:

```
pixP(config)# write memory
```

**Step 7** Reboot your PIX Firewall:

```
pixP(config)# reload
```

**Step 8** Telnet to the PIX Firewall console:

```
C:\> telnet 10.0.P.1
Username: aaauser
Password: aaapass
Type help or '?' for a list of available commands.
pixP>
```

(where P = pod number)

On your PIX Firewall console, you should see the following:

```
307002: Permitted Telnet login session from 10.0.P.3
111006: Console Login from aaauser at console
```

## Task 7—Configure and Test Virtual Telnet Authentication

Perform the following steps to enable the use of authentication with virtual Telnet on the PIX Firewall:

**Step 1** Configure the PIX Firewall to accept authentication to a virtual Telnet service:

```
pixP(config)# virtual telnet 192.168.P.5
```

(where P = pod number)

**Step 2** Verify the virtual Telnet configuration:

```
pixP(config)# show virtual telnet
virtual telnet 192.168.P.5
```

(where P = pod number)

**Step 3** Clear the uauth timer:

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

**Step 4** Telnet to the virtual Telnet IP address to authenticate from your Windows NT server:

```
C:\> telnet 192.168.P.5
LOGIN Authentication
Username: aaauser
Password: aaapass
Authentication Successful
```

(where P = pod number)

**Note** If your web browser is open, close it. Choose **File>Close** from the web browser's menu.

**Step 5** Test that you are authenticated. Open your web browser and enter the following in the URL field:

```
http://172.26.26.50
```

You should not be prompted to authenticate.

**Step 6** Clear the uauth timer:

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

**Note** If your web browser is open, close it. Choose **File>Close** from the web browser's menu.

**Step 7** Test that you are not authenticated and need to reauthenticate. Open your web browser and enter the following in the URL field:

```
http://172.26.26.50
```

**Step 8** When you are prompted, enter **aaauser** for the username and **aaapass** for the password.

## Task 8—Change and Test Authentication Timeouts and Prompts

Perform the following steps to change the authentication timeouts and prompts:

**Step 1** View the current uauth timeout settings:

```
pixP(config)# show timeout uauth
timeout uauth 0:05:00 absolute
```

**Step 2** Set the uauth absolute timeout to 3 hours:

```
pixP(config)# timeout uauth 3 absolute
```

**Step 3** Set the uauth inactivity timeout to 30 minutes:

```
pixP(config)# timeout uauth 0:30 inactivity
```

**Step 4** Verify the new uauth timeout settings:

```
pixP(config)# show timeout uauth
timeout uauth 3:00:00 absolute uauth 0:30:00 inactivity
```

**Step 5** View the current authentication prompt settings:

```
pixP(config)# show auth-prompt
```

Nothing should be displayed.

**Step 6** Set the prompt that users get when authenticating:

```
pixP(config)# auth-prompt prompt Please Authenticate
```

**Step 7** Set the message that users get when successfully authenticating:

```
pixP(config)# auth-prompt accept You’ve been Authenticated
```

**Step 8** Set the message that users get when their authentication is rejected:

```
pixP(config)# auth-prompt reject Authentication Failed, Try Again
```

**Step 9** Verify the new prompt settings:

```
pixP(config)# show auth-prompt
auth-prompt prompt Please Authenticate
auth-prompt accept You've been Authenticated
auth-prompt reject Authentication Failed, Try Again
```

**Step 10** Clear the uauth timer:

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

**Step 11** Telnet to the Virtual Telnet IP address to test your new authentication prompts. From your Windows NT server, enter the following:

```
C:\> telnet 192.168.P.5
LOGIN Authentication
Please Authenticate
Username: wronguser
Password: wrongpass Authentication Failed, Try Again
LOGIN Authentication
Please Authenticate
Username: aaauser
Password: aaapass
You’ve been Authenticated
Authentication Successful
```

(where P = pod number)

## Task 9—Configure and Test Authorization

Perform the following steps to enable the use of authorization on the PIX Firewall:

**Step 1** Configure the PIX Firewall to require authorization for all outbound FTP traffic:

```
pixP(config)# aaa authorization include ftp outbound 0.0.0.0 0.0.0.0 0.0.0.0
0.0.0.0 MYTACACS
```

**Step 2** Configure the PIX Firewall to require authorization for all outbound ICMP traffic:

```
pixP(config)# aaa authorization include icmp/8 outbound 0.0.0.0 0.0.0.0 0.0.0.0
0.0.0.0 MYTACACS
```

**Step 3** Verify your configuration:

```
pixP(config)# show aaa authorization
aaa authorization include ftp inside 0.0.0.0 0.0.0.0 0.0.0.0 0.0.0.0 MYTACACS
aaa authorization include 1/8 inside 0.0.0.0 0.0.0.0 0.0.0.0 0.0.0.0 MYTACACS
```

**Step 4** Test ICMP Echo Request failure from your Windows NT server:

```
C:\> ping 172.26.26.50
Pinging 172.26.26.50 with 32 bytes of data:
Request timed out.
Request timed out.
Request timed out.
Request timed out.
```

On your PIX Firewall console, you should see the following:

```
109001: Auth start for user 'aaauser' from 10.0.P.3/0 to 172.26.26.50/0
109008: Authorization denied for user 'aaauser' from 10.0.P.2/0 to 172.26.26.50/0
on interface inside
```

**Step 5** Test FTP authorization failure from your Windows NT server:

```
C:\> ftp 172.26.26.50
Connected to 172.26.26.50
220-FTP authentication :
220
User (172.26.26.50:(none)): aaauser@ftpuser
331-Password:
331
Password: aaapass@ftppass
530
530-Authorization Denied
530
Error:  Connection closed by foreign host.
```

On your PIX Firewall console, you should see the following:

```
109001: Auth start for user '???' from 10.0.P.3/1364 to 172.26.26.50/21
109011: Authen Session Start: user 'aaauser', sid 5
109005: Authentication succeeded for user 'aaauser' from 10.0.P.3/1364 to
172.26.26.50/21 on interface inside
109008: Authorization denied for user 'aaauser' from 10.0.P.3/1364 to
172.26.26.50/21 on interface inside
```

(where P = pod number)

**Step 6** Click **Group Setup** to open the Group Setup interface.

**Step 7** Choose **Default Group (1 user)** from the Group drop-down menu.

**Step 8** Verify that your user belongs to the selected group. Click **Users in Group** to display the users under that group. The following information should be shown for the user:

- User: **aaauser**
- Status: **Enabled**
- Group: **Default Group (1 user)**

**Step 9** Click **Edit Settings** to go to the Group Settings for your group.

**Step 10** Scroll down in Group Settings until you find IOS Commands, and select the **IOS Commands** check box.

**Step 11** Select the **Command** check box.

**Step 12** Enter **ftp** in the Command field.

**Step 13** Enter **permit 172.26.26.50** in the Arguments field.

**Step 14** Click **Submit** to save the changes. Wait for the interface to return to the Group Setup main window.

**Step 15** Click **Edit Settings** to go to the Group Settings for your group again.

**Step 16** Scroll down in Group Settings until you find IOS Commands.

**Step 17** Select the **Command** check box.

**Step 18** Enter **1/8** in the Command field.

**Step 19** Select Permit in the Unlisted arguments field.

**Step 20** Click Submit + Restart to save the changes and restart CSACS. Wait for the interface to return to the Group Setup main window.

**Step 21** Test FTP authorization success from your Windows NT server:

```
C:\> ftp 172.26.26.50
Connected to 172.26.26.50
220-FTP authentication :
220
User (172.26.26.50:(none)): aaauser@ftpuser
331-Password:
331
Password: aaapass@ftppass
230-220 172.26.26.50 FTP server ready.
331-Password required for ftpuser
230-User ftpuser logged in.
230
ftp>
```

On your PIX Firewall console, you should see the following:

```
109001: Auth start for user '???' from 10.0.P.3/1726 to 172.26.26.50/21
109011: Authen Session Start: user 'aaauser', sid 11
109005: Authentication succeeded for user 'aaauser' from 10.0.P.3/1726 to
172.26.26.50/21 on interface inside
109011: Authen Session Start: user 'aaauser', sid 11
109007: Authorization permitted for user 'aaauser' from 10.0.P.3/1726 to
172.26.26.50/21 on interface inside
```

```
302001: Built outbound TCP connection 3928 for faddr 172.26.26.50/21 gaddr
192.168.P.10/1726 laddr 10.0.P.3/1726 (aaauser)
```

(where P = pod number)

**Step 22** Test ICMP Echo Request success from your Windows NT server:

```
C:\> ping 172.26.26.50
Pinging 172.26.26.50 with 32 bytes of data:
Request timed out.
Request timed out.
Reply from 172.26.26.50: bytes=32 time<10ms TTL=128
Reply from 172.26.26.50: bytes=32 time<10ms TTL=128
```

On your PIX Firewall console, you should see the following:

```
109001: Auth start for user 'aaauser' from 10.0.P.3/0 to 172.26.26.50/0
109011: Authen Session Start: user 'aaauser', sid 1
109007: Authorization permitted for user 'aaauser' from 10.0.P.2/0 to
172.26.26.50/0 on interface inside
```

(where P = pod number)

## Task 10—Configure and Test Accounting

Perform the following steps to enable the use of accounting on the PIX Firewall:

**Step 1** Configure the PIX Firewall to perform accounting for all outbound traffic:

```
pixP(config)# aaa accounting include any outbound 0.0.0.0 0.0.0.0 0.0.0.0 0.0.0.0
MYTACACS
```

**Step 2** Verify your configuration:

```
pixP(config)# show aaa accounting
aaa accounting include any inside 0.0.0.0 0.0.0.0 0.0.0.0 0.0.0.0 MYTACACS
```

**Step 3** Clear the uauth timer:

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

**Step 4** Test FTP outbound accounting from your Windows NT server:

```
C:\> ftp 172.26.26.50
Connected to 172.26.26.50
220-FTP authentication :
220
User (172.26.26.50:(none)): aaauser@ftpuser
331-Password:
331
Password: aaapass@ftppass
230-220 172.26.26.50 FTP server ready.
331-Password required for ftpuser
230-User ftpuser logged in.
230
ftp>
```

**Step 5** View the accounting records. On CSACS, click **Reports and Activity** to open the Reports and Activity interface.

**Step 6** Click the **TACACS+ Accounting** link.

**Step 7** Click the **TACACS+ Accounting active.csv** link to open the accounting records. You should see the following:

| Date | Time | User-Name | Group-Name | Caller-Id | Acct-Flags | • • • | NAS-Portname | NAS-IP-Address | cmd |
|---|---|---|---|---|---|---|---|---|---|
| 4/27/00 | 11:14:45 | aaauser | Default Group | 10.0.P.2 | start | • • • | PIX | 10.0.P.1 | ftp |

**Note** If your web browser is open, close it. Choose **File>Exit** from the web browser's menu.

**Step 8** Test web outbound accounting. Open your web browser and enter the following in the URL field:

```
http://172.26.26.50
```

**Step 9** Click the **TACACS+ Accounting** link.

**Step 10** Click the **TACACS+ Accounting active.csv** link to open the accounting records. You should see the following:

| Date | Time | User-Name | Group-Name | Caller-Id | Acct-Flags | | NAS-Portname | NAS-IP-Address | cmd |
|---|---|---|---|---|---|---|---|---|---|
| 4/27/00 | 11:16:35 | aaauser | Default Group | 10.0.0.2 | start | • • • | PIX | 10.0.0.1 | http |
| 4/27/00 | 11:16:35 | aaauser | Default Group | 10.0.0.2 | start | • • • | PIX | 10.0.0.1 | http |
| | | | | | | | | | |
| 4/27/00 | 11:16:34 | aaauser | Default Group | 10.0.0.2 | start | • • • | PIX | 10.0.0.1 | http |
| 4/27/00 | 11:16:34 | aaauser | Default Group | 10.0.0.2 | stop | • • • | PIX | 10.0.0.1 | http |
| 4/27/00 | 11:16:34 | aaauser | Default Group | 10.0.0.2 | stop | • • • | PIX | 10.0.0.1 | http |
| 4/27/00 | 11:16:34 | aaauser | Default Group | 10.0.0.2 | start | • • • | PIX | 10.0.0.1 | http |
| 4/27/00 | 11:16:34 | aaauser | Default Group | 10.0.0.2 | start | • • • | PIX | 10.0.0.1 | http |
| 4/27/00 | 11:16:34 | aaauser | Default Group | 10.0.0.2 | start | • • • | PIX | 10.0.0.1 | http |
| 4/27/00 | 11:16:33 | aaauser | Default Group | 10.0.0.2 | start | • • • | PIX | 10.0.0.1 | http |
| 4/27/00 | 11:16:32 | aaauser | Default Group | 10.0.0.2 | start | • • • | PIX | 10.0.0.1 | http |
| 4/27/00 | 11:16:29 | aaauser | Default Group | 10.0.0.2 | start | • • • | PIX | 10.0.0.1 | http |
| 4/27/00 | 11:14:45 | aaauser | Default Group | 10.0.0.2 | start | • • • | PIX | 10.0.0.1 | ftp |

**Step 11** Disable AAA by entering the following command:

```
pixP(config)# clear aaa
```

**Step 12** Remove the **aaa-server** commands from the configuration:

```
pixP(config)# clear aaa-server
```

**Step 13** Turn off the logging:

```
pixP(config)# no logging console debug
```

14

# Failover

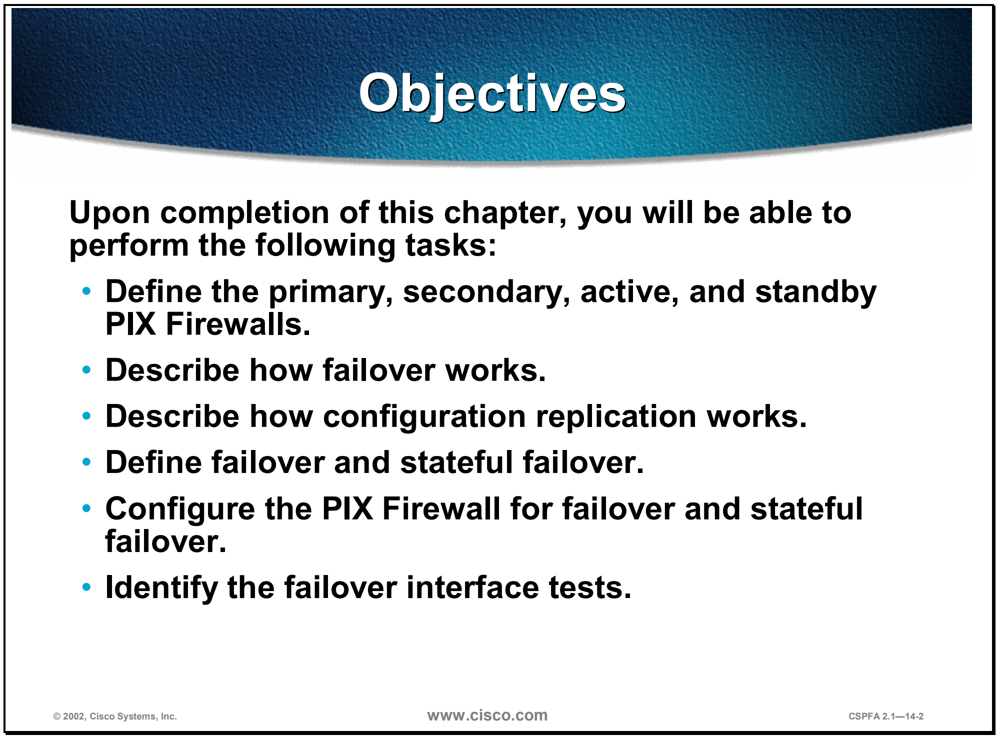
## Overview

This chapter includes the following topics:

- Objectives
- Understand failover
- Configure failover
- Summary
- Lab exercise

# Objectives

This section lists the chapter's objectives.



## Objectives

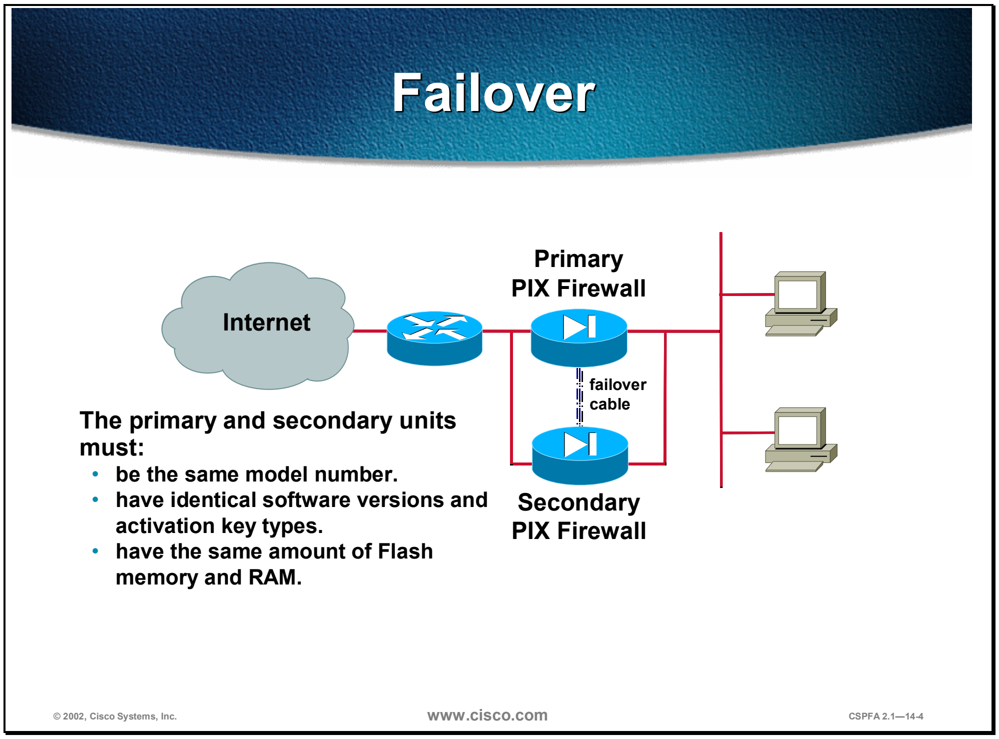
**Upon completion of this chapter, you will be able to perform the following tasks:**

- **Define the primary, secondary, active, and standby PIX Firewalls.**
- **Describe how failover works.**
- **Describe how configuration replication works.**
- **Define failover and stateful failover.**
- **Configure the PIX Firewall for failover and stateful failover.**
- **Identify the failover interface tests.**

© 2002, Cisco Systems, Inc. www.cisco.com CSPFA 2.1—14-2

# Understand Failover

This section discusses what failover is and how it works.



## Failover

**The primary and secondary units must:**

- **be the same model number.**
- **have identical software versions and activation key types.**
- **have the same amount of Flash memory and RAM.**

© 2002, Cisco Systems, Inc. www.cisco.com CSPFA 2.1—14-4

The failover function for the Cisco Secure PIX Firewall™ provides a safeguard in case a PIX Firewall fails. Specifically, when one PIX Firewall fails, another immediately takes its place. In order for failover to work, both units must have the same software version, activation key type, Flash memory, and RAM.

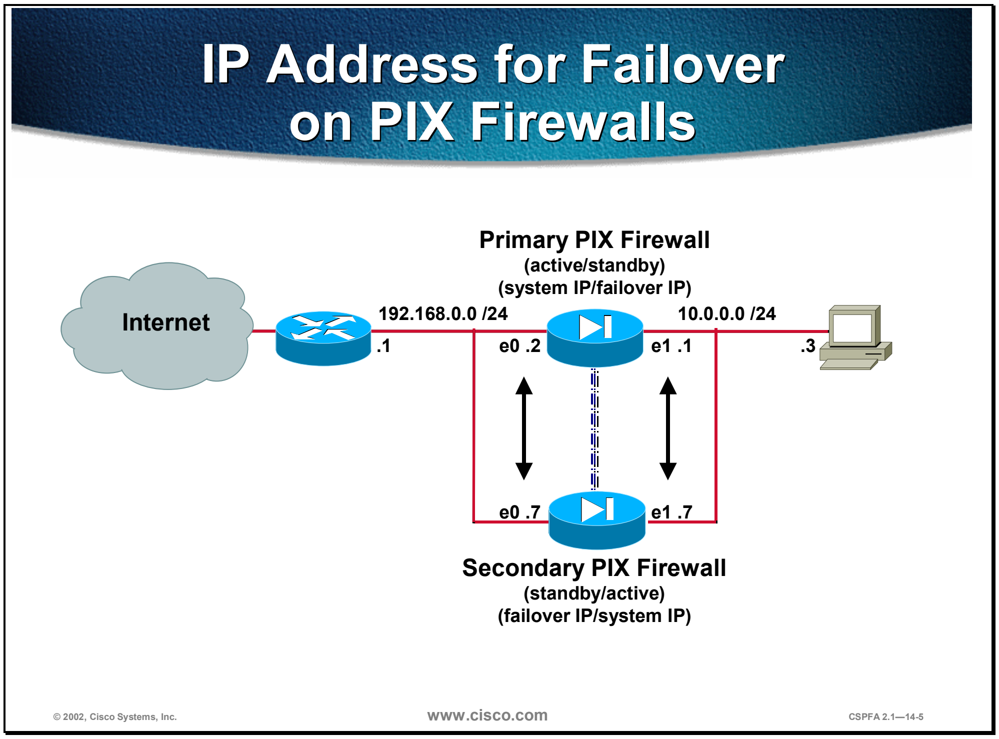
In the failover process, there are two PIX Firewalls: the primary PIX Firewall and the secondary PIX Firewall. The primary PIX Firewall functions as the active PIX Firewall, performing normal network functions. The secondary PIX Firewall functions as the standby PIX Firewall, ready to take control should the active PIX Firewall fail to perform. When the primary PIX Firewall fails, the secondary PIX Firewall becomes active while the primary PIX Firewall goes on standby. This entire process is called *failover*.

The primary PIX Firewall is connected to the secondary PIX Firewall through a failover connection: a failover cable. The failover cable has one end labeled *primary*, which plugs into the primary PIX Firewall, and the other end labeled *secondary*, which plugs into the secondary PIX Firewall.

A failover occurs when one of the following situations takes place:

- A power-off or a power-down condition occurs on the active PIX Firewall
- The active PIX Firewall is rebooted
- A link goes down on the active PIX Firewall for more than 30 seconds

- The message "Failover active" occurs on the standby PIX Firewall
- Block memory exhaustion occurs for 15 consecutive seconds or more on the active PIX Firewall

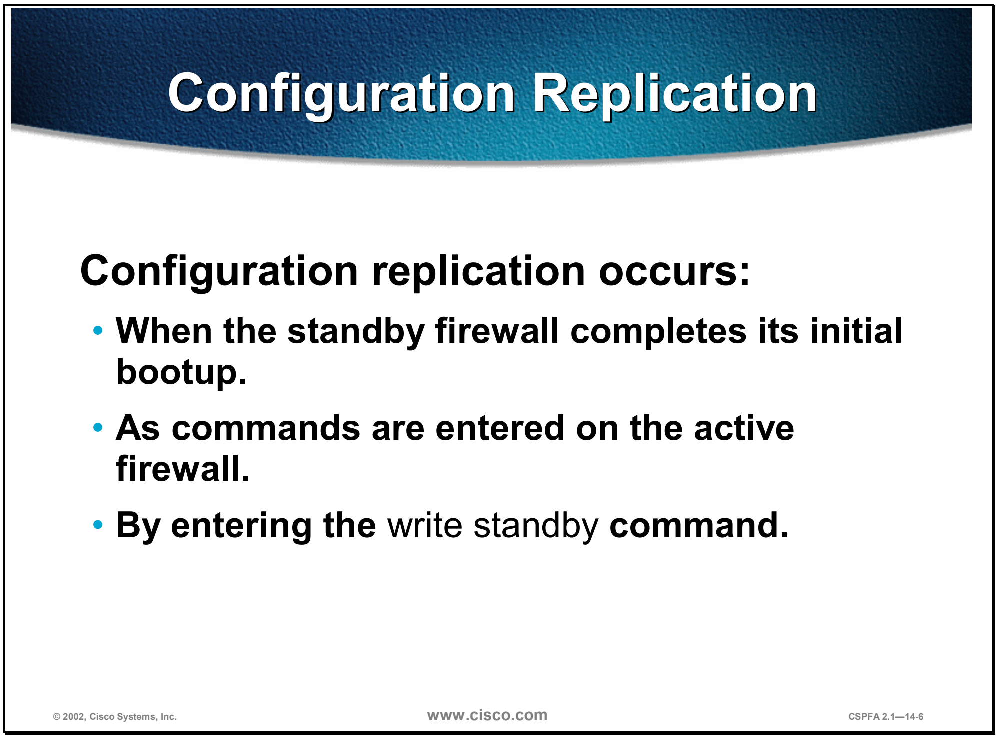


# IP Address for Failover on PIX Firewalls

© 2002, Cisco Systems, Inc.    www.cisco.com    CSPFA 2.1—14-5

When actively functioning, the primary PIX Firewall uses system IP addresses and MAC addresses. The secondary PIX Firewall, when on standby, uses failover IP addresses and MAC addresses.

When the primary PIX Firewall fails and the secondary PIX Firewall becomes active, the secondary PIX Firewall assumes the system IP addresses and MAC addresses of the primary PIX Firewall. The primary PIX Firewall, functioning in standby, then assumes the failover IP addresses and MAC addresses of the secondary PIX Firewall.



## Configuration Replication

**Configuration replication occurs:**

- **When the standby firewall completes its initial bootup.**
- **As commands are entered on the active firewall.**
- **By entering the** write standby **command.**

© 2002, Cisco Systems, Inc. www.cisco.com CSPFA 2.1—14-6

Configuration replication is when the configuration of the primary PIX Firewall is replicated to the secondary PIX Firewall. To perform configuration replication, both the primary and secondary PIX Firewalls must be configured exactly the same and run the same software release. Configuration replication occurs over the failover cable from the active PIX Firewall to the standby PIX Firewall in three ways:

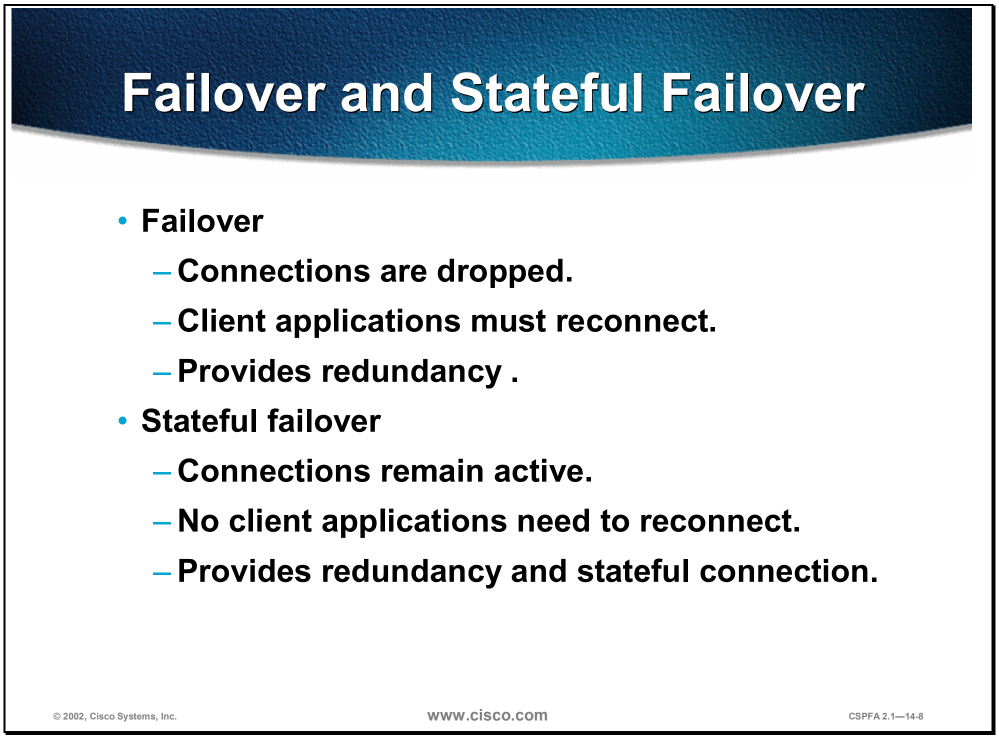
- When the standby PIX Firewall completes its initial bootup, the active PIX Firewall replicates its entire configuration to the standby PIX Firewall.
- As commands are entered on the active PIX Firewall, they are sent across the failover cable to the standby PIX Firewall.
- Entering the **write standby** command on the active PIX Firewall forces the entire configuration in memory to be sent to the standby PIX Firewall.

Configuration replication only occurs from memory to memory. Because this is not a permanent place to store configurations, you must use the **write memory** command to write the configuration into Flash memory. If a failover occurs during the replication, the new active PIX Firewall will have only a partial configuration. The newly active PIX Firewall will then reboot itself to recover the configuration from the Flash memory or resynchronize with the new standby PIX Firewall.

When replication starts, the PIX Firewall console displays the message “Sync Started”, and when complete, displays the message “Sync Completed”. During replication, information cannot be entered on the PIX Firewall console. Replication can take a long time to complete for a large configuration because the failover cable is used.

# Configure Failover

This section discusses what failover and stateful failover modes are, and how to configure stateful failover.



## Failover and Stateful Failover

- Failover
  - Connections are dropped.
  - Client applications must reconnect.
  - Provides redundancy .
- Stateful failover
  - Connections remain active.
  - No client applications need to reconnect.
  - Provides redundancy and stateful connection.

© 2002, Cisco Systems, Inc. www.cisco.com CSPFA 2.1—14-8

As stated earlier in the chapter, failover enables the standby PIX Firewall to take over the duties of the active PIX Firewall when the active PIX Firewall fails. There are two types of failover:

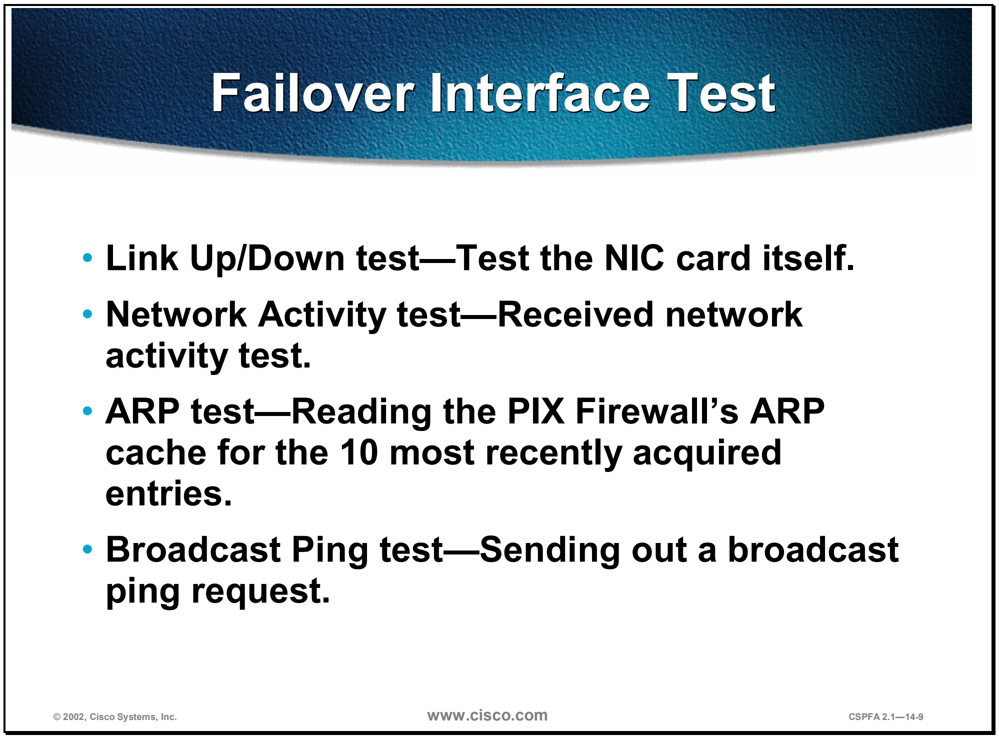
- Failover—When the active PIX Firewall fails and the standby PIX Firewall becomes active, all connections are lost and client applications must perform a new connection to restart communication through the PIX Firewall. The disconnection happens because the active PIX Firewall does not pass the stateful connection information to the standby PIX Firewall.
- Stateful failover—When the active PIX Firewall fails and the standby PIX Firewall becomes active, the same connection information is available at the new active PIX Firewall, and end-user applications are not required to do a reconnect to keep the same communication session. The connections remain because the stateful failover feature passes per-connection stateful information to the standby PIX Firewall.

Stateful failover requires a 100 Mbps Ethernet interface to be used exclusively for passing state information between the two PIX Firewalls. This interface can be connected to any of the following:

- Category 5 crossover cable directly connecting the primary PIX Firewall to the secondary PIX Firewall
- 100BaseTX half-duplex hub using straight Category 5 cables

- 100BaseTX full duplex on a dedicated switch or dedicated virtual LAN (VLAN) of a switch

**Note** The PIX Firewall does not support the use of either Token Ring or FDDI for the stateful failover dedicated interface. Data is passed over the dedicated interface using IP protocol 105. No hosts or routers should be on this interface.



## Failover Interface Test

- Link Up/Down test—Test the NIC card itself.
- Network Activity test—Received network activity test.
- ARP test—Reading the PIX Firewall's ARP cache for the 10 most recently acquired entries.
- Broadcast Ping test—Sending out a broadcast ping request.

© 2002, Cisco Systems, Inc. www.cisco.com CSPFA 2.1—14-9

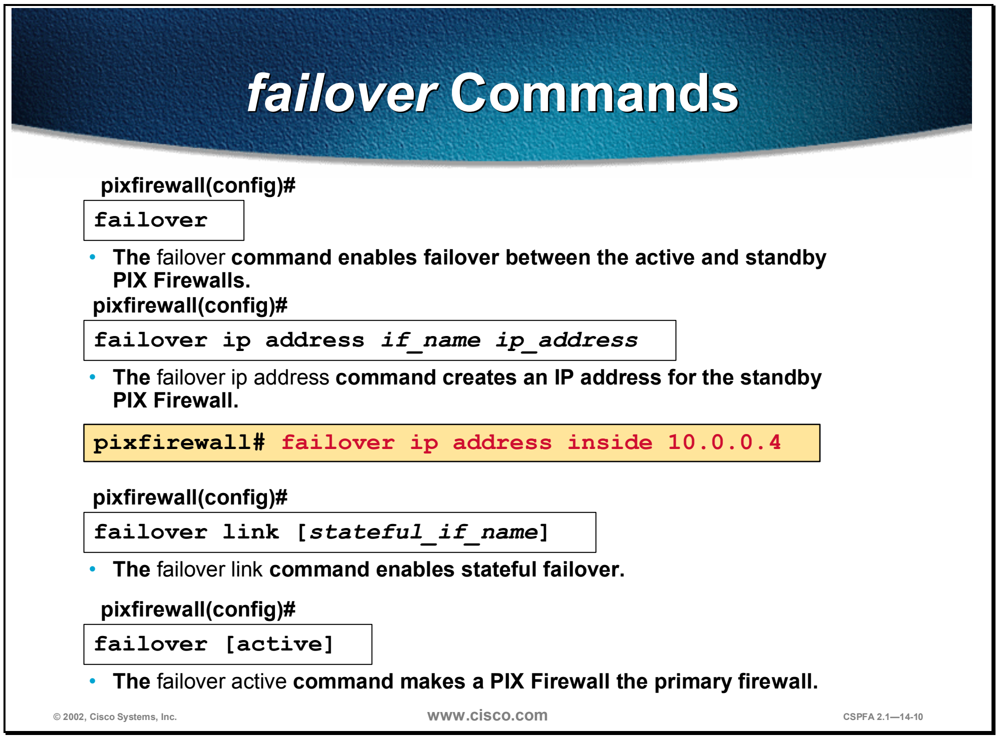
Both the primary and secondary PIX Firewalls send special failover "hello" packets to each other over all network interfaces and the failover cable every 15 seconds to make sure that everything is working. When a failure occurs in the active PIX Firewall, and it is not because of a loss of power in the standby PIX Firewall, failover begins a series of tests to determine which PIX Firewall has failed. The purpose of these tests is to generate network traffic to determine which (if either) PIX Firewall has failed.

At the start of each test, each PIX Firewall clears its received packet count for its interfaces. At the conclusion of each test, each PIX Firewall looks to see if it has received any traffic. If it has, the interface is considered operational. If one PIX Firewall receives traffic for a test and the other PIX Firewall does not, the PIX Firewall that did not receive traffic is considered failed. If neither PIX Firewall has received traffic, the tests then continue.

The following are the four different tests used to test for failover:

- LinkUp/Down—This is a test of the NIC itself. If an interface card is not plugged into an operational network, it is considered failed (for example, the hub or switch has failed, has a failed port, or a cable is unplugged). If this test does not find anything, the Network Activity test begins.
- Network Activity—This is a received network activity test. The PIX Firewall counts all received packets for up to five seconds. If any packets are received at any time during this interval, the interface is considered operational and testing stops. If no traffic is received, the ARP test begins.
- ARP—The ARP test consists of reading the PIX Firewall's ARP cache for the ten most recently acquired entries. The PIX Firewall sends ARP requests one at a time to these machines, attempting to stimulate network traffic. After each request, the PIX Firewall counts all received traffic for up to five seconds. If traffic is received, the interface is considered operational. If no traffic is received, an ARP request is sent to the next machine. If at the end of the list no traffic has been received, the ping test begins.

- Broadcast Ping—The ping test consists of sending out a broadcast ping request. The PIX Firewall then counts all received packets for up to five seconds. If any packets are received at any time during this interval, the interface is considered operational and testing stops. If no traffic is received, the testing starts over again with the ARP test.



## *failover* Commands

pixfirewall(config)#

```
failover
```

- **The** failover **command enables failover between the active and standby PIX Firewalls.**

pixfirewall(config)#

```
failover ip address if_name ip_address
```

- **The** failover ip address **command creates an IP address for the standby PIX Firewall.**

```
pixfirewall# failover ip address inside 10.0.0.4
```

pixfirewall(config)#

```
failover link [stateful_if_name]
```

- **The** failover link **command enables stateful failover.**

pixfirewall(config)#

```
failover [active]
```

- **The** failover active **command makes a PIX Firewall the primary firewall.**

© 2002, Cisco Systems, Inc. www.cisco.com CSPFA 2.1—14-10

Use the **failover** command to enable failover between two PIX Firewalls. The syntax for the **failover** command is as follows:

**failover**

Use the **failover ip address** command to configure the failover IP address for the standby PIX Firewall. The syntax for the **failover ip address** command is as follows:

**failover ip address** ***if_name ip_address***

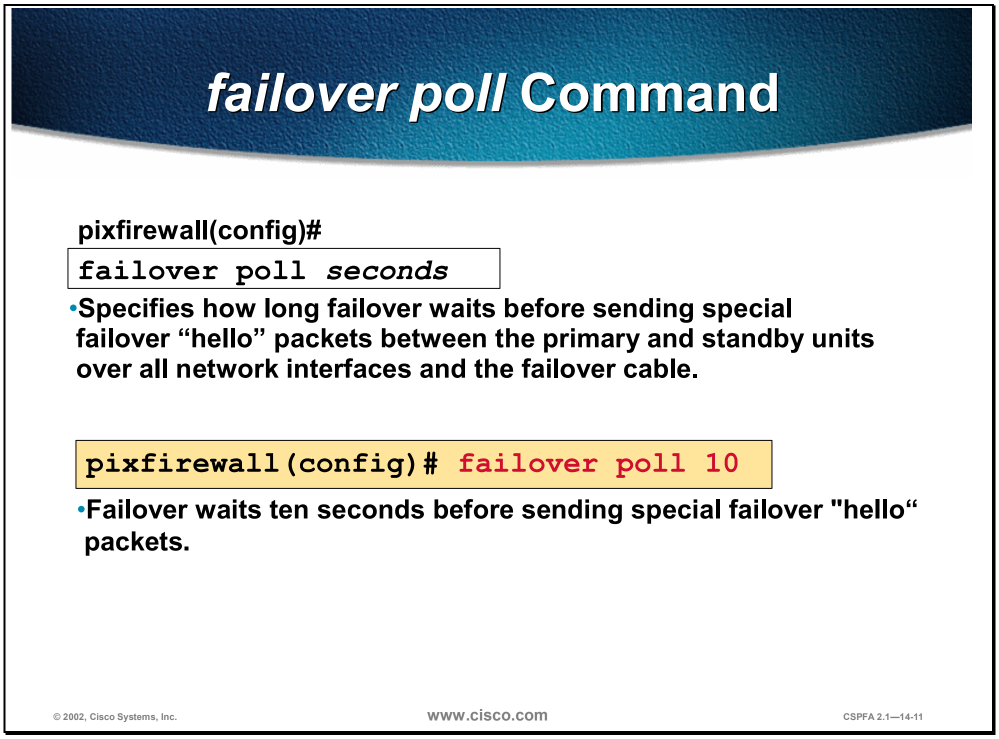
The **failover link** command enables stateful failover. The syntax for the **failover link** command is as follows:

**failover link [*stateful_if_name*]**

Use the **failover active** command to make a PIX Firewall the active PIX Firewall. The syntax for the **failover active** command is as follows:

**failover** [*active*]

| | |
|---|---|
| **active** | Makes a PIX Firewall the active PIX Firewall. Use this command when you need to force control of the connection back to the unit you are accessing, such as when you want to switch control back from a PIX Firewall after you have fixed a problem and want to restore service to the primary PIX Firewall. |
| ***if_name*** | Interface on which the standby PIX Firewall resides. |
| ***ip_address*** | The IP address used by the standby PIX Firewall to communicate with the active PIX Firewall. |
| **link** | Specifies the interface where a fast LAN link is available for stateful failover. |
| ***stateful_if_name*** | In addition to the failover cable, a dedicated fast LAN link is required to support stateful failover. The default interface is the highest LAN port with failover configured. |



# *failover poll* Command

pixfirewall(config)#

```
failover poll seconds
```

- Specifies how long failover waits before sending special failover “hello” packets between the primary and standby units over all network interfaces and the failover cable.

```
pixfirewall(config)# failover poll 10
```

- Failover waits ten seconds before sending special failover "hello“ packets.

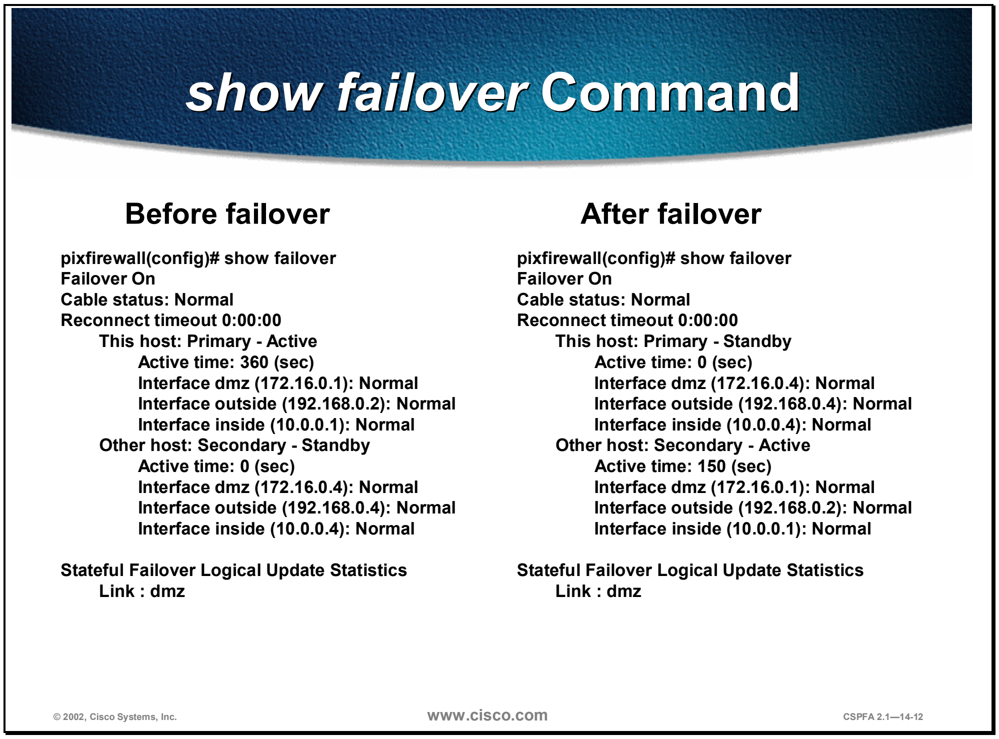
© 2002, Cisco Systems, Inc. www.cisco.com CSPFA 2.1—14-11

The **failover poll** command enables you to determine how long failover waits before sending the failover "hello" packets between the primary and standby units over all network interfaces and the failover cable. The default is 15 seconds. The minimum value is 3 seconds and the maximum is 15 seconds. Set the seconds to a lower value for stateful failover. With a faster poll time, the PIX Firewall can detect failure and trigger failover faster. However, faster detection may cause unnecessary switchovers when the network is temporarily congested or a network card starts slowly.

The syntax for the failover poll command is as follows:

**failover poll** ***seconds***

| | |
|---|---|
| **poll seconds** | Specifies how long failover waits before sending special failover “hello” packets between the primary and standard PIX Firewalls over all network interfaces and the failover cable. The default is 15 seconds, the minimum value is 3 seconds, and the maximum is 15 seconds. Set it to a lower value for stateful failover. With a faster poll time, the PIX Firewall can detect failure and trigger a failover faster. However, faster detection may cause unnecessary switchovers when the network is temporarily congested or a network card starts slowly. |



## *show failover* Command

### Before failover

```
pixfirewall(config)# show failover
Failover On
Cable status: Normal
Reconnect timeout 0:00:00
    This host: Primary - Active
        Active time: 360 (sec)
        Interface dmz (172.16.0.1): Normal
        Interface outside (192.168.0.2): Normal
        Interface inside (10.0.0.1): Normal
    Other host: Secondary - Standby
        Active time: 0 (sec)
        Interface dmz (172.16.0.4): Normal
        Interface outside (192.168.0.4): Normal
        Interface inside (10.0.0.4): Normal

Stateful Failover Logical Update Statistics
    Link : dmz
```

### After failover

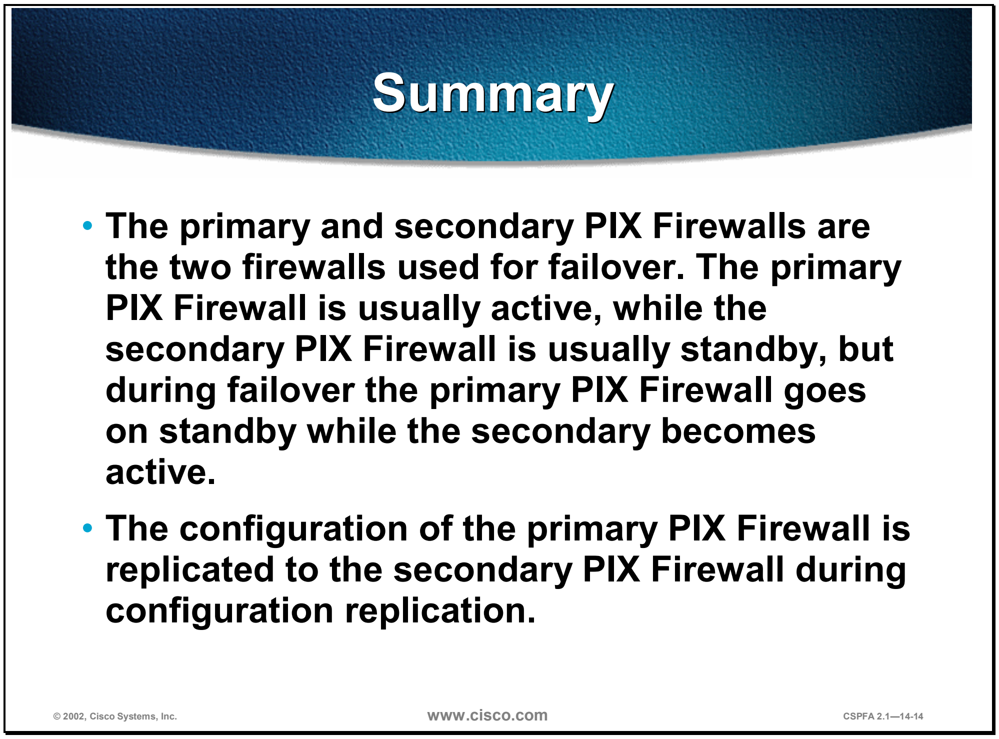
```
pixfirewall(config)# show failover
Failover On
Cable status: Normal
Reconnect timeout 0:00:00
    This host: Primary - Standby
        Active time: 0 (sec)
        Interface dmz (172.16.0.4): Normal
        Interface outside (192.168.0.4): Normal
        Interface inside (10.0.0.4): Normal
    Other host: Secondary - Active
        Active time: 150 (sec)
        Interface dmz (172.16.0.1): Normal
        Interface outside (192.168.0.2): Normal
        Interface inside (10.0.0.1): Normal

Stateful Failover Logical Update Statistics
    Link : dmz
```

© 2002, Cisco Systems, Inc. www.cisco.com CSPFA 2.1—14-12

The figure above is an example of the **show failover** command both before and after failure to the primary PIX Firewall. This example shows the primary PIX Firewall going from active mode to standby mode and the secondary PIX Firewall going from standby mode to active mode during a failover. During this process, the primary PIX Firewall swaps its system IP addresses with the secondary PIX Firewall's failover IP addresses.

# Summary

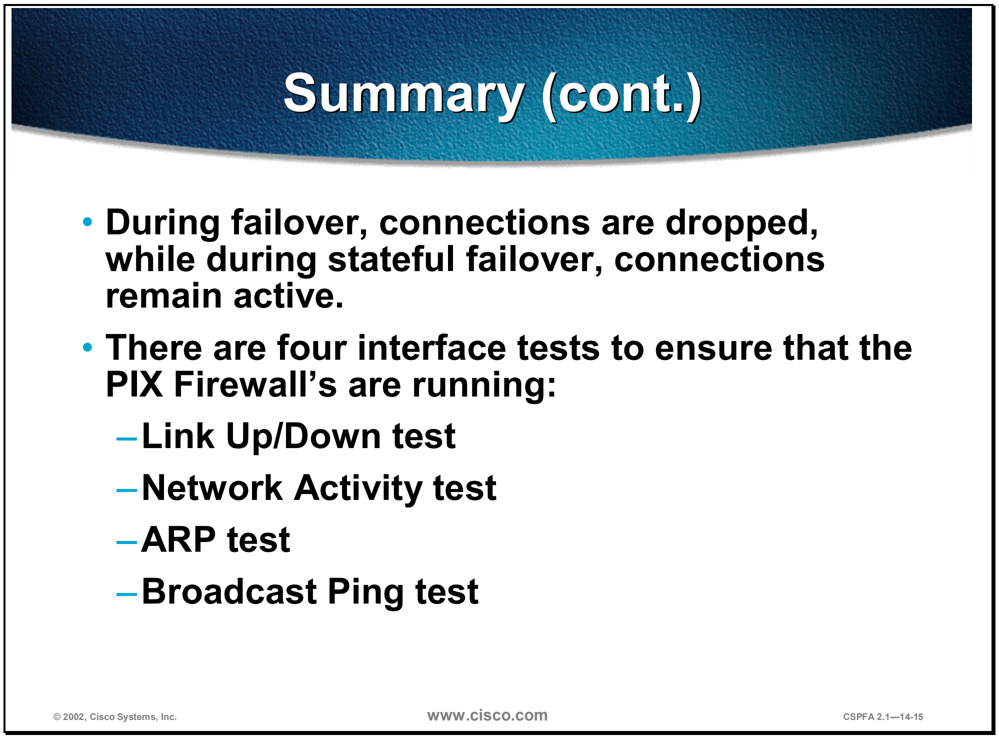
This section summarizes what you learned in this chapter.



## Summary

- The primary and secondary PIX Firewalls are the two firewalls used for failover. The primary PIX Firewall is usually active, while the secondary PIX Firewall is usually standby, but during failover the primary PIX Firewall goes on standby while the secondary becomes active.
- The configuration of the primary PIX Firewall is replicated to the secondary PIX Firewall during configuration replication.

© 2002, Cisco Systems, Inc. www.cisco.com CSPFA 2.1—14-14



## Summary (cont.)

- During failover, connections are dropped, while during stateful failover, connections remain active.
- There are four interface tests to ensure that the PIX Firewall’s are running:
  - Link Up/Down test
  - Network Activity test
  - ARP test
  - Broadcast Ping test

© 2002, Cisco Systems, Inc. www.cisco.com CSPFA 2.1—14-15

# Lab Exercise—Configure Failover

Complete the following lab exercise to practice what you learned in this chapter.

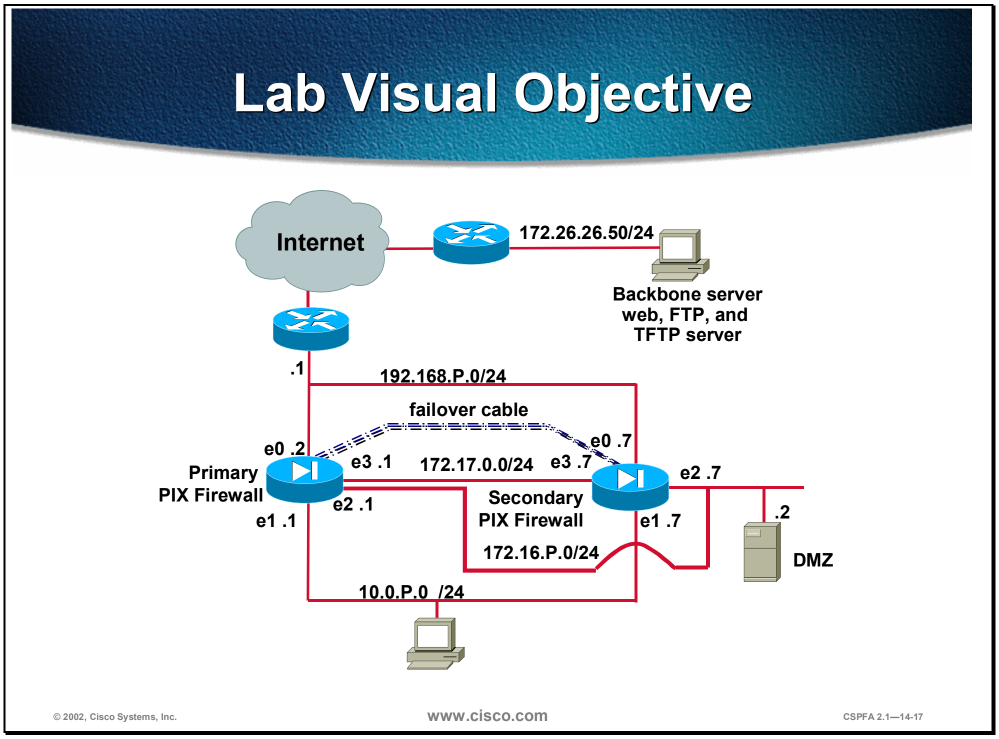
## Objectives

In this lab exercise you will complete the following tasks:

- Configure the primary PIX Firewall for failover to the secondary PIX Firewall.
- Make the primary PIX Firewall active.
- Configure the primary PIX Firewall for stateful failover.

## Visual Objectives

The following figure displays the configuration you will complete in this lab exercise.



Lab Visual Objective

© 2002, Cisco Systems, Inc. www.cisco.com CSPFA 2.1—14-17

## Task 1—Configure the Primary PIX Firewall for Failover to the Secondary PIX Firewall

Perform the following steps to configure the primary PIX Firewall for failover to the secondary PIX Firewall. Configure another interface for stateful failover, for later in Task 3.

**Step 1** Assign the PIX Firewall interface name (MYFAILOVER) and security level (55).

```
pixP(config)# nameif e3 MYFAILOVER security55
```

**Step 2** Enable the interface for an Intel full duplex.

```
pixP(config)# interface e3 100full
```

**Step 3** Assign an IP address to the interface.

```
pixP(config)# ip address MYFAILOVER 172.17.P.1 255.255.255.0
```

(where P = pod number)

**Step 4** Specify the clock time using the **clock set** command:

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

**Step 5** Save all changes to Flash memory.

```
pixP(config)# write memory
```

**Step 6** Test connections to 172.26.26.50 by using FTP and HTTP connections.

**Step 7** Make sure that you are connected to the primary PIX Firewall. Enter the **failover** command to enable failover.

```
pixP(config)# failover
```

**Step 8** Make sure that the primary PIX Firewall is enabled by using the **show failover** command.

```
pixP(config)# show failover
```

**Step 9** Enter the **failover ip address** command to configure the primary PIX Firewall with the secondary PIX Firewall IP addresses for each interface that is being used.

```
pixP(config)# failover ip address outside 192.168.P.7
pixP(config)# failover ip address inside 10.0.P.7
pixP(config)# failover ip address dmz 172.16.P.7
pixP(config)# failover ip address MYFAILOVER 172.17.P.7
```

(where P = pod number)

**Step 10** Write the configuration to Flash memory.

```
pixP(config)# write memory
```

**Step 11** Connect the failover cable to the primary PIX Firewall making sure to use the primary end of the cable.

**Step 12** Connect the other end of the failover cable marked *Secondary* to the secondary PIX Firewall.

**Step 13** Power up the secondary PIX Firewall so that the replication of information from the primary PIX Firewall to the secondary PIX Firewall can occur.

**Step 14** After the secondary PIX Firewall is operational, enter the show failover command on the primary PIX Firewall to make sure that the replication is complete and that communication between the PIX Firewalls is working.

```
pixfirewall(config)# show failover
Failover on
Cable status: Normal
Reconnect timeout 0:00:00
         This host: Primary - Active
                  Active time: 7350 (sec)
                  Interface intf5 (127.0.0.1): Link Down (Waiting)
                  Interface intf4 (127.0.0.1): Link Down (Waiting)
                  Interface MYFAILOVER (172.17.P.1): Normal
                  Interface dmz (172.16.P.1): Normal
                  Interface outside (192.168.P.2): Normal
                  Interface inside (10.0.P.1): Normal
         Other host: Secondary - Standby
                  Active time: 0 (sec)
                  Interface intf5 (0.0.0.1): Link Down (Waiting)
                  Interface intf4 (0.0.0.1): Link Down (Waiting)
                  Interface MYFAILOVER (172.17.P.7): Normal
                  Interface dmz (172.16.P.7): Normal
                  Interface outside (192.168.P.7): Normal
                  Interface inside (10.0.P.7): Normal
```

(where P = pod number)

**Step 15** Test connections to 172.26.26.50 by using FTP.

**Step 16** Log back in to your FTP server.

**Step 17** Reload the primary PIX Firewall to test failover. This ensures that the active PIX Firewall has switched from the primary PIX Firewall to the secondary PIX Firewall.

```
pixP(config)# reload
```

**Step 18** When asked to confirm the reload, press Enter.

**Step 19** To verify that the secondary PIX Firewall is active, enter the show failover command.

```
pixP(config)# show failover
Failover On
Cable status: Normal
Reconnect time 0:00:00
        This host: Primary - Standby
                Active time: 0 (sec)
                Interface intf5 (127.0.0.1): Link Down (Waiting)
                Interface intf4 (127.0.0.1): Link Down (Waiting)
                Interface MYFAILOVER (172.17.P.7): Normal
                Interface dmz (172.16.P.7): Normal
                Interface outside (192.168.P.7): Normal
                Interface inside (10.0.P.7): Normal
        Other host: Secondary - Active
                  Active time: 7350 (sec)
                  Interface intf5 (0.0.0.0): Link Down (Waiting)
                  Interface intf4 (0.0.0.0): Link Down (Waiting)
                Interface MYFAILOVER (172.17.P.1): Normal
                Interface dmz (172.16.P.1): Normal
                Interface outside (192.168.P.2): Normal
                Interface inside (10.0.P.1): Normal
```

**Step 20** Return to your Windows NT command prompt and do a directory listing. You should receive the message "Connection closed by remote host."

```
ftp> dir
```

**Step 21** Quit the ftp session.

```
ftp> quit
```

## Task 2—Make the Primary PIX Firewall Active

Perform the following lab steps to make the primary PIX Firewall the active PIX Firewall:

**Step 1** Make the primary PIX Firewall the active PIX Firewall by using the **failover active** command. Make sure that you are connected to the primary PIX Firewall's console port.

```
pixP(config)# failover active
```

**Step 2** Verify that the failover active command worked by using the **show failover** command. The primary PIX Firewall should show that it is in active mode and the secondary PIX Firewall should show that it is in the standby mode.

```
pixP(config)# show failover
Failover On
Cable status: Normal
Reconnect timeout 0:00:00
        This host: Primary - Active
                Active time: 7350 (sec)
                Interface intf5 (127.0.0.1): Link Down (Waiting)
                Interface intf4 (127.0.0.1): Link Down (Waiting)
                Interface MYFAILOVER (172.17.P.1): Normal
                Interface dmz (172.16.P.1): Normal
                Interface outside (192.168.P.2): Normal
                Interface inside (10.0.P.1): Normal
        Other host: Secondary - Standby
                Active time: 7300 (sec)
                Interface intf5 (0.0.0.0): Link Down (Waiting)
                Interface intf4 (0.0.0.0): Link Down (Waiting)
                Interface MYFAILOVER (172.17.P.7): Normal
                Interface dmz (172.16.P.7): Normal
                Interface outside (192.168.P.7): Normal
                Interface inside (10.0.P.7): Normal
```

(where P = pod number)

## Task 3—Configure the Primary PIX Firewall for Stateful Failover

Perform the following lab steps to configure the primary PIX Firewall for stateful failover:

**Step 1** Change the failover poll time to 3 seconds so the PIX Firewall will trigger failover faster.

```
pixP(config)# failover poll 3
```

**Step 2** Configure the primary PIX Firewall for stateful failover to the secondary PIX Firewall by using the failover link command.

```
pixP(config)# failover link MYFAILOVER
```

**Step 3** Save your changes to flash memory.

```
pixP(config)# write memory
```

**Step 4** Make sure that the secondary PIX Firewall has the latest changes to the configuration by using the **write standby** command. This will sync up the configuration on both firewalls.

```
pixP(config)# write standby
```

**Step 5** Go to your web browser and ftp to 172.26.26.50.

**Step 6** To download a zip file from the FTP server, double-click the **getme.zip** file.

**Step 7** Select **Open this file from its current location** in the File Download group box.

**Step 8** Click **OK**.

**Step 9** Start a continuous ping to 172.26.26.50.

```
C:\ ping 172.26.26.50 -t
```

**Step 10** Reload the primary PIX Firewall.

```
pixP(config)# reload
```

**Step 11** When asked to confirm the reload, press **Enter**.

**Step 12** To verify that stateful failover is working, observe the **ftp** transfer and the continuous ping. Both should still be active.

**Step 13** Make the primary PIX Firewall the active PIX Firewall by using the **failover active** command.

```
pixP(config)# failover active
```

**Step 14** Verify that the **failover active** command worked by using the **show failover** command. The output should reveal that the primary PIX Firewall is in active mode and the secondary PIX Firewall is in standby mode.

15

# Virtual Private Network Configuration

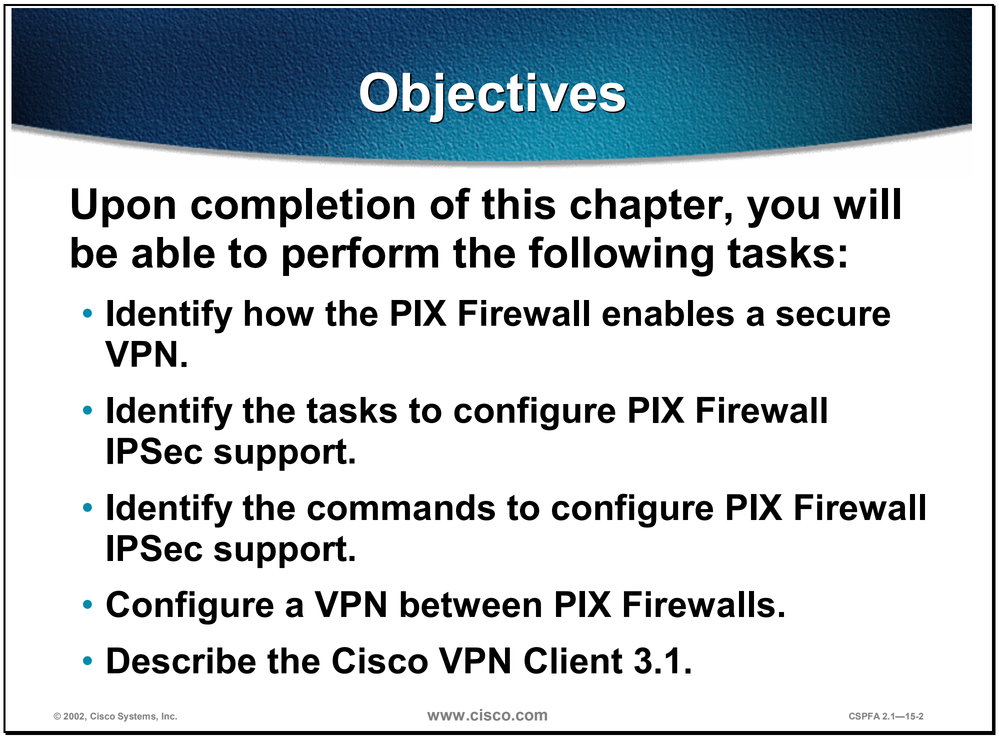
## Overview

This chapter includes the following topics:

- Objectives
- The PIX Firewall enables a secure VPN
- IPSec configuration tasks
- Task 1—Prepare to configure VPN support
- Task 2—Configure IKE parameters
- Task 3—Configure IPSec parameters
- Task 4—Test and verify VPN configuration
- The Cisco VPN Client 3.1
- Scale PIX Firewall VPNs
- Summary
- Lab exercise

# Objectives

This section lists the chapter's objectives.



## Objectives

**Upon completion of this chapter, you will be able to perform the following tasks:**

- **Identify how the PIX Firewall enables a secure VPN.**
- **Identify the tasks to configure PIX Firewall IPSec support.**
- **Identify the commands to configure PIX Firewall IPSec support.**
- **Configure a VPN between PIX Firewalls.**
- **Describe the Cisco VPN Client 3.1.**

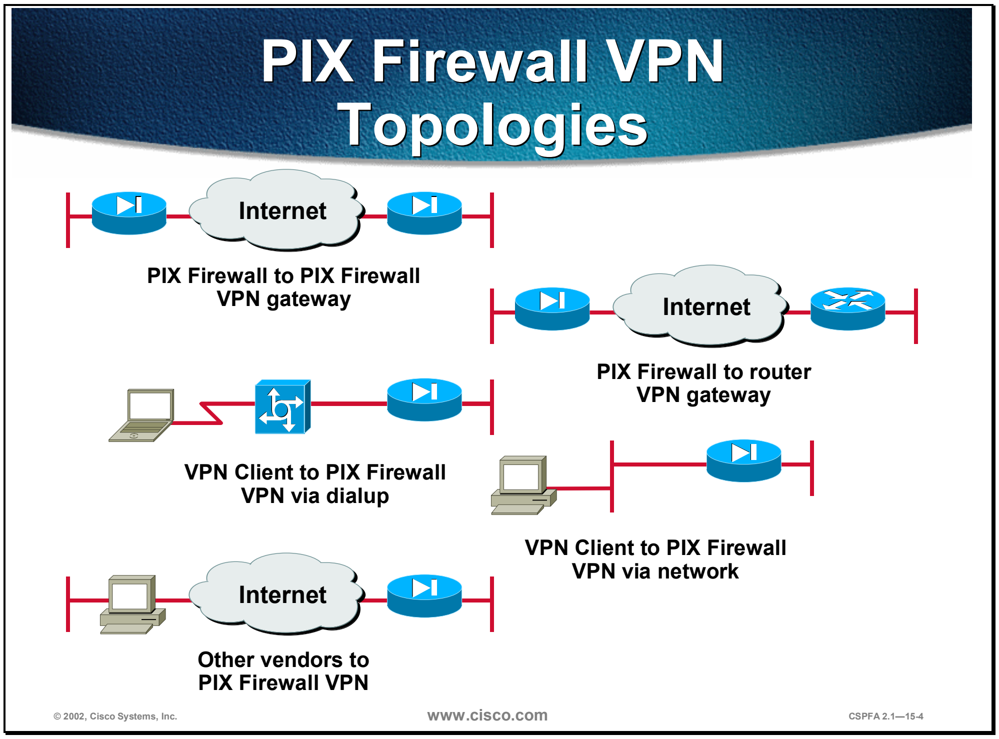
© 2002, Cisco Systems, Inc. www.cisco.com CSPFA 2.1—15-2

# The PIX Firewall Enables a Secure VPN

A virtual private network (VPN) is a service offering secure, reliable connectivity over a shared, public network infrastructure such as the Internet. Because the infrastructure is shared, connectivity can be provided at lower cost than existing dedicated private networks.

The Cisco PIX™ Firewall is a powerful enabler of VPN services. The PIX Firewall's high performance, conformance to open standards, and ease of configuration make it a versatile VPN gateway.

The PIX Firewall VPN chapter covers the basics of IPSec and PIX Firewall VPNs with a focus on PIX Firewall gateway to PIX Firewall gateway communications.



PIX Firewall VPN Topologies

PIX Firewall to PIX Firewall VPN gateway

PIX Firewall to router VPN gateway

VPN Client to PIX Firewall VPN via dialup

VPN Client to PIX Firewall VPN via network

Other vendors to PIX Firewall VPN

© 2002, Cisco Systems, Inc. www.cisco.com CSPFA 2.1—15-4

The PIX Firewall enables VPNs in several topologies, as illustrated in the figure:

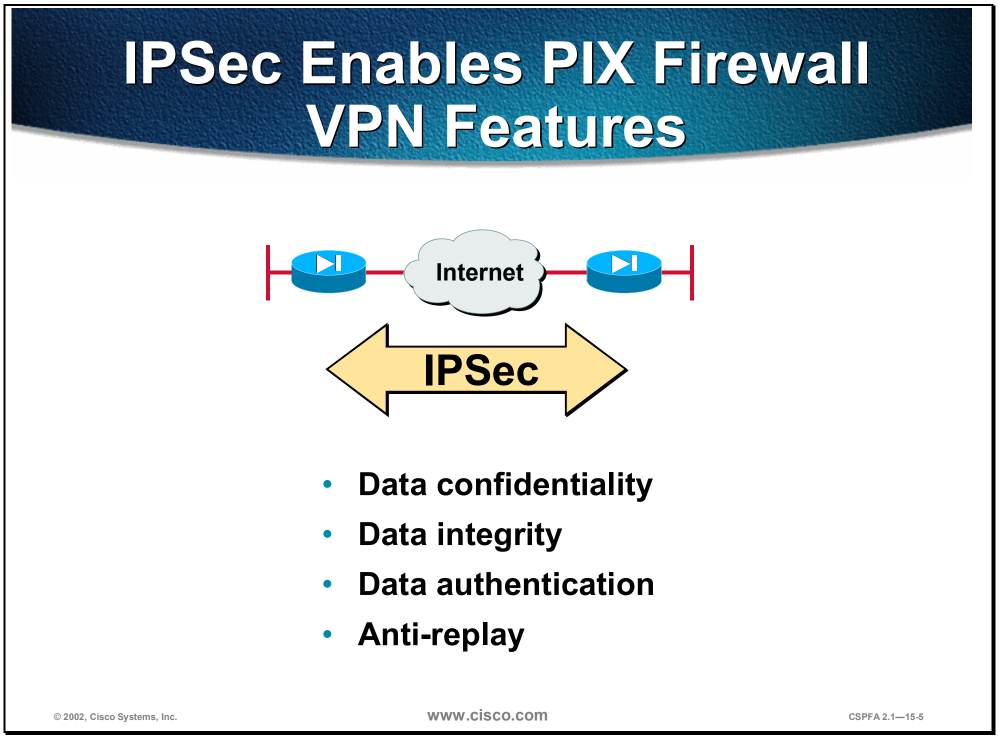
- PIX Firewall to PIX Firewall secure VPN gateway—Two or more PIX Firewalls can enable a VPN, which secures traffic from devices behind the PIX Firewalls. The secure VPN gateway topology prevents the user from having to implement VPN devices or software inside the network, making the secure gateway transparent to users.
- PIX Firewall to Cisco IOS™ router secure VPN gateway—The PIX Firewall and Cisco router, running CVPN software, can interoperate to create a secure VPN gateway between networks.
- CVPN Client to PIX Firewall via dialup—The PIX Firewall can become a VPN endpoint for the CVPN Client over a dialup network. The dialup network can consist of ISDN, Public Switched Telephone Network (PSTN) (analog modem), or digital subscriber line (DSL) communication channels.

- CVPN Client to PIX Firewall via network—The PIX Firewall can become a VPN endpoint for the CVPN Client over an IP network.
- Other vendor products to PIX Firewall—Products from other vendors can connect to the PIX Firewall if they conform to open VPN standards.

A VPN itself can be constructed in a number of scenarios. The most common are as follows:

- Internet VPN—A private communications channel over the public access Internet. This type of VPN can be divided into the following:
  - Connecting remote offices across the Internet
  - Connecting remote dial users to their home gateway via an Internet Service Provider (ISP) (sometimes called a Virtual Private Dial Network or VPDN)
- Intranet VPN—A private communication channel within an enterprise or organization that may or may not involve traffic traversing a WAN.
- Extranet VPN—A private communication channel between two or more separate entities that may involve data traversing the Internet or some other WAN.

In all cases the VPN or tunnel consists of two endpoints that may be represented by PIX Firewalls, Cisco routers, individual client workstations running the CVPN Client, or other vendors' VPN products that conform to open standards.



## IPSec Enables PIX Firewall VPN Features

- Data confidentiality
- Data integrity
- Data authentication
- Anti-replay

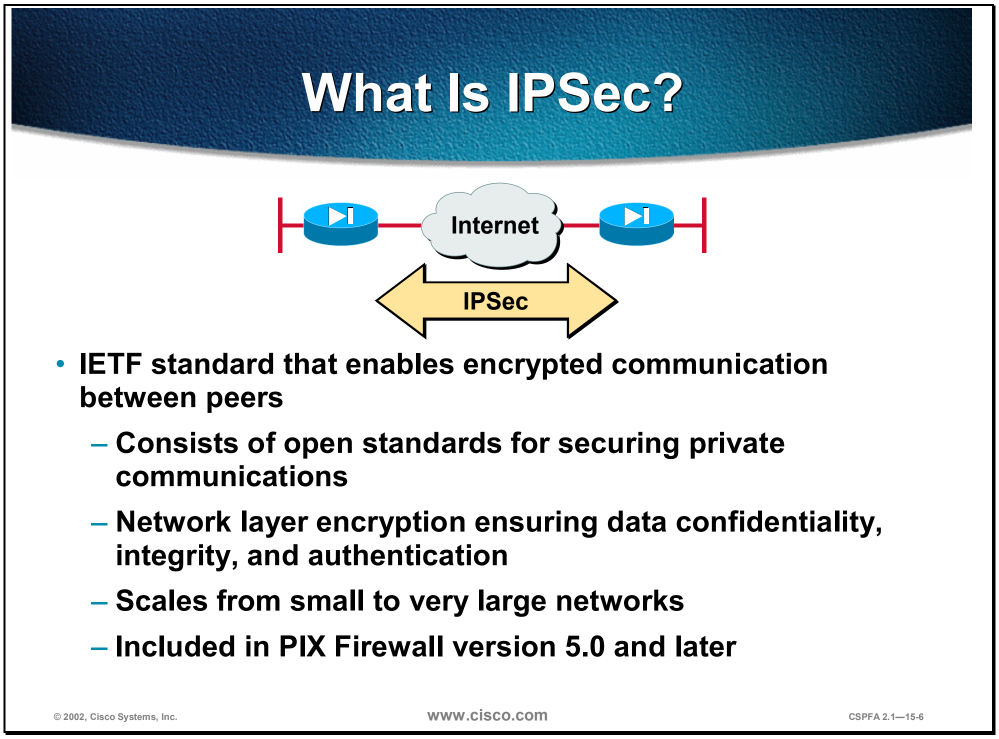
© 2002, Cisco Systems, Inc. www.cisco.com CSPFA 2.1—15-5

PIX Firewall versions 5.0 and higher use the industry-standard IP Security (IPSec) protocol suite to enable advanced VPN features. The PIX Firewall IPSec implementation is based on Cisco IOS IPSec that runs in Cisco routers.

IPSec provides a mechanism for secure data transmission over IP networks, ensuring confidentiality, integrity, and authenticity of data communications over unprotected networks such as the Internet.

IPSec enables the following PIX Firewall VPN features:

- Data confidentiality—The IPSec sender can encrypt packets before transmitting them across a network.
- Data integrity—The IPSec receiver can authenticate IPSec peers and packets sent by the IPSec sender to ensure that the data has not been altered during transmission.
- Data origin authentication—The IPSec receiver can authenticate the source of the IPSec packets sent. This service is dependent upon the data integrity service.
- Anti-replay—The IPSec receiver can detect and reject replayed packets, helping prevent spoofing and man-in-the-middle attacks.



Internet
IPSec

# What Is IPSec?

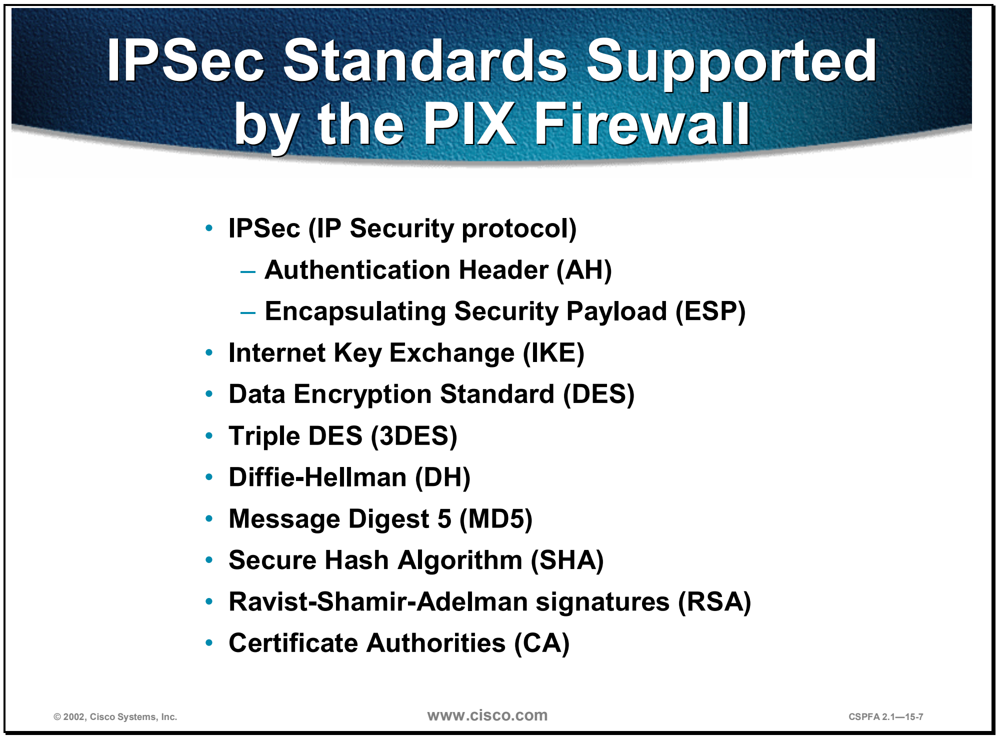
- IETF standard that enables encrypted communication between peers
  - Consists of open standards for securing private communications
  - Network layer encryption ensuring data confidentiality, integrity, and authentication
  - Scales from small to very large networks
  - Included in PIX Firewall version 5.0 and later

© 2002, Cisco Systems, Inc. www.cisco.com CSPFA 2.1—15-6

The PIX Firewall uses the open IPSec protocol to enable secure VPNs. IPSec is a set of security protocols and algorithms used to secure data at the network layer. IPSec and related security protocols conform to open standards promulgated by the Internet Engineering Task Force (IETF) and documented RFCs and IETF-draft papers.

IPSec acts at the network layer, protecting and authenticating IP packets between a PIX Firewall and other participating IPSec devices (peers), such as PIX Firewalls, Cisco routers, the CVPN Client, and other IPSec-compliant products.

IPSec can be used to scale from small to very large networks. It is included in PIX Firewall version 5.0 and later.



## IPSec Standards Supported by the PIX Firewall

- IPSec (IP Security protocol)
  - Authentication Header (AH)
  - Encapsulating Security Payload (ESP)
- Internet Key Exchange (IKE)
- Data Encryption Standard (DES)
- Triple DES (3DES)
- Diffie-Hellman (DH)
- Message Digest 5 (MD5)
- Secure Hash Algorithm (SHA)
- Ravist-Shamir-Adelman signatures (RSA)
- Certificate Authorities (CA)

© 2002, Cisco Systems, Inc. www.cisco.com CSPFA 2.1—15-7

The PIX Firewall supports the following IPSec and related standards:

- IPSec (IP Security protocol)
- Internet Key Exchange (IKE)
- Data Encryption Standard (DES)
- Triple DES (3DES)
- Diffie-Hellman (DH)
- Message Digest 5 (MD5)
- Secure Hash Algorithm-1 (SHA-1)
- Rivet, Shamir, and Adelman signatures (RSA)
- Certificate Authorities (CA)

## IPSec

IPSec is a framework of open standards that provides data confidentiality, data integrity, and data authentication between participating peers at the IP layer. IPSec can be used to protect one or more data flows between IPSec peers. IPSec is documented in a series of Internet RFCs, all available at http://www.ietf.org/html.charters/ipsec-charter.html. The overall IPSec implementation is guided by “Security Architecture for the Internet Protocol,” RFC 2401. IPSec consists of the following two main protocols:

- Authentication Header (AH)—A security protocol that provides authentication and optional replay-detection services. AH acts as a “digital signature” to ensure that tampering has not occurred with the data in the IP packet. AH does not provide data encryption and decryption services. AH can be used either by itself or with Encapsulating Security Payload (ESP).

- Encapsulating Security Payload (ESP)—A security protocol that provides data confidentiality and protection with optional authentication and replay-detection services. The PIX Firewall uses ESP to encrypt the data payload of IP packets. ESP can be used either by itself or in conjunction with AH.

## IKE

IKE is a hybrid protocol that provides utility services for IPSec: authentication of the IPSec peers, negotiation of IKE and IPSec security associations (SAs), and establishment of keys for encryption algorithms used by IPSec. IKE is synonymous with ISAKMP in PIX Firewall configuration.

## SA

The concept of an SA is fundamental to IPSec. An SA is a connection between IPSec peers that determines the IPSec services available between the peers, similar to a TCP or UDP port. Each IPSec peer maintains an SA database in memory containing SA parameters. SAs are uniquely identified by the IPSec peer address, security protocol, and security parameter index (SPI). You will need to configure SA parameters and monitor SAs on the PIX Firewall.

## DES

DES is used to encrypt and decrypt packet data. DES is used by both IPSec and IKE. DES uses a 56-bit key, ensuring high performance encryption.

## 3DES

3DES is a variant of DES, which iterates three times with three separate keys, effectively doubling the strength of DES. 3DES is used by IPSec to encrypt and decrypt data traffic. 3DES uses a 168-bit key, ensuring strong encryption.

## D-H

DH is a public-key cryptography protocol. It enables two parties to establish a shared secret key over an insecure communications channel. D-H is used within IKE to establish session keys. 768-bit and 1024-bit D-H groups are supported in the PIX Firewall. The 1024-bit group is more secure.

## MD5

MD5 is a hash algorithm used to authenticate packet data. The PIX Firewall uses the MD5 hashed message authentication code (HMAC) variant, which provides an additional level of hashing. A hash is a one-way encryption algorithm that takes an input message of arbitrary length and produces a fixed-length output message. IKE, AH, and ESP use MD5 for authentication.

## SHA-1

SHA is a hash algorithm used to authenticate packet data. The PIX Firewall uses the SHA-1 HMAC variant, which provides an additional level of hashing. IKE, AH, and ESP use SHA-1 for authentication.

## RSA Signatures

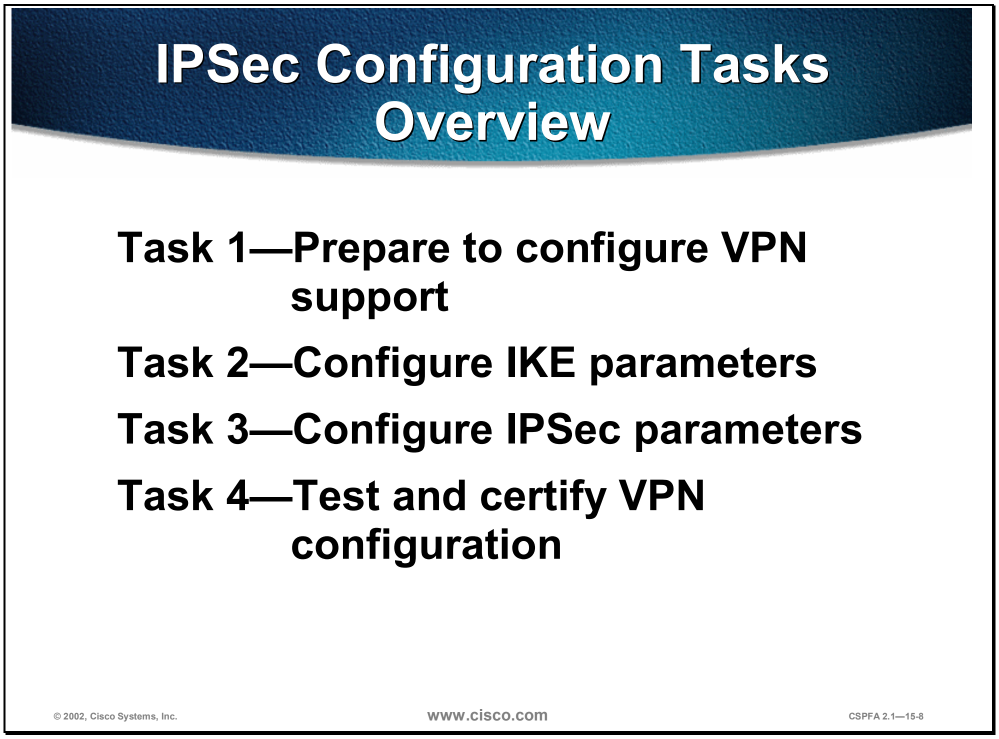
RSA is a public-key cryptographic system used for authentication. IKE on the PIX Firewall uses a D-H exchange to determine secret keys on each IPSec peer used by encryption algorithms. The D-H exchange can be authenticated with RSA (or pre-shared keys).

## CA

The CA support of the PIX Firewall enables the IPSec-protected network to scale by providing the equivalent of a digital identification card to each device. When two IPSec peers wish to communicate, they exchange digital certificates to prove their identities (thus removing the need to manually exchange public keys with each peer or to manually specify a shared key at each peer). The digital certificates are obtained from a CA. CA support on the PIX Firewall uses RSA signatures to authenticate the CA exchange.

# IPSec Configuration Tasks

This section describes the tasks you perform when configuring an IPSec-based VPN.



## IPSec Configuration Tasks Overview

Task 1—Prepare to configure VPN support

Task 2—Configure IKE parameters

Task 3—Configure IPSec parameters

Task 4—Test and certify VPN configuration

© 2002, Cisco Systems, Inc. www.cisco.com CSPFA 2.1—15-8

The rest of this chapter demonstrates how to configure an IPSec-based VPN between two PIX Firewalls operating as secure gateways, using pre-shared keys for authentication. The IPSec configuration process can be summed up as two major tasks: configuring an IPSec encryption policy, and applying the policy to an interface.

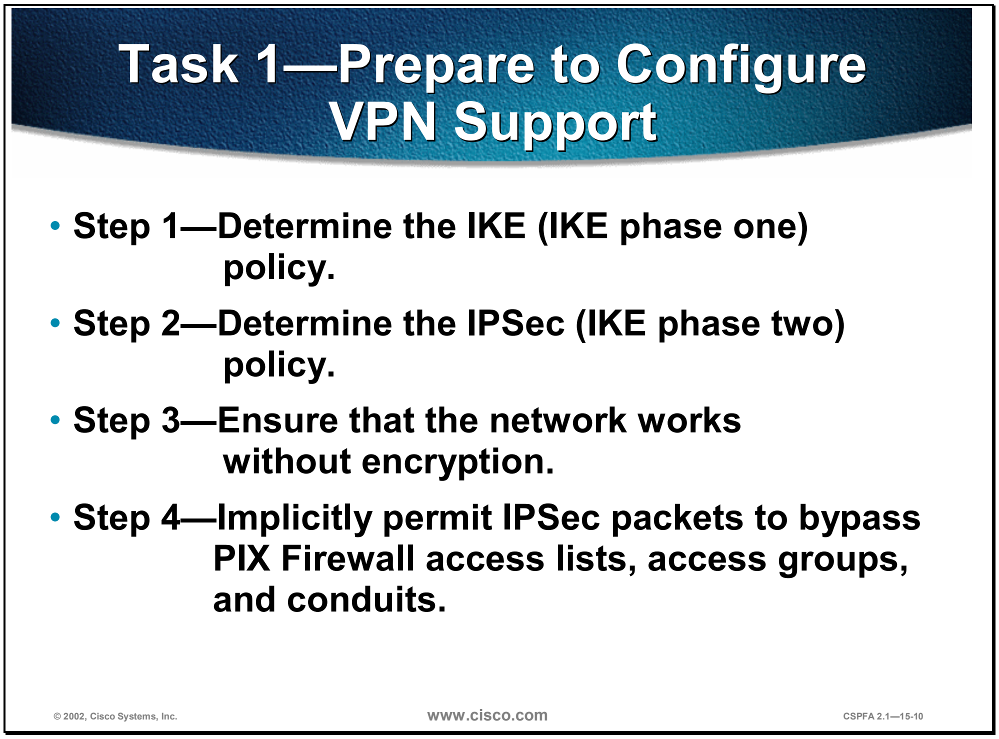
The four overall tasks used to configure IPSec encryption on the PIX Firewall are summarized below. Subsequent sections of this chapter discuss each configuration task in greater detail. The following are the four tasks:

- Task 1—Prepare to configure VPN support. This task consists of several steps that determine IPSec policies, ensure that the network works, and ensure that the PIX Firewall can support IPSec.
- Task 2—Configure IKE parameters. This task consists of several configuration steps that ensure that IKE can set up secure channels to desired IPSec peers. IKE can set up IPSec SAs, enabling IPSec sessions. IKE negotiates IKE parameters and sets up IKE SAs during an IKE phase one exchange called "main mode."
- Task 3—Configure IPSec parameters. This task consists of several configuration steps that specify IPSec SA parameters between peers, and set global IPSec values. IKE negotiates SA parameters and sets up IPSec SAs during an IKE phase two exchange called "quick mode."

- Task 4—Test and verify VPN configuration. After you configure IPSec, you will need to verify that you have configured it correctly, and ensure that it works.

# Task 1—Prepare to Configure VPN Support

Successful implementation of an IPSec network requires advance preparation before beginning the configuration of individual PIX Firewalls. This section outlines how to determine network design details.



## Task 1—Prepare to Configure VPN Support

- Step 1—Determine the IKE (IKE phase one) policy.
- Step 2—Determine the IPSec (IKE phase two) policy.
- Step 3—Ensure that the network works without encryption.
- Step 4—Implicitly permit IPSec packets to bypass PIX Firewall access lists, access groups, and conduits.

© 2002, Cisco Systems, Inc. www.cisco.com CSPFA 2.1—15-10

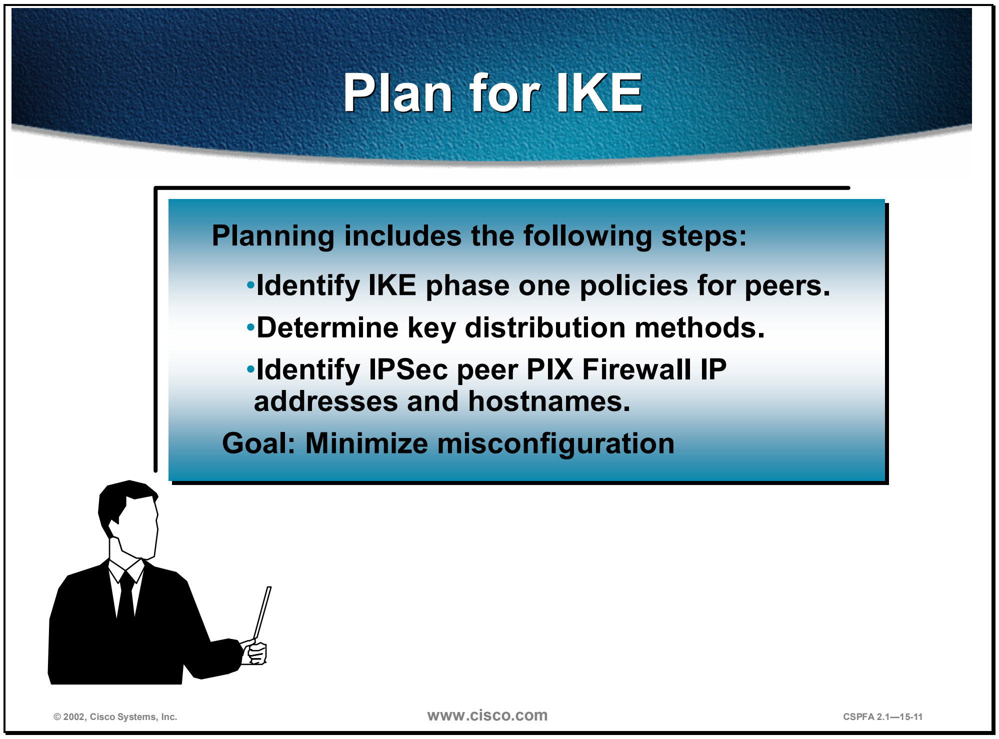
Configuring IPSec encryption can be complicated. You must plan in advance if you want to configure IPSec encryption correctly the first time and minimize misconfiguration. You should begin this task by defining the overall security needs and strategy based on the overall company security policy. Some planning steps include the following:

**Step 1** Determine the IKE (IKE phase one) policy—Determine the IKE policies between peers based on the number and location of IPSec peers.

**Step 2** Determine the IPSec (IKE phase two) policy—You need to identify IPSec peer details such as IP addresses and IPSec modes. You then configure crypto maps to gather all IPSec policy details together.

**Step 3** Ensure that the network works without encryption—Ensure that basic connectivity has been achieved between IPSec peers using the desired IP services before configuring PIX Firewall IPSec.

**Step 4** Implicitly permit IPSec packets to bypass PIX Firewall Access Control Lists (ACLs), access groups, and conduits—In this step you enter the sysopt connection permit-ipsec command.



## Plan for IKE

Planning includes the following steps:

- Identify IKE phase one policies for peers.
- Determine key distribution methods.
- Identify IPSec peer PIX Firewall IP addresses and hostnames.

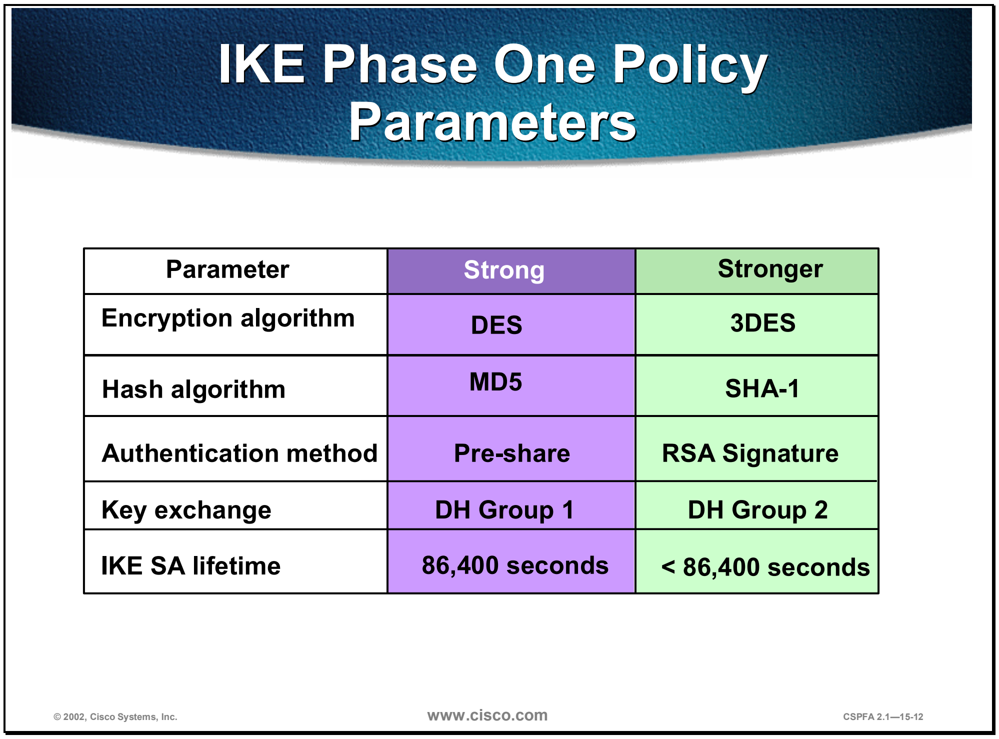
Goal: Minimize misconfiguration

© 2002, Cisco Systems, Inc. www.cisco.com CSPFA 2.1—15-11

An IKE policy defines a combination of security parameters to be used during the IKE negotiation. You should determine the IKE policy, and then configure it. Some planning steps include:

- Determine IKE phase one (ISAKMP) policies for peers—An IKE policy defines a combination of security parameters to be used during the IKE negotiation. Each IKE negotiation begins by each peer agreeing on a common (shared) IKE policy. The IKE policy suites must be determined in advance of configuration.
- Determine key distribution methods based on the numbers and locations of IPSec peers—You may wish to use a CA server to support scalability of IPSec peers. You must then configure IKE to support the selected key distribution method.
- Identify IPSec peer router IP addresses and hostnames—You need to determine the details of all of the IPSec peers that will use IKE for establishing SAs.

The goal of advance planning is to minimize misconfiguration.



## IKE Phase One Policy Parameters

| Parameter | Strong | Stronger |
|---|---|---|
| Encryption algorithm | DES | 3DES |
| Hash algorithm | MD5 | SHA-1 |
| Authentication method | Pre-share | RSA Signature |
| Key exchange | DH Group 1 | DH Group 2 |
| IKE SA lifetime | 86,400 seconds | < 86,400 seconds |

© 2002, Cisco Systems, Inc. www.cisco.com CSPFA 2.1—15-12

An IKE policy defines a combination of security parameters to be used during the IKE negotiation. A group of policies makes up a “protection suite” of multiple policies that enable IPSec peers to establish IKE sessions and SAs with a minimum of configuration.

## Create IKE Policies for a Purpose

IKE negotiations must be protected, so each IKE negotiation begins by each peer agreeing on a common (shared) IKE policy. This policy states which security parameters will be used to protect subsequent IKE negotiations.

After the two peers agree upon a policy, an SA established at each peer identifies the security parameters of the policy. These SAs apply to all subsequent IKE traffic during the negotiation.

You can create multiple, prioritized policies at each peer to ensure that at least one policy will match a remote peer’s policy.

## Define IKE Policy Parameters

You can select specific values for each IKE parameter, per the IKE standard. You choose one value over another based on the security level you desire and the type of IPSec peer to which you will connect.

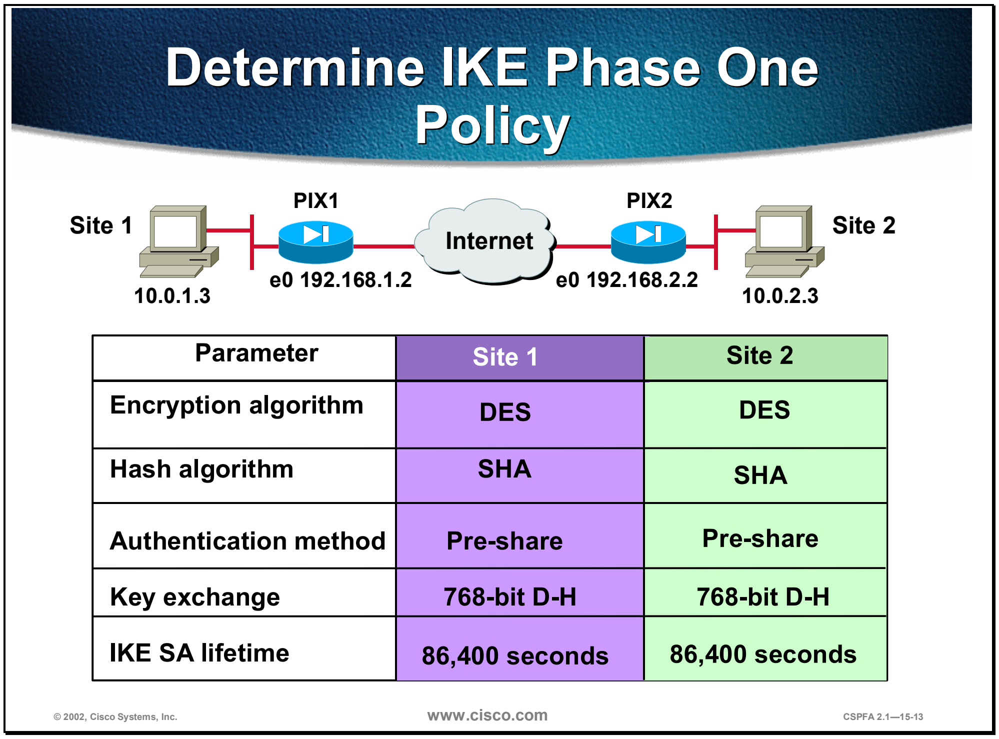
There are five parameters to define in each IKE policy, as outlined in the previous figure and in the following table. The figure shows the relative strength of each parameter, and the table shows the default values.

**IKE Policy Parameters**

| Parameter | Accepted Values | Keyword | Default |
|---|---|---|---|
| Message encryption algorithm | 56-bit DES<br>168-bit 3DES | **des**<br>**3des** | DES |
| Message integrity (hash) algorithm | SHA-1 (HMAC variant)<br>MD5 (HMAC variant) | **sha**<br>**md5** | SHA-1 |
| Peer authentication method | Pre-shared keys<br>RSA signatures | **pre-share**<br>**rsa-sig** | RSA signatures |
| Key exchange parameters (Diffie-Hellman group identifier) | 768-bit Diffie-Hellman or<br>1024-bit Diffie-Hellman | **1**<br>**2** | 768-bit Diffie-Hellman |
| ISAKMP-established security association's lifetime | Can specify any number of seconds | — | 86,400 seconds (1 day) |

**Note** 3DES provides stronger encryption than DES. Some tradeoffs of 3DES are that it takes more processing power, and it may be restricted for export or import into some countries.

**Note** RSA signatures are used with CA support, and require enrollment to a CA server.



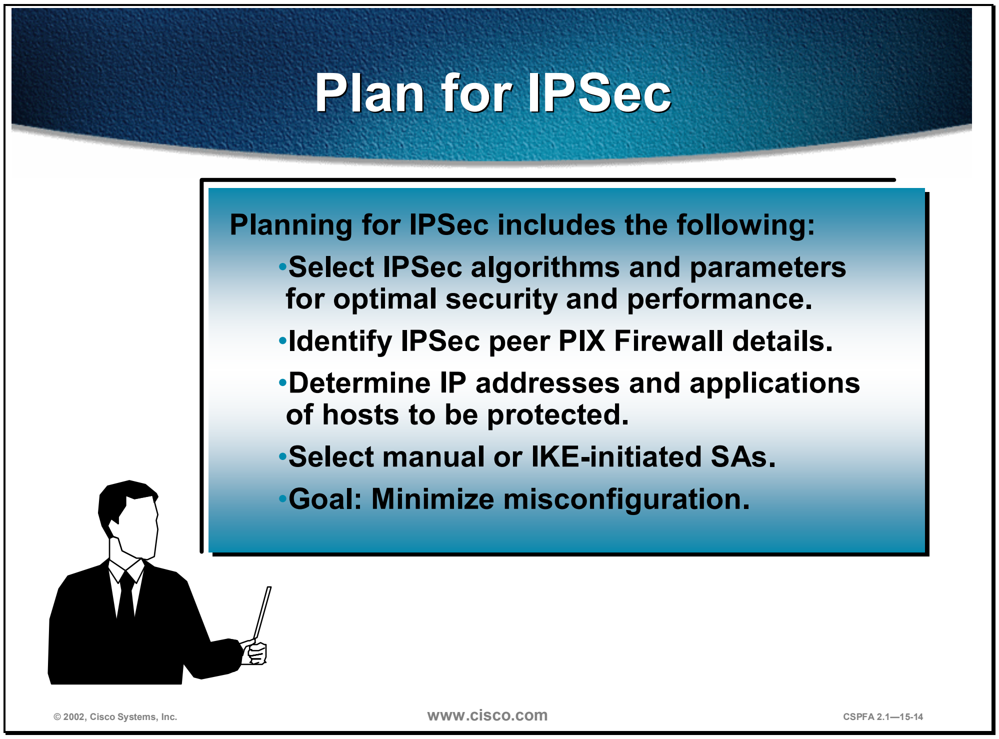
## Determine IKE Phase One Policy

| Parameter | Site 1 | Site 2 |
|---|---|---|
| Encryption algorithm | DES | DES |
| Hash algorithm | SHA | SHA |
| Authentication method | Pre-share | Pre-share |
| Key exchange | 768-bit D-H | 768-bit D-H |
| IKE SA lifetime | 86,400 seconds | 86,400 seconds |

© 2002, Cisco Systems, Inc. www.cisco.com CSPFA 2.1—15-13

An IKE policy defines a combination of security parameters to be used during the IKE negotiation. A group of policies makes up a “protection suite” of multiple policies that enable IPSec peers to establish IKE sessions and SAs with a minimum of configuration.

You should determine IKE policy details for each IPSec peer before configuring IKE. The figure shows a summary of some IKE policy details that will be configured in the examples in this chapter.



## Plan for IPSec

Planning for IPSec includes the following:

- Select IPSec algorithms and parameters for optimal security and performance.
- Identify IPSec peer PIX Firewall details.
- Determine IP addresses and applications of hosts to be protected.
- Select manual or IKE-initiated SAs.
- Goal: Minimize misconfiguration.

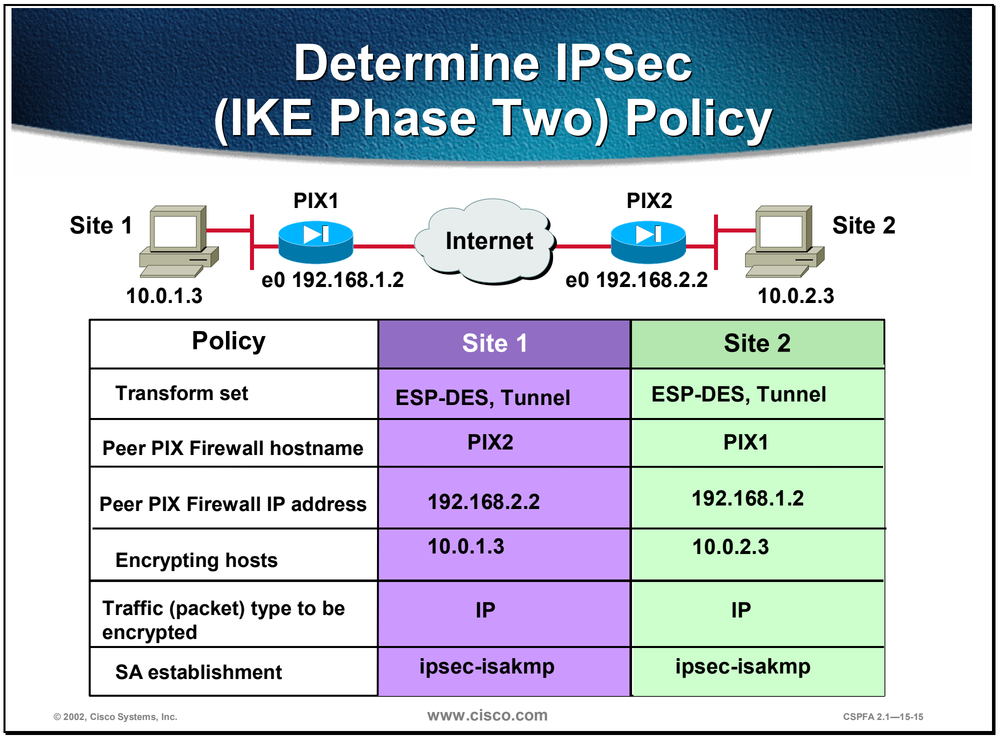
© 2002, Cisco Systems, Inc. www.cisco.com CSPFA 2.1—15-14

Planning for IPSec (IKE phase two) is another important step you should complete before actually configuring the PIX Firewall. Items to determine at this stage include the following:

- Select IPSec algorithms and parameters for optimal security and performance. You should determine what type of IPSec security will be used to secure interesting traffic. Some IPSec parameters require you to make tradeoffs between high performance and stronger security.
- Identify IPSec peer details. You must identify the IP addresses and hostnames of all IPSec peers you will connect to.
- Determine IP addresses and applications of hosts to be protected at the local peer and remote peer.
- Decide whether SAs are manually established or are established via IKE.

**Note** IPSec SAs can be configured manually, but this is not recommended because IKE is easier to configure.

The goal of this planning step is to gather the precise data you will need in later steps to minimize misconfiguration.



## Determine IPSec (IKE Phase Two) Policy

| Policy | Site 1 | Site 2 |
|---|---|---|
| Transform set | ESP-DES, Tunnel | ESP-DES, Tunnel |
| Peer PIX Firewall hostname | PIX2 | PIX1 |
| Peer PIX Firewall IP address | 192.168.2.2 | 192.168.1.2 |
| Encrypting hosts | 10.0.1.3 | 10.0.2.3 |
| Traffic (packet) type to be encrypted | IP | IP |
| SA establishment | ipsec-isakmp | ipsec-isakmp |

www.cisco.com CSPFA 2.1—15-15

Determining network design details includes defining a more detailed security policy for protecting traffic. You can then use the detailed policy to help select IPSec transform sets and modes of operation. Your security policy should answer the following questions:

- What protections are required or are acceptable for the protected traffic?
- What traffic should or should not be protected?
- Which PIX Firewall interfaces are involved in protecting internal networks, external networks, or both?
- What are the peer IPSec endpoints for the traffic?
- How should SAs be established?

The figure above shows a summary of IPSec encryption policy details that will be configured in the examples later in this chapter.

# Task 2—Configure IKE Parameters

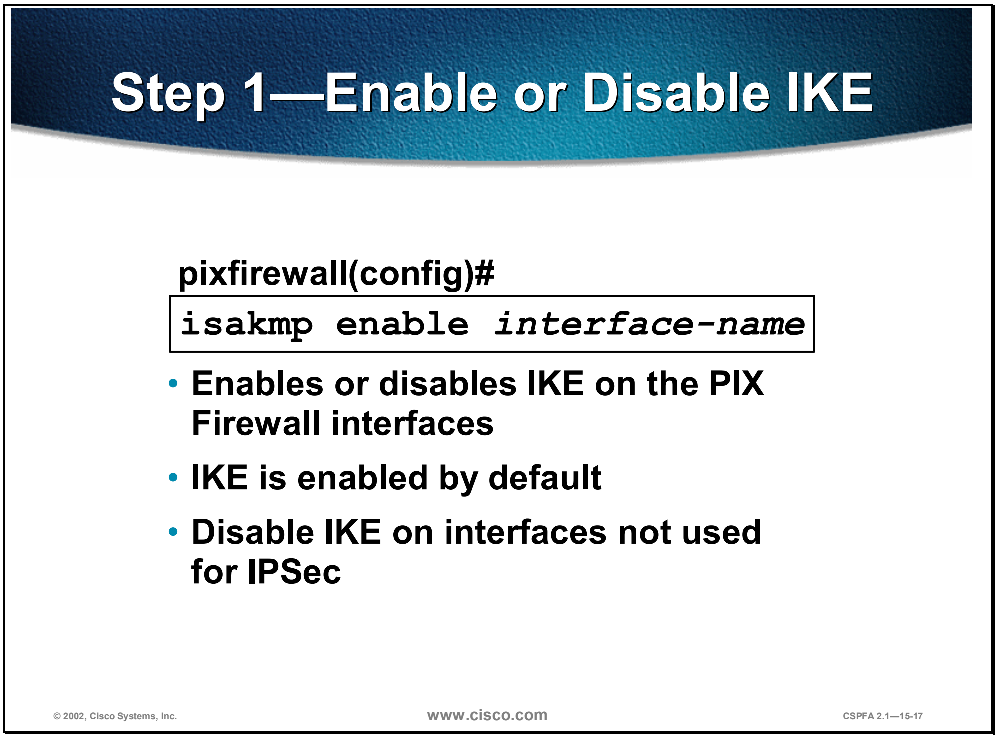
The next major task in configuring PIX Firewall IPSec is to configure IKE parameters gathered in the previous task. This section presents the steps used to configure IKE parameters for IKE pre-shared keys:

**Step 1** Enable or disable IKE.

**Step 2** Configure an IKE phase one policy.

**Step 3** Configure the IKE pre-shared key.

**Step 4** Verify IKE phase one details.



## Step 1—Enable or Disable IKE

pixfirewall(config)#

```
isakmp enable interface-name
```

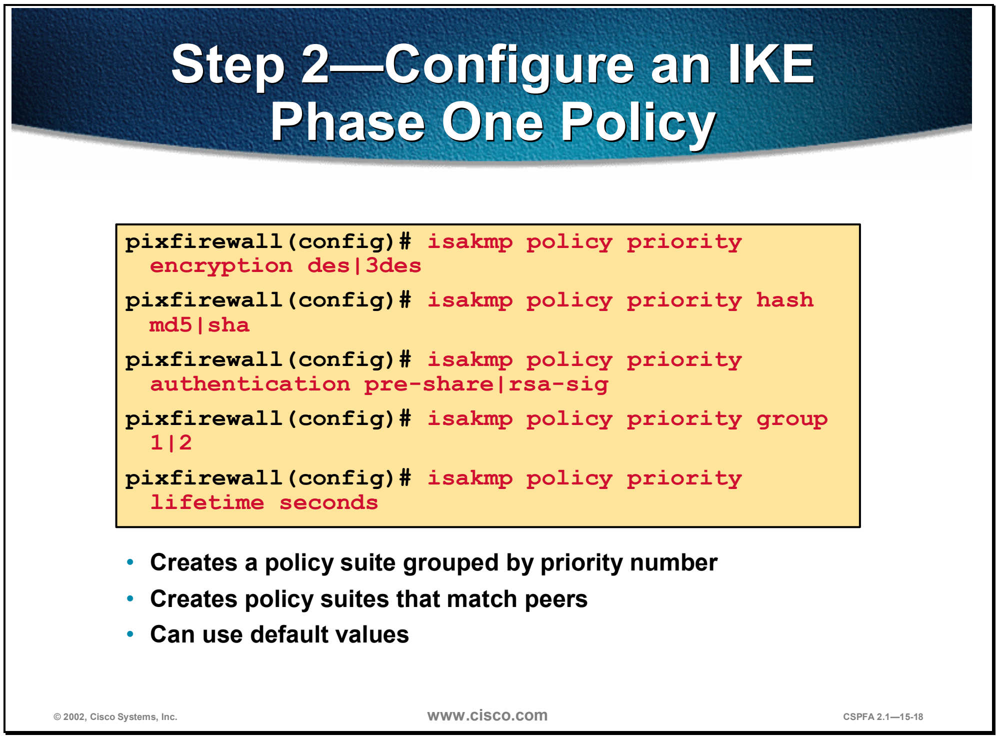
- Enables or disables IKE on the PIX Firewall interfaces
- IKE is enabled by default
- Disable IKE on interfaces not used for IPSec

© 2002, Cisco Systems, Inc. www.cisco.com CSPFA 2.1—15-17

**Step 1** Enable or disable IKE (ISAKMP) negotiation:

```
pixfirewall(config)# isakmp enable interface-name
```

Specify the PIX Firewall interface on which the IPSec peer will communicate. IKE is enabled by default and for individual PIX Firewall interfaces. Use the **no isakmp enable** ***interface-name*** command to disable IKE.



## Step 2—Configure an IKE Phase One Policy

```
pixfirewall(config)# isakmp policy priority
  encryption des|3des
pixfirewall(config)# isakmp policy priority hash
  md5|sha
pixfirewall(config)# isakmp policy priority
  authentication pre-share|rsa-sig
pixfirewall(config)# isakmp policy priority group
  1|2
pixfirewall(config)# isakmp policy priority
  lifetime seconds
```

- **Creates a policy suite grouped by priority number**
- **Creates policy suites that match peers**
- **Can use default values**

© 2002, Cisco Systems, Inc. www.cisco.com CSPFA 2.1—15-18

**Step 2** Configure an IKE Phase one policy with the **isakmp policy** command to match expected IPSec peers:

(a) Identify the policy with a unique priority number:

```
pixfirewall(config)# isakmp policy priority
```

(b) Specify the encryption algorithm (the default is des):

```
pixfirewall(config)# isakmp policy priority encryption des | 3des
```

(c) Specify the hash algorithm (the default is **sha**):

```
pixfirewall(config)# isakmp policy priority hash md5 | sha
```

(d) Specify the authentication method:

```
pixfirewall(config)# isakmp policy priority authentication pre-share | rsa-sig
```

**Note** If you specify the authentication method of pre-shared keys, you must manually configure these keys, which is outlined in Step 3.

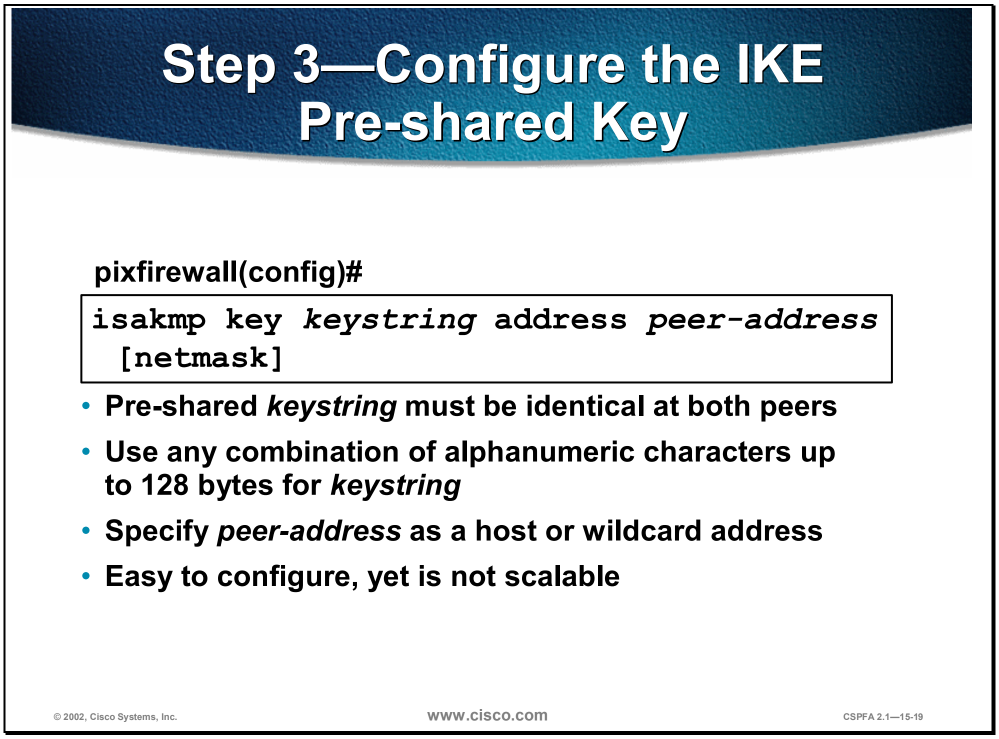
(e) Specify the Diffie-Hellman group identifier (the default is group 1):

```
pixfirewall(config)# isakmp policy priority group 1|2
```

(f) Specify the IKE SA's lifetime (the default is ***86400***).

```
pixfirewall(config)# isakmp policy priority lifetime seconds
```

**Note** PIX Firewall software has preset default values. If you enter a default value for a given policy parameter, it will not be written in the configuration. If you do not specify a value for a given policy parameter, the default value is assigned. You can observe configured and default values with the **show isakmp policy** command.



## Step 3—Configure the IKE Pre-shared Key

**pixfirewall(config)#**

```
isakmp key keystring address peer-address
  [netmask]
```

- **Pre-shared *keystring* must be identical at both peers**
- **Use any combination of alphanumeric characters up to 128 bytes for *keystring***
- **Specify *peer-address* as a host or wildcard address**
- **Easy to configure, yet is not scalable**

www.cisco.com CSPFA 2.1—15-19

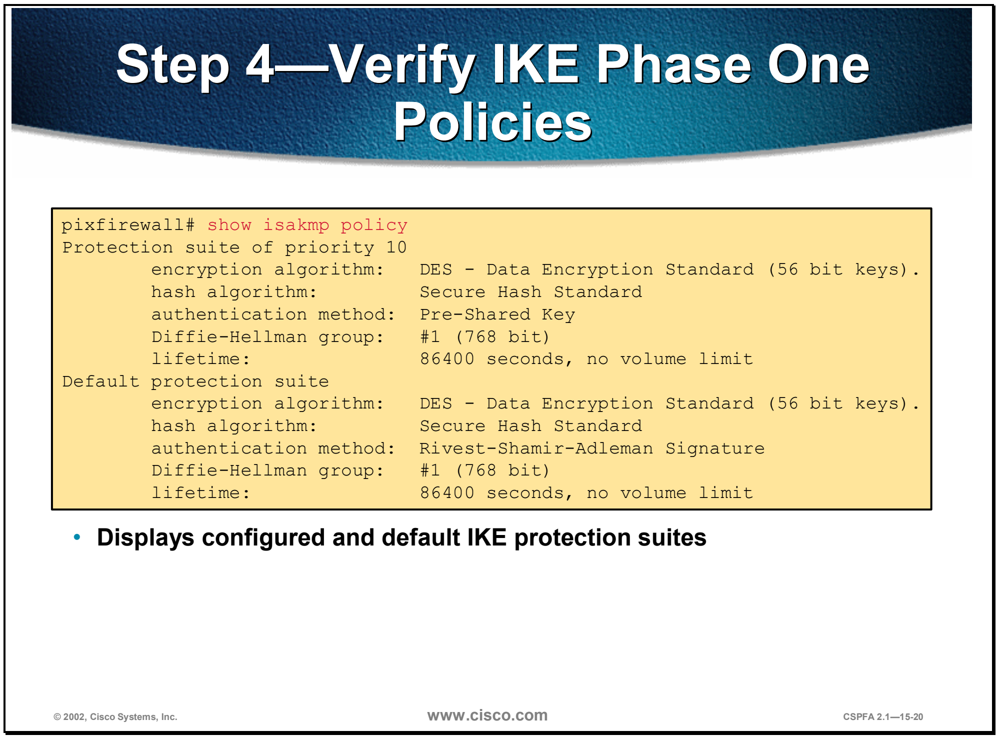
**Step 3** Configure the IKE pre-shared key:

```
pixfirewall(config)# isakmp key keystring address peer-address [netmask]
```

The *keystring* is any combination of alphanumeric characters up to 128 bytes. This pre-shared key must be identical at both peers.

The *peer-address* and *netmask* should point to the IP address of the IPSec peer. A wildcard peer address and netmask of 0.0.0.0 0.0.0.0 may be configured to share the pre-shared key among many peers. However, Cisco strongly recommends using a unique key for each peer.

You can also use the peer's hostname for the pre-shared key.



## Step 4—Verify IKE Phase One Policies

```
pixfirewall# show isakmp policy
Protection suite of priority 10
        encryption algorithm:   DES - Data Encryption Standard (56 bit keys).
        hash algorithm:         Secure Hash Standard
        authentication method:  Pre-Shared Key
        Diffie-Hellman group:   #1 (768 bit)
        lifetime:               86400 seconds, no volume limit
Default protection suite
        encryption algorithm:   DES - Data Encryption Standard (56 bit keys).
        hash algorithm:         Secure Hash Standard
        authentication method:  Rivest-Shamir-Adleman Signature
        Diffie-Hellman group:   #1 (768 bit)
        lifetime:               86400 seconds, no volume limit
```

- **Displays configured and default IKE protection suites**

© 2002, Cisco Systems, Inc. www.cisco.com CSPFA 2.1—15-20

**Step 4** Verify IKE phase one policies.

The **show isakmp policy** command displays configured and default policies, as shown in the figure. The **show isakmp** command displays configured policies much as they would appear with the **write terminal** command, as follows:

```
pix1(config)# show isakmp
isakmp enable outside
isakmp key ******** address 192.168.2.2 netmask 255.255.255.255
isakmp policy 10 authentication pre-share
isakmp policy 10 encryption des
isakmp policy 10 hash sha
isakmp policy 10 group 1
isakmp policy 10 lifetime 86400
```

# Task 3—Configure IPSec Parameters

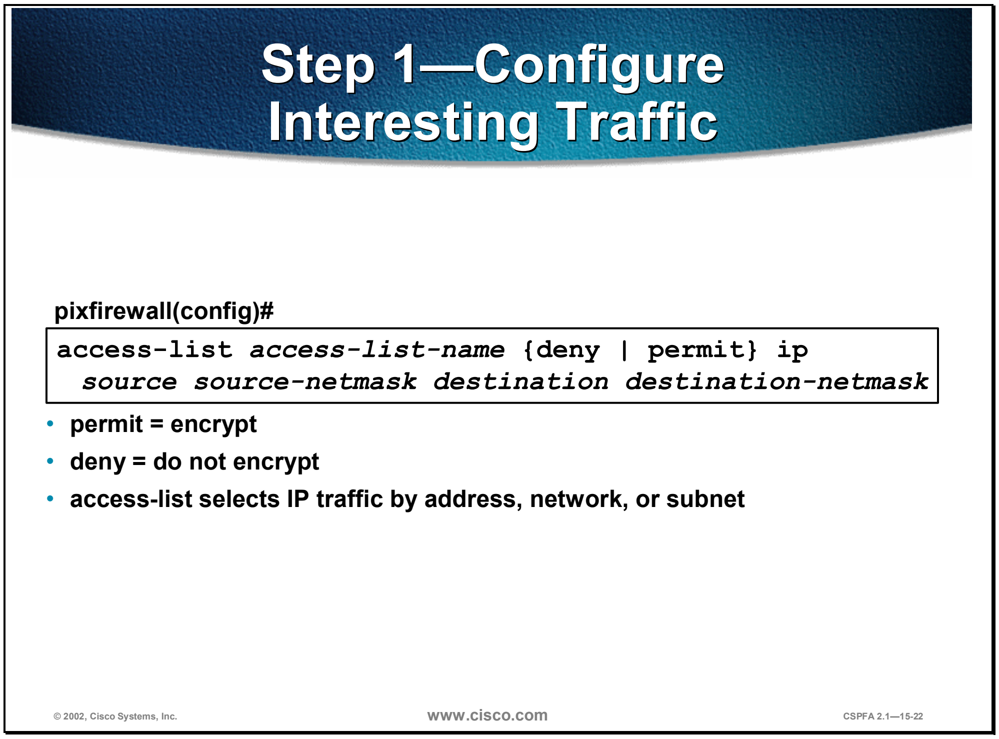
The next major task in configuring PIX Firewall IPSec is to configure the previously gathered IPSec parameters. This section presents the steps used to configure IPSec parameters for IKE pre-shared keys:

**Step 1** Configure interesting traffic.

**Step 2** Configure a transform set.

**Step 3** Configure the crypto map.

**Step 4** Apply the crypto map to the interface.



Step 1—Configure Interesting Traffic

pixfirewall(config)#

```
access-list access-list-name {deny | permit} ip
  source source-netmask destination destination-netmask
```

- permit = encrypt
- deny = do not encrypt
- access-list selects IP traffic by address, network, or subnet

© 2002, Cisco Systems, Inc. www.cisco.com CSPFA 2.1—15-22

**Step 1** Configure interesting traffic with crypto ACLs:

```
pixfirewall(config)# access-list access-list-name {deny | permit} protocol source
source-netmask destination destination-netmask
```

- Permit—Causes all IP traffic that matches the specified conditions to be protected by crypto, using the policy described by the corresponding crypto map entry.
- Deny—Instructs the PIX Firewall to route traffic in the clear.
- Source and destination—Are networks, subnets, or hosts.
- Protocol—Indicates which IP packet types to encrypt.

**Note** PIX Firewall version 5.0 only supports the IP protocol. PIX Firewall versions 5.1 and higher support greater protocol and port granularity.

Crypto ACLs are traffic selection ACLs. They are used to define which IP traffic is interesting and will be protected by IPSec, and which traffic will not be protected by IPSec. Crypto ACLs perform the following functions:

- Indicate the data flow to be protected by IPSec
- Select outbound traffic to be protected by IPSec
- Process inbound traffic in order to filter out and discard traffic that should be protected by IPSec
- Determine whether or not to accept requests for IPSec SAs for the requested data flows when processing IKE negotiations

**Note**Although the ACL syntax is unchanged from ACL applied to PIX Firewall interfaces, the meanings are slightly different for crypto ACLs—**permit** specifies that matching packets must be encrypted while **deny** specifies that matching packets need not be encrypted.

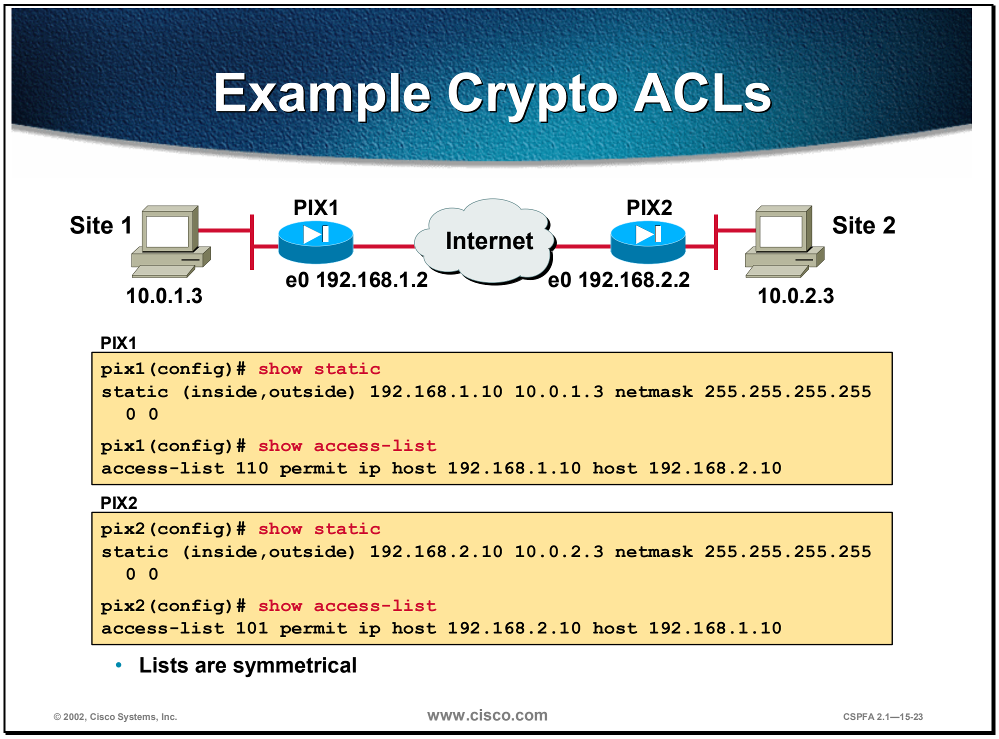
Any unprotected inbound traffic that matches a permit entry in the ACL for a crypto map entry, flagged as IPSec, will be dropped because this traffic was expected to be protected by IPSec.

If you want certain traffic to receive one combination of IPSec protection (for example, authentication only) and other traffic to receive a different combination of IPSec protection (for example, both authentication and encryption), you must create two different crypto ACLs to define the two different types of traffic. These different ACLs are then used in different crypto map entries, which specify different IPSec policies.

In a later configuration step, you will associate the crypto ACLs to particular interfaces when you configure and apply crypto map sets to the interfaces.

***WARNING*** *Cisco recommends that you avoid using the* **any** *keyword to specify source or destination addresses. The* **permit any any** *statement is strongly discouraged, as this will cause all outbound traffic to be protected (and all protected traffic sent to the peer, specified in the corresponding crypto map entry), and will require protection for all inbound traffic. All inbound packets that lack IPSec protection will then be silently dropped, including packets for routing protocols, NTP, echo, echo response, and so on.*

Try to be as restrictive as possible when defining which packets to protect in a crypto ACL. If you must use the **any** keyword in a permit statement, you must preface that statement with a series of deny statements to filter out any traffic (that would otherwise fall within that permit statement) that you do not want protected.



## Example Crypto ACLs

PIX1

```
pix1(config)# show static
static (inside,outside) 192.168.1.10 10.0.1.3 netmask 255.255.255.255
  0 0

pix1(config)# show access-list
access-list 110 permit ip host 192.168.1.10 host 192.168.2.10
```

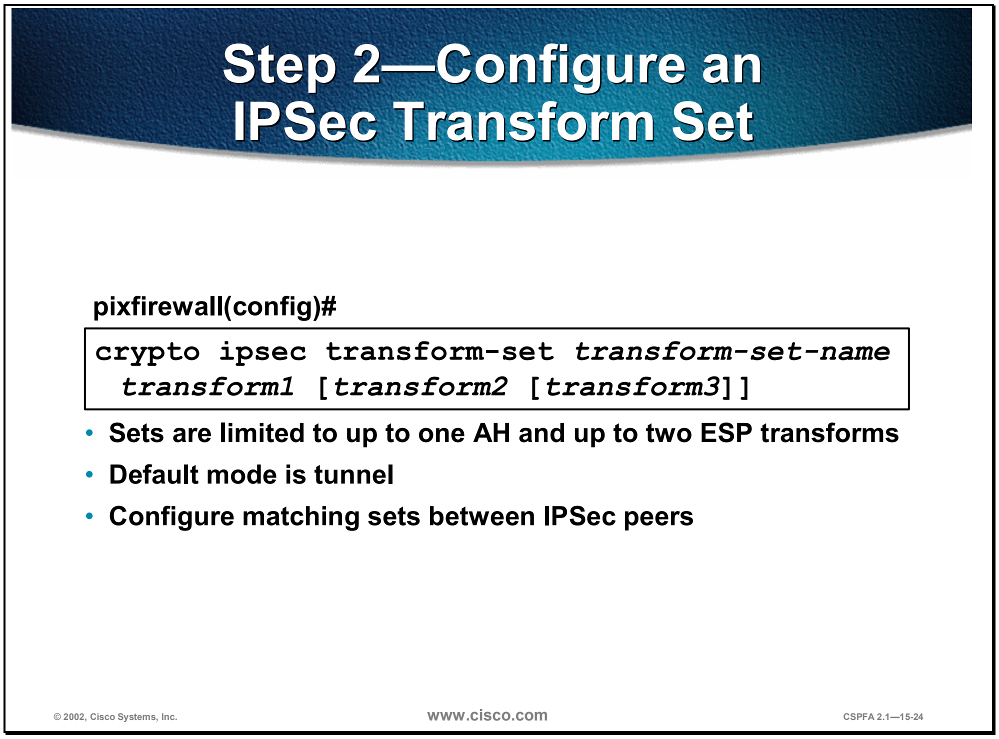
PIX2

```
pix2(config)# show static
static (inside,outside) 192.168.2.10 10.0.2.3 netmask 255.255.255.255
  0 0

pix2(config)# show access-list
access-list 101 permit ip host 192.168.2.10 host 192.168.1.10
```

- Lists are symmetrical

www.cisco.com CSPFA 2.1—15-23

Use the **show access-list** command to display currently configured ACLs. The figure above contains an example ACL for each of the peer PIX Firewalls. Each PIX Firewall in this example has static mapping of a global IP address to an inside host. The ACL *source* field is configured for the global IP address of the local PIX Firewall's static, which is the *destination* field for the peer PIX Firewall's global IP address. The ACLs are symmetrical.



## Step 2—Configure an IPSec Transform Set

pixfirewall(config)#

```
crypto ipsec transform-set transform-set-name
  transform1 [transform2 [transform3]]
```

- Sets are limited to up to one AH and up to two ESP transforms
- Default mode is tunnel
- Configure matching sets between IPSec peers

© 2002, Cisco Systems, Inc. www.cisco.com CSPFA 2.1—15-24

**Step 2** Configure an IPSec transform set:

```
pixfirewall(config)# crypto ipsec transform-set transform-set-name transform1
[transform2 [transform3]]
```

- transform-set-name—The name of the transform set to create or modify
- transform1 [transform2 [transform3]]—You can specify up to three transforms

- Sets are limited to up to one AH and up to two ESP transforms
- The default mode is tunnel.
- Configure matching transform sets between IPSec peers

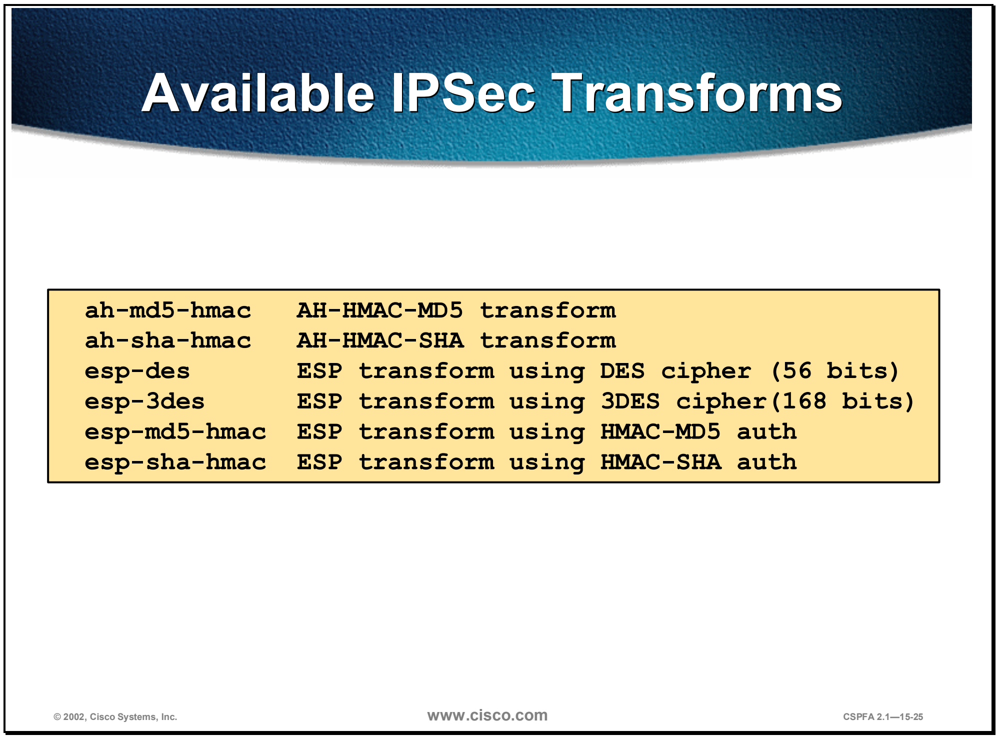
Transforms define the IPSec security protocols and algorithms. Each transform represents an IPSec security protocol (ESP, AH, or both) plus the algorithm you want to use.

You can specify multiple transform sets, and then specify one or more of these transform sets in a crypto map entry. The transform set defined in the crypto map entry will be used in the IPSec SA negotiation to protect the data flows specified by the ACL of that crypto map entry.

During the IPSec SA negotiation, the peers agree to use a particular transform set for protecting a particular data flow.

A transform set equals an AH transform and an ESP transform plus the mode (transport or tunnel.)

**Note** In PIX Firewall versions 6.0 and higher, L2TP is the only protocol that can use the IPSec transport mode. The PIX Firewall discards all other types of packets using IPSec transport mode.



## Available IPSec Transforms

```
ah-md5-hmac   AH-HMAC-MD5 transform
ah-sha-hmac   AH-HMAC-SHA transform
esp-des       ESP transform using DES cipher (56 bits)
esp-3des      ESP transform using 3DES cipher(168 bits)
esp-md5-hmac  ESP transform using HMAC-MD5 auth
esp-sha-hmac  ESP transform using HMAC-SHA auth
```

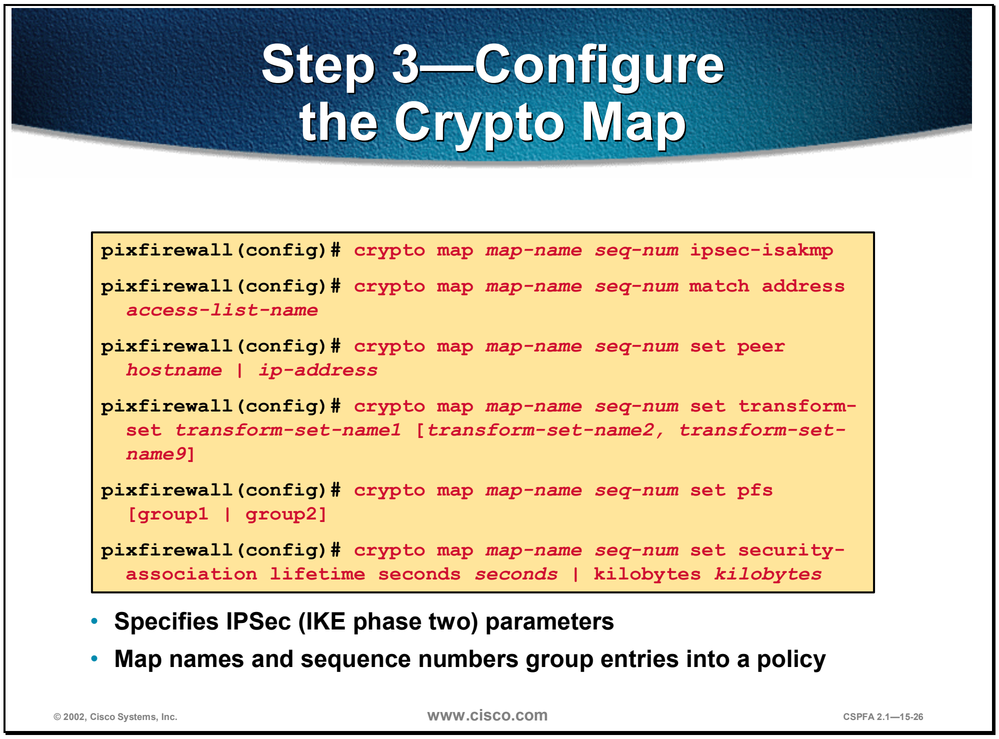
www.cisco.com

CSPFA 2.1—15-25

The PIX Firewall supports the transform sets listed in the figure.

Choosing IPSec transforms combinations can be complex. The following tips may help you select transforms that are appropriate for your situation:

- If you want to provide data confidentiality, include an ESP encryption transform.
- Also consider including an ESP authentication transform or an AH transform to provide authentication services for the transform set:
  - To ensure data authentication for the outer IP header as well as the data, include an AH transform.
  - To ensure data authentication, use either ESP or AH. You can choose from the MD5 or SHA (HMAC keyed hash variants) authentication algorithms.
- The SHA algorithm is generally considered stronger than MD5, but it is slower.
- Examples of acceptable transform combinations are as follows:
  - esp-des for high performance encryption
  - ah-md5-hmac for authenticating packet contents with no encryption
  - esp-3des and esp-md5-hmac for strong encryption and authentication
  - ah-sha-hmac and esp-3des and esp-sha-hmac for strong encryption and authentication



## Step 3—Configure the Crypto Map

```
pixfirewall(config)# crypto map map-name seq-num ipsec-isakmp

pixfirewall(config)# crypto map map-name seq-num match address
  access-list-name

pixfirewall(config)# crypto map map-name seq-num set peer
  hostname | ip-address

pixfirewall(config)# crypto map map-name seq-num set transform-
  set transform-set-name1 [transform-set-name2, transform-set-
  name9]

pixfirewall(config)# crypto map map-name seq-num set pfs
  [group1 | group2]

pixfirewall(config)# crypto map map-name seq-num set security-
  association lifetime seconds seconds | kilobytes kilobytes
```

- Specifies IPSec (IKE phase two) parameters
- Map names and sequence numbers group entries into a policy

© 2002, Cisco Systems, Inc. www.cisco.com CSPFA 2.1—15-26

**Step 3** Configure the crypto map with the **crypto map** command:

(a) Create a crypto map entry in IPSec ISAKMP mode:

```
pixfirewall(config)# crypto map map-name seq-num ipsec-isakmp
```

- This identifies the crypto map with a unique crypto map name and sequence number.

(b) Assign an ACL to the crypto map entry:

```
pixfirewall(config)# crypto map map-name seq-num match address access-list-name
```

(c) Specify the peer to which the IPSec protected traffic can be forwarded:

```
pixfirewall(config)# crypto map map-name seq-num set peer hostname |ip-address
```

- Set the peer hostname or IP address.
- Specify multiple peers by repeating this command.

(d) Specify which transform sets are allowed for this crypto map entry.

```
pixfirewall(config)# crypto map map-name seq-num set transform-set
transform-set-name1 [transform-set-name2, transform-set-name9]
```

- List multiple transform sets in order of priority (highest priority first).
- You can specify up to nine transform sets.

(e) (Optional.) Specify whether IPSec should ask for perfect forward secrecy (PFS) when requesting new SAs for this crypto map entry, or should require PFS in requests received from the peer:

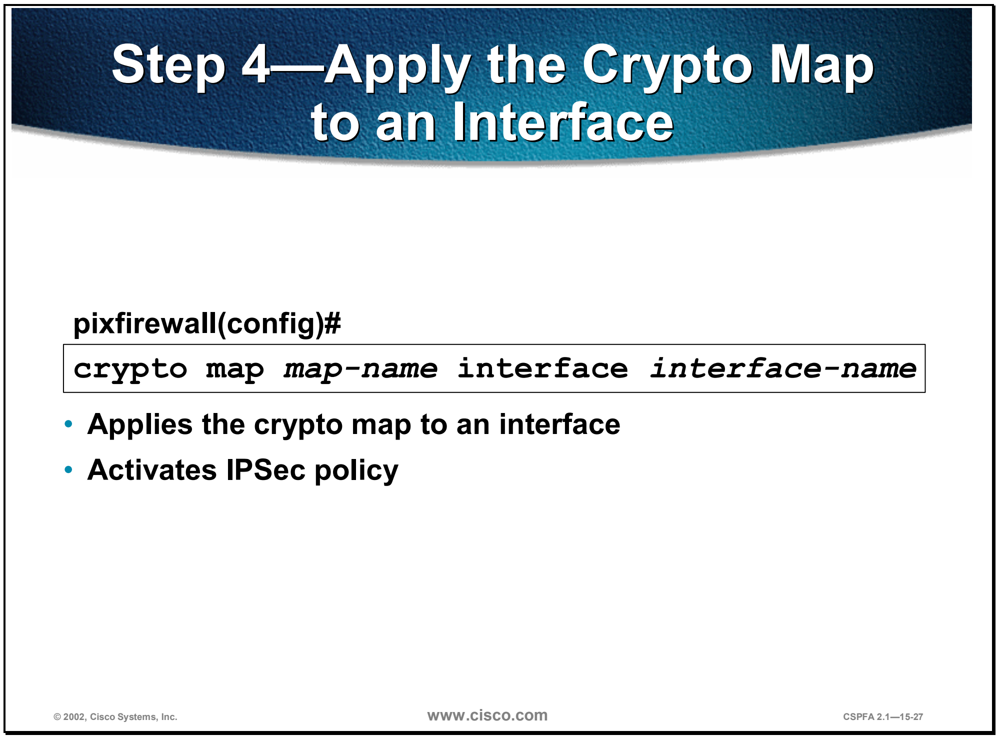
```
pixfirewall(config)# crypto map map-name seq-num set pfs [group1 | group2]
```

**Note** PFS provides additional security for Diffie-Hellman key exchanges at a cost of additional processing.

(f) (Optional.) Specify the SA lifetime for the crypto map entry if you want the SAs for this entry to be negotiated using different IPSec SA lifetimes other than the global lifetimes:

```
pixfirewall(config)# crypto map map-name seq-num set security-association
lifetime seconds seconds | kilobytes kilobytes
```

(g) (Optional.) Specify dynamic crypto maps with the **crypto dynamic-map** ***dynamic-map-name dynamic-seq-num*** command. A dynamic crypto map entry is essentially a crypto map entry without all the parameters configured. It acts as a policy template where the missing parameters are later dynamically configured (as the result of an IPSec negotiation) to match a peer's requirements. This allows peers to exchange IPSec traffic with the PIX Firewall even if the PIX Firewall does not have a crypto map entry specifically configured to meet all the peer's requirements.



## Step 4—Apply the Crypto Map to an Interface

pixfirewall(config)#

```
crypto map map-name interface interface-name
```

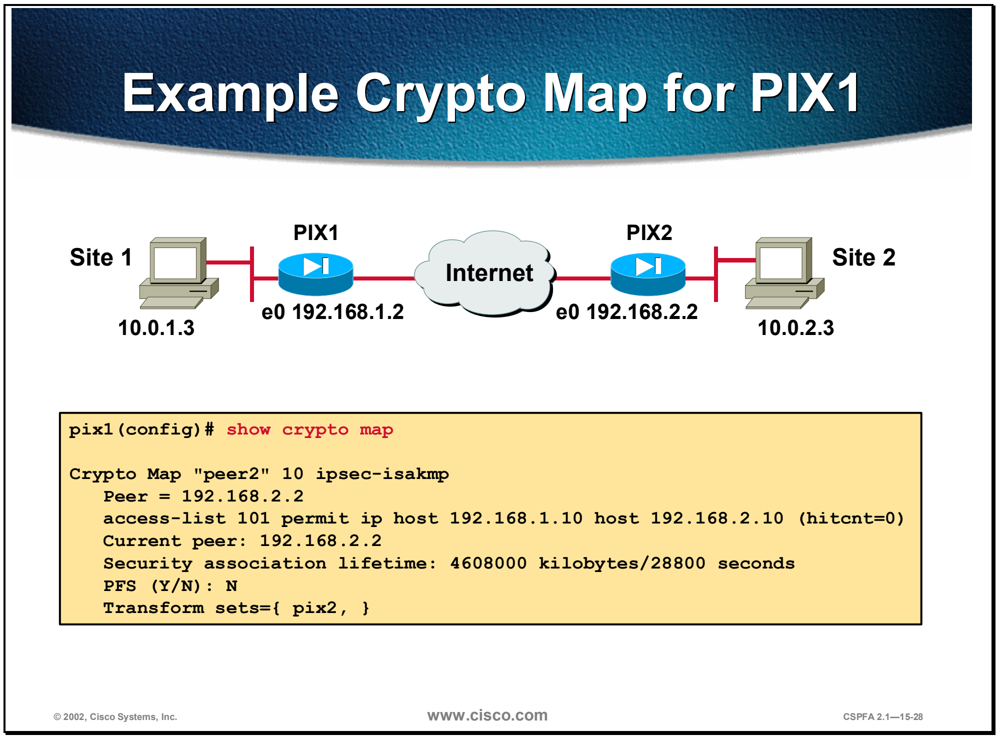
- Applies the crypto map to an interface
- Activates IPSec policy

© 2002, Cisco Systems, Inc. www.cisco.com CSPFA 2.1—15-27

**Step 4** Apply the crypto map to an interface:

```
pixfirewall(config)# crypto map map-name interface interface-name
```

This command applies the crypto map to an interface and the command activates the IPSec policy.

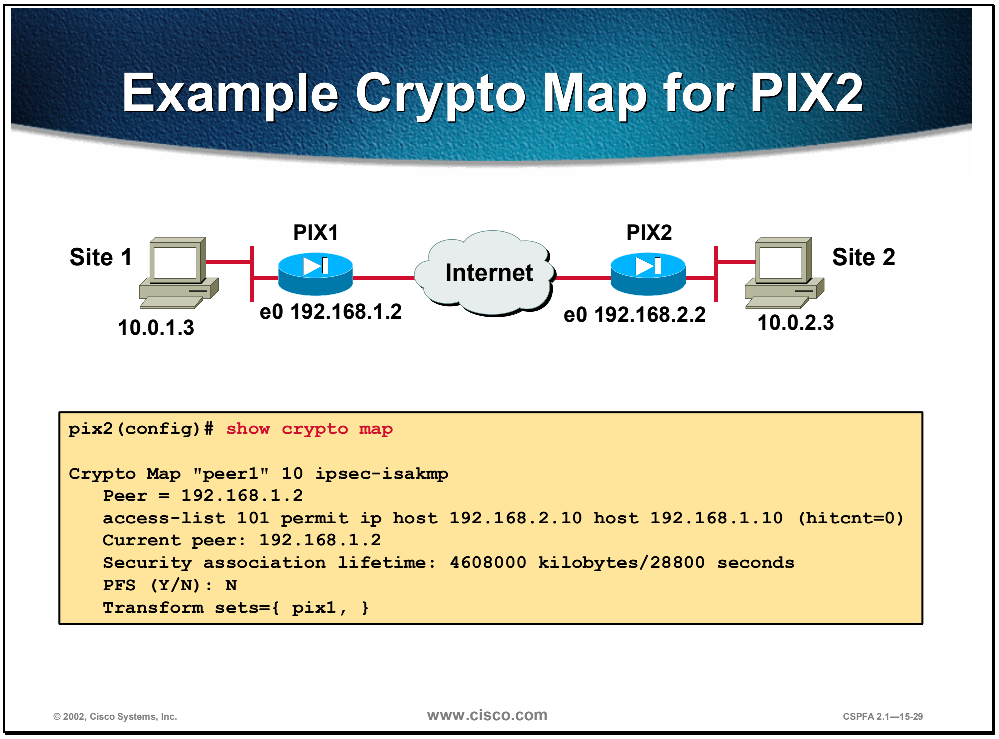


## Example Crypto Map for PIX1

```
pix1(config)# show crypto map

Crypto Map "peer2" 10 ipsec-isakmp
   Peer = 192.168.2.2
   access-list 101 permit ip host 192.168.1.10 host 192.168.2.10 (hitcnt=0)
   Current peer: 192.168.2.2
   Security association lifetime: 4608000 kilobytes/28800 seconds
   PFS (Y/N): N
   Transform sets={ pix2, }
```

www.cisco.com CSPFA 2.1—15-28

Use the **show crypto map** command to verify the crypto map configuration. Consider the example of a crypto map for PIX1 in the figure.



## Example Crypto Map for PIX2

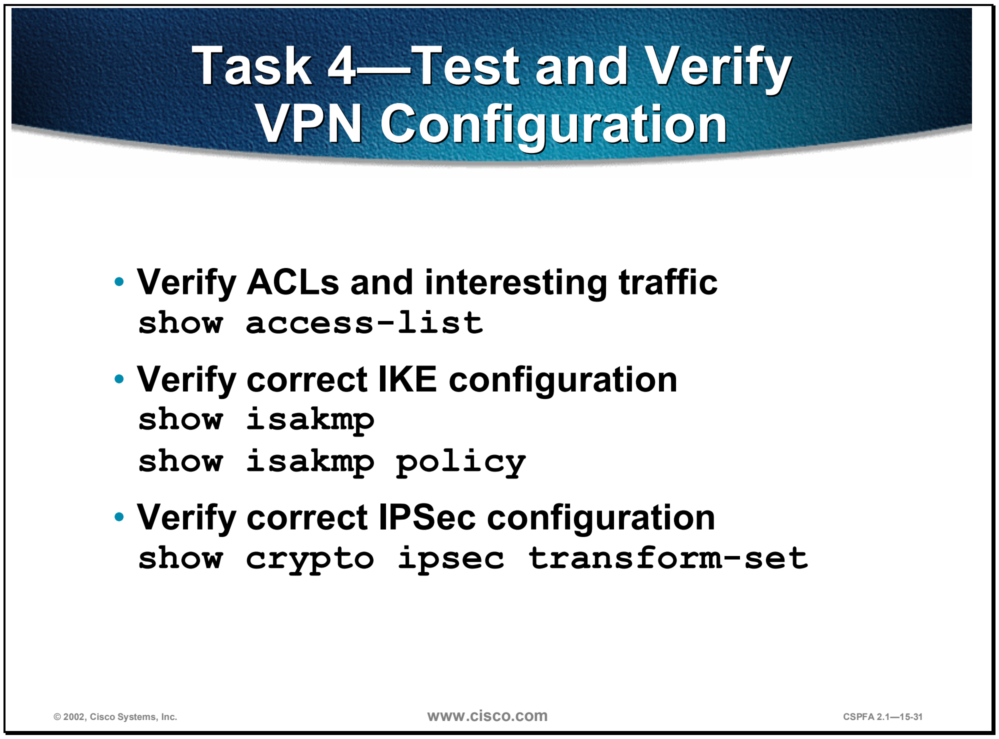
```
pix2(config)# show crypto map

Crypto Map "peer1" 10 ipsec-isakmp
   Peer = 192.168.1.2
   access-list 101 permit ip host 192.168.2.10 host 192.168.1.10 (hitcnt=0)
   Current peer: 192.168.1.2
   Security association lifetime: 4608000 kilobytes/28800 seconds
   PFS (Y/N): N
   Transform sets={ pix1, }
```

www.cisco.com CSPFA 2.1—15-29

Consider the example of a crypto map for PIX2 in the figure.

# Task 4—Test and Verify VPN Configuration

The last major task in configuring PIX Firewall IPSec is to test and verify the IKE and IPSec configurations accomplished in the previous tasks. This section presents the methods and commands used to test and verify VPN configuration.



## Task 4—Test and Verify VPN Configuration

- **Verify ACLs and interesting traffic**

  ```
  show access-list
  ```

- **Verify correct IKE configuration**

  ```
  show isakmp
  show isakmp policy
  ```

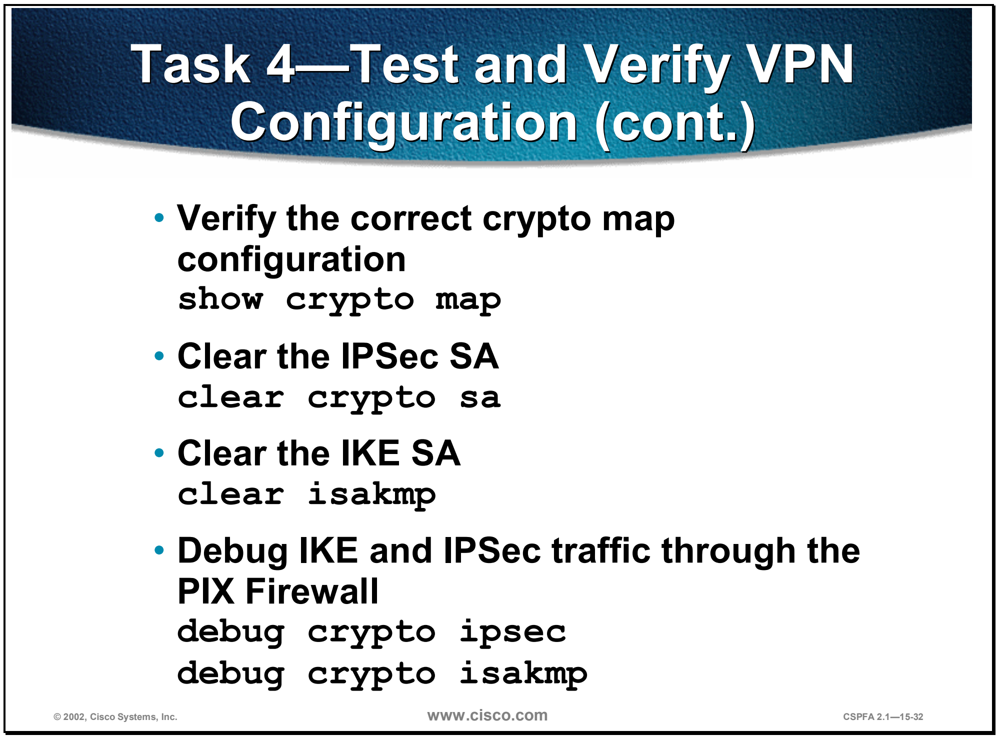
- **Verify correct IPSec configuration**

  ```
  show crypto ipsec transform-set
  ```

© 2002, Cisco Systems, Inc. www.cisco.com CSPFA 2.1—15-31

You can perform the following actions to test and verify that you have correctly configured the VPN on the PIX Firewall:

- Verify ACLs and selects interesting traffic with the **show access-list** command.
- Verify correct IKE configuration with the **show isakmp** and **show isakmp policy** commands.
- Verify correct IPSec configuration of transform sets with the **show crypto ipsec transform-set** command.



## Task 4—Test and Verify VPN Configuration (cont.)

- Verify the correct crypto map configuration
  ```
  show crypto map
  ```
- Clear the IPSec SA
  ```
  clear crypto sa
  ```
- Clear the IKE SA
  ```
  clear isakmp
  ```
- Debug IKE and IPSec traffic through the PIX Firewall
  ```
  debug crypto ipsec
  debug crypto isakmp
  ```

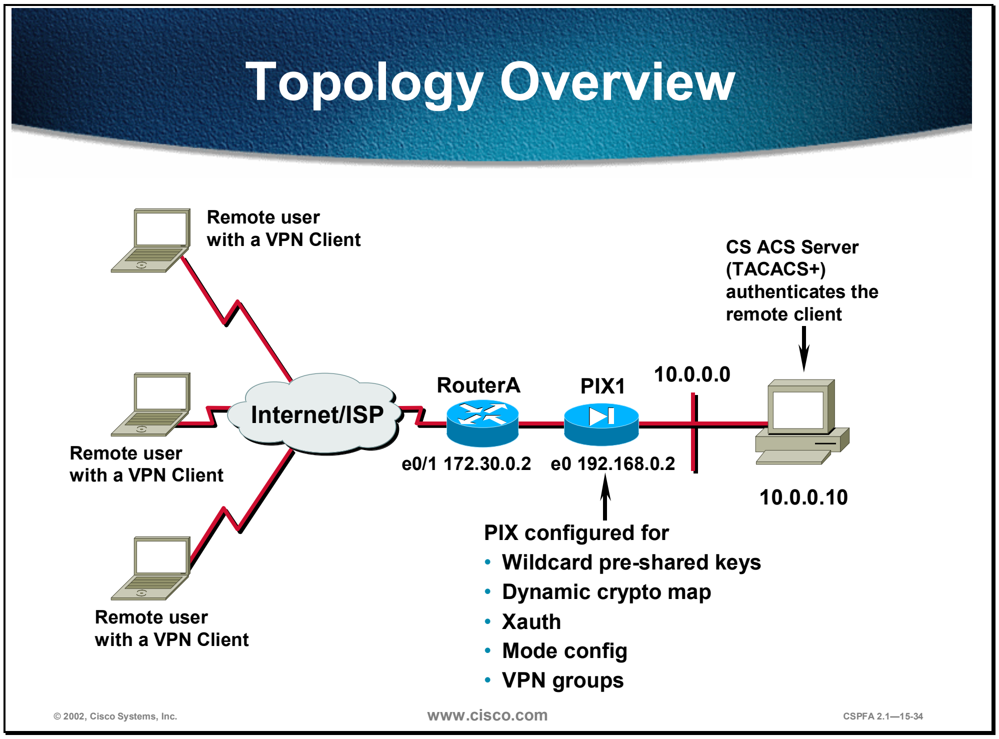
www.cisco.com CSPFA 2.1—15-32

You can perform the following actions to test and verify that you have correctly configured the VPN on the PIX Firewall:

- Verify the correct crypto map configuration with the **show crypto map** command.
- Clear IPSec SAs for testing of SA establishment with the **clear crypto sa** command.
- Clear IKE SAs for testing of IKE SA establishment with the **clear isakmp** command.
- Debug IKE and IPSec traffic through the PIX Firewall with the **debug crypto ipsec** and **debug crypto isakmp** commands.

# The Cisco VPN Client 3.1

This section introduces the Cisco VPN Client and explains how to use it to create a secure VPN with the PIX Firewall.



Topology Overview

PIX configured for

- Wildcard pre-shared keys
- Dynamic crypto map
- Xauth
- Mode config
- VPN groups

© 2002, Cisco Systems, Inc. www.cisco.com CSPFA 2.1—15-34

PIX Firewall software versions 6.0 and higher support the Cisco VPN Client version 3.1, a software program that runs on all current Windows operating systems including Windows 95, 98, Millennium Edition (ME), NT 4.0, 2000 and XP. The Cisco VPN Client enables you to establish secure, end-to-end encrypted tunnels to Cisco remote access VPN devices supporting the Unified Client Framework.

The VPN Client is intended for use by low or high-speed remote users who need to securely access their corporate networks across the Internet. Users can load it on their PC's and launch tunnels as needed to establish connections to a Cisco VPN device that supports the VPN Client. The software VPN Client supports only the device on which it is loaded.

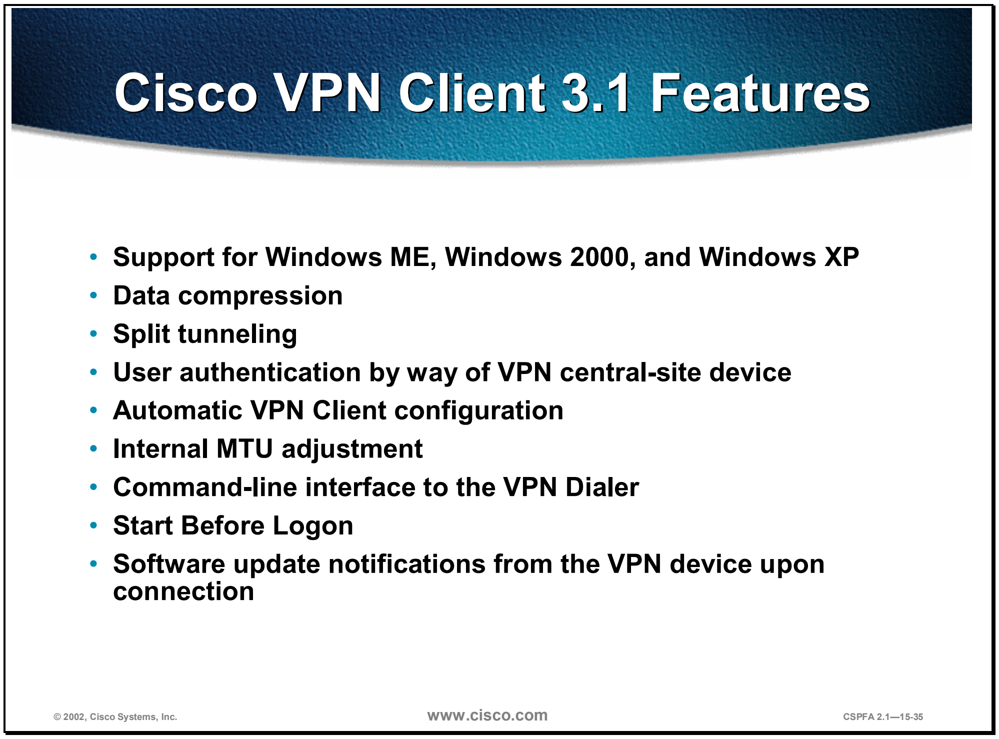
As a remote user, you probably first connect to the Internet before establishing your VPN tunnel. The VPN Client enables you to use Plain Old Telephone Service, ISDN, DSL, or cable modem connection technologies for this connection. It is compatible with Point-to-point Protocol over Ethernet-based DSL and has been tested with Network Telesystems Ethernet, Wind River PoET, and PPOE for Windows 98198SE/ME/2000/XP/2002.

The VPN Client can also be pre-configured for mass deployments, and initial logins require very little user intervention. VPN access policies and configurations are downloaded from the central gateway and pushed to the VPN Client when a connection is established, allowing simple deployment and management as well as high scalability. Items pushed to the VPN Client from the central site concentrator include the following:

- Domain Name System (DNS)
- Windows Internet Naming System (WINS)
- Split tunneling networks
- Default domain name
- IP address
- Ability to save a password for the VPN connection

**Note** The VPN Client 3.1 works with any Cisco VPN-enabled products that support the Unified Client framework. The Cisco Unified Client framework is an initiative that enables consistent VPN Client operation between Service Providers and Enterprises and compatibility across various Cisco VPN platforms. The Unified Client framework currently includes the Cisco VPN 3000 Concentrator Series and the Cisco PIX 500 Firewall series. In the near future, it will also include Cisco VPN 5000 Concentrator Series and Cisco IOS software-based platforms.

The network topology in the figure shows that the remote user with the VPN Client installed will set up an IPSec tunnel to the PIX Firewall via remote access. The PIX Firewall is configured for wildcard pre-shared keys, dynamic crypto maps, Xauth, and IKE mode configuration, and has VPN groups configured with the vpngroup command to enable the PIX Firewall to push IPSec policy to the VPN Client. Pre-shared keys are used for authentication, although the PIX Firewall and VPN Client also support digital certificates. Cisco Secure Access Control Server (CSACS) TACACS+ is used for user authentication via Xauth.



## Cisco VPN Client 3.1 Features

- **Support for Windows ME, Windows 2000, and Windows XP**
- **Data compression**
- **Split tunneling**
- **User authentication by way of VPN central-site device**
- **Automatic VPN Client configuration**
- **Internal MTU adjustment**
- **Command-line interface to the VPN Dialer**
- **Start Before Logon**
- **Software update notifications from the VPN device upon connection**

© 2002, Cisco Systems, Inc. www.cisco.com CSPFA 2.1—15-35

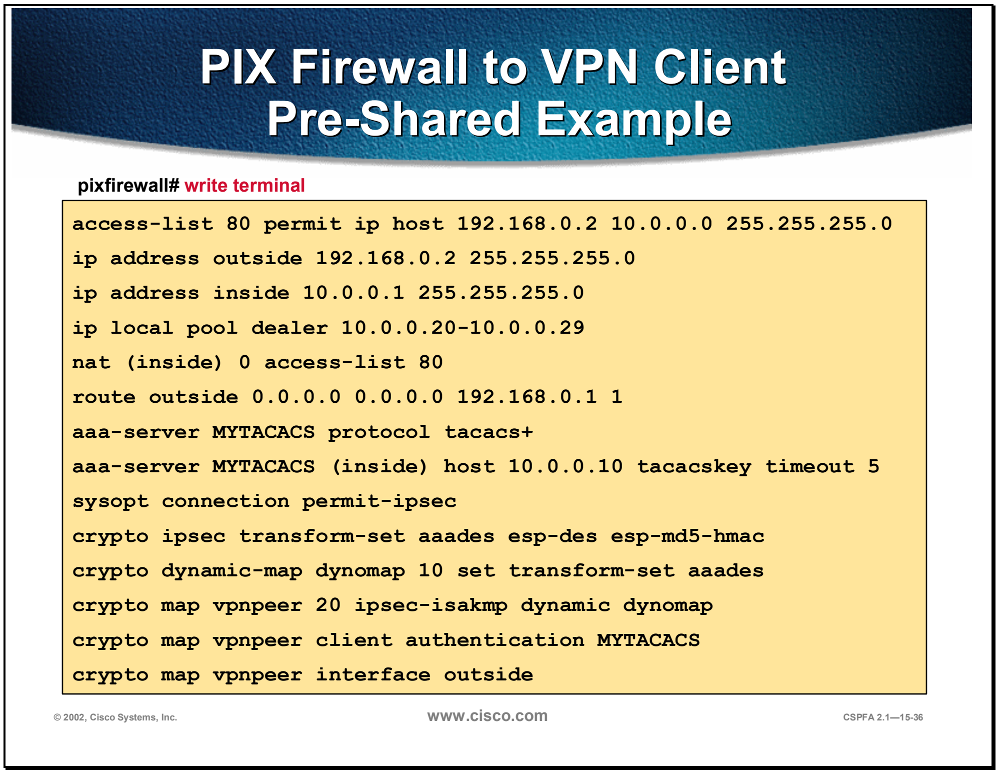
The Cisco VPN Client 3.1 features the following:

- IPSec tunneling protocol.
- IKE key management protocol.
- IKE Keepalives—Monitoring the continued presence of a peer and reporting the VPN Client's continued presence to the peer. This lets the VPN Client notify you when the peer is no longer present. Another type of keepalive keeps NAT ports alive.
- Data compression for modem users, which speeds transmission.
- LZS data compression, which also benefits modem users.
- Split tunneling—The ability to simultaneously direct packets over the Internet in clear text and encrypted through an IPSec tunnel.
- Local LAN access—The ability to access resources on a local LAN while connected through a secure gateway to a central-site VPN device (if the central site grants permission).
- User authentication by way of VPN central-site device:
  - Internal through the VPN device's database.
  - RADIUS (Remote Authentication Dial-In User Service).
  - NT Domain (Windows NT).
  - RSA (formerly SDI) SecurID or SoftID.
- Automatic connection by way of Microsoft Dial-Up Networking or any other third-party remote access dialer.
- Automatic VPN Client configuration option—the ability to import a configuration file.
- Log Viewer—An application that collects events for viewing and analysis.

- Set MTU size—The VPN Client automatically sets a size that is optimal for your environment. However, you can set the MTU size manually as well.
- Certificate Manager—An application that enables you to manage your identity certificates.
- Complete browser-based context-sensitive HTML help.
- Support for PIX Firewall platforms that run Release 6.0 and above.
- CLI to the VPN Dialer.
- Start Before Logon feature—The ability to establish a VPN connection before logging on to a Windows NT platform, which includes Windows NT 4.0, Windows 2000, and Windows XP systems.
- Ability to disable automatic disconnect when logging off of a Windows NT platform. This allows for roaming profile synchronization.
- Application Launcher—The ability to launch an application or a third-party dialer from the VPN Client.
- Software update notifications from the VPN device upon connection.
- Ability to use Entrust Entelligence certificates.

The VPN Client also supports the following IPSec attributes:

- Main mode for negotiating phase one of establishing ISAKMP Security Associations (SAs).
- Aggressive mode for negotiating phase one of establishing ISAKMP SAs.
- Authentication algorithm—HMAC with MD5 hash function.
- HMAC with SHA-1 (Secure Hash Algorithm) hash function.
- Authentication Mode—Pre-shared Keys.
- X.509 digital certificates.
- Diffie-Hellman (DH) Groups 1, 2, and 5.
- Encryption algorithms.
- 56-bit Data Encryption Standard (DES).
- 168-bit Triple-DES (3DES).
- Extended Authentication (XAUTH).
- Mode Configuration (also known as ISAKMP Configuration Method).
- Tunnel Encapsulation Mode.
- IP compression (IPCOMP) using LZS.



## PIX Firewall to VPN Client Pre-Shared Example

pixfirewall# write terminal

```
access-list 80 permit ip host 192.168.0.2 10.0.0.0 255.255.255.0
ip address outside 192.168.0.2 255.255.255.0
ip address inside 10.0.0.1 255.255.255.0
ip local pool dealer 10.0.0.20-10.0.0.29
nat (inside) 0 access-list 80
route outside 0.0.0.0 0.0.0.0 192.168.0.1 1
aaa-server MYTACACS protocol tacacs+
aaa-server MYTACACS (inside) host 10.0.0.10 tacacskey timeout 5
sysopt connection permit-ipsec
crypto ipsec transform-set aaades esp-des esp-md5-hmac
crypto dynamic-map dynomap 10 set transform-set aaades
crypto map vpnpeer 20 ipsec-isakmp dynamic dynomap
crypto map vpnpeer client authentication MYTACACS
crypto map vpnpeer interface outside
```

www.cisco.com

CSPFA 2.1—15-36

The figure shows an example configuration for the PIX Firewall to support the VPN Client, which is explained as follows. An ACL for the PIX Firewall and VPN Clients is configured. The source is the PIX Firewall's outside interface IP address, and the destination is a subnet pointing to the inside address assigned to the VPN Client as specified in the **ip local pool** command:

```
access-list 80 permit ip host 192.168.0.2 10.0.0.0 255.255.255.0
ip address outside 192.168.0.2 255.255.255.0
ip address inside 10.0.0.1 255.255.255.0
```

A pool of IP addresses that will dynamically be assigned to the VPN clients via IKE mode configuration is set up**: ip local pool dealer 10.0.0.20-10.0.0.29**.

Nat 0 is used to point to the ACL so that no NAT is taking place inside the tunnel:

```
nat (inside) 0 access-list 80
route outside 0.0.0.0 0.0.0.0 192.168.0.1 1
```

The CSACS server is set up for Xauth user authentication. CSACS (or a remote database it uses such as Microsoft NT) must be configured with usernames and passwords, and must point to the PIX Firewall as a Network Access Server (NAS). The TACACS+ key (tacacskey in this example) must match in CSACS and in the PIX Firewall:

```
aaa-server MYTACACS protocol tacacs+
aaa-server MYTACACS (inside) host 10.0.0.10 tacacskey timeout 5
sysopt connection permit-ipsec
```

A transform is set up, which will be used for the VPN clients: **crypto ipsec transform-set aaades esp-des esp-md5-hmac**.

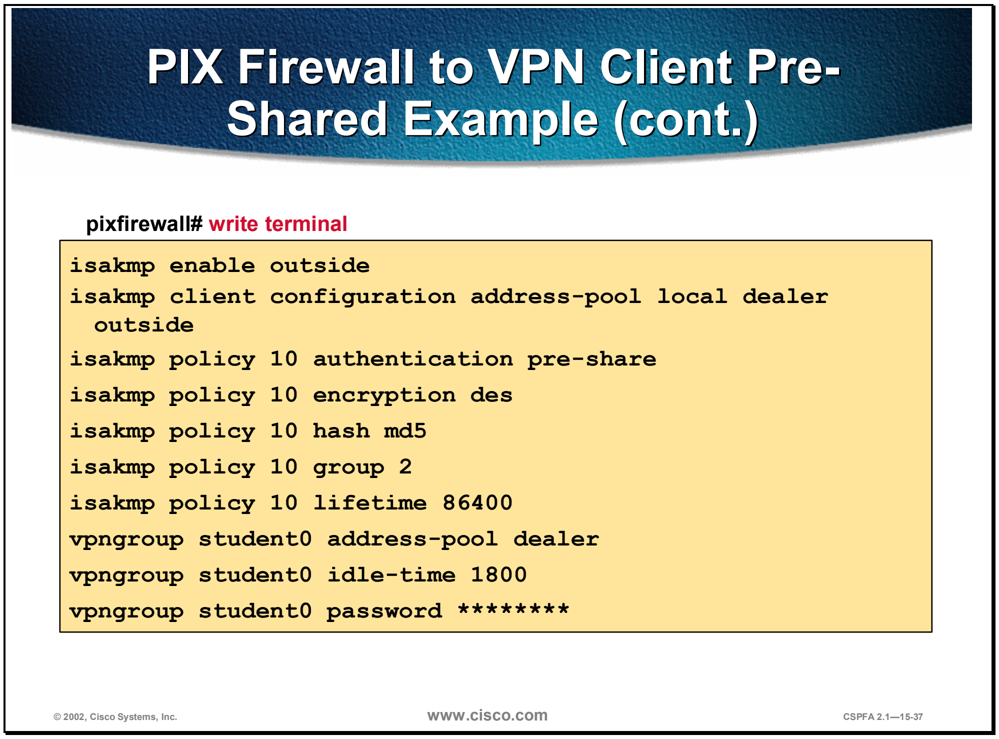
A dynamic crypto map is set up, which enables the VPN Clients to connect to the PIX Firewall. You must set a transform set in the dynamic map as a minimum.

You may also want to set up an ACL for additional security: **crypto dynamic-map dynomap 10 set transform-set aaades**.

A crypto map is created, and the dynamic crypto map is assigned to it: **crypto map vpnpeer 20 ipsec-isakmp dynamic dynomap**.

The Xauth is configured to point to the TACACS+ server: **crypto map vpnpeer client authentication MYTACACS**.

The crypto map is applied to the PIX Firewall interface: **crypto map vpnpeer interface outside**.



## PIX Firewall to VPN Client Pre-Shared Example (cont.)

pixfirewall# write terminal

```
isakmp enable outside
isakmp client configuration address-pool local dealer
  outside
isakmp policy 10 authentication pre-share
isakmp policy 10 encryption des
isakmp policy 10 hash md5
isakmp policy 10 group 2
isakmp policy 10 lifetime 86400
vpngroup student0 address-pool dealer
vpngroup student0 idle-time 1800
vpngroup student0 password ********
```

© 2002, Cisco Systems, Inc. www.cisco.com CSPFA 2.1—15-37

The figure continues to show the example configuration for the PIX Firewall to support the VPN Client, which is explained as follows. The ISAKMP policy is configured  just as you would with any IPSec peer: **isakmp enable outside**.

The IKE mode configuration related parameters are configured: **isakmp client configuration address-pool local dealer outside**.

The ISAKMP policy is configured:

```
isakmp policy 10 authentication pre-share
isakmp policy 10 encryption des
isakmp policy 10 hash md5
isakmp policy 10 group 2
isakmp policy 10 lifetime 86400
```

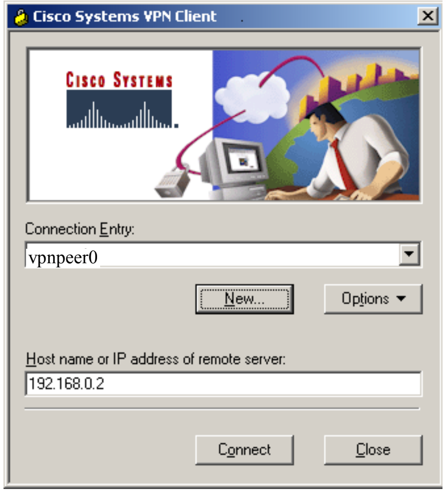
The VPN group is configured to support pushing mode configuration parameters to the VPN Client. The VPN group name of *student0* must match the group name in the VPN Client. The VPN group password must match the password in the VPN Client. You can also configure the VPN group to push DNS, WINS, domain name, and split tunneling information to the VPN Client:

```
vpngroup student0 address-pool dealer
vpngroup student0 idle-time 1800
vpngroup student0 password ********
```

**Note** Cisco VPN Client versions 3.0 and higher use DH group 2, and Cisco VPN Client 3000 version 2.5 uses DH group 1. If you are using Cisco VPN Client version 3.0 or higher, configure DH group 2 by using the **isakmp policy** command.

## VPN Client to PIX Firewall Example

- A new connection entry named vpnpeer0 is created.
- The remote server IP is the PIX Firewall outside interface.

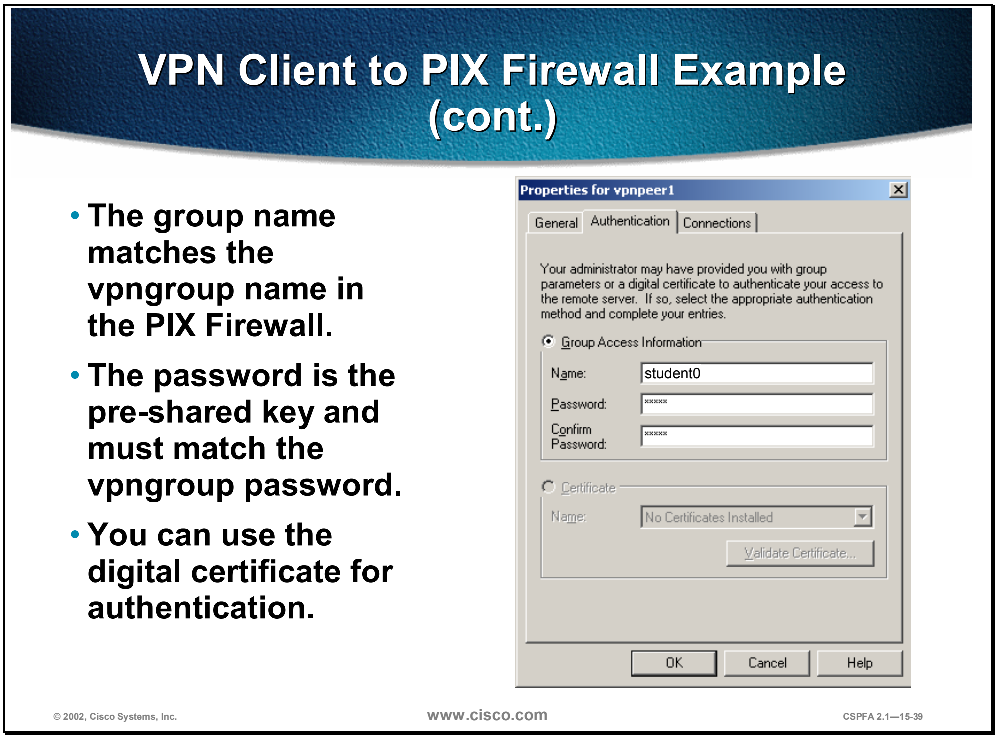


Connection Entry: vpnpeer0

Host name or IP address of remote server: 192.168.0.2

www.cisco.com CSPFA 2.1—15-38

You must configure the VPN Client to interoperate with the PIX Firewall. You need to create a new connection entry (vpnpeer0, as shown in the figure). The host name or IP address of the peer should be the PIX Firewall outside interface.



## VPN Client to PIX Firewall Example (cont.)

- The group name matches the vpngroup name in the PIX Firewall.
- The password is the pre-shared key and must match the vpngroup password.
- You can use the digital certificate for authentication.

**Properties for vpnpeer1**

Your administrator may have provided you with group parameters or a digital certificate to authenticate your access to the remote server. If so, select the appropriate authentication method and complete your entries.

● Group Access Information

Name: student0

Password: *****

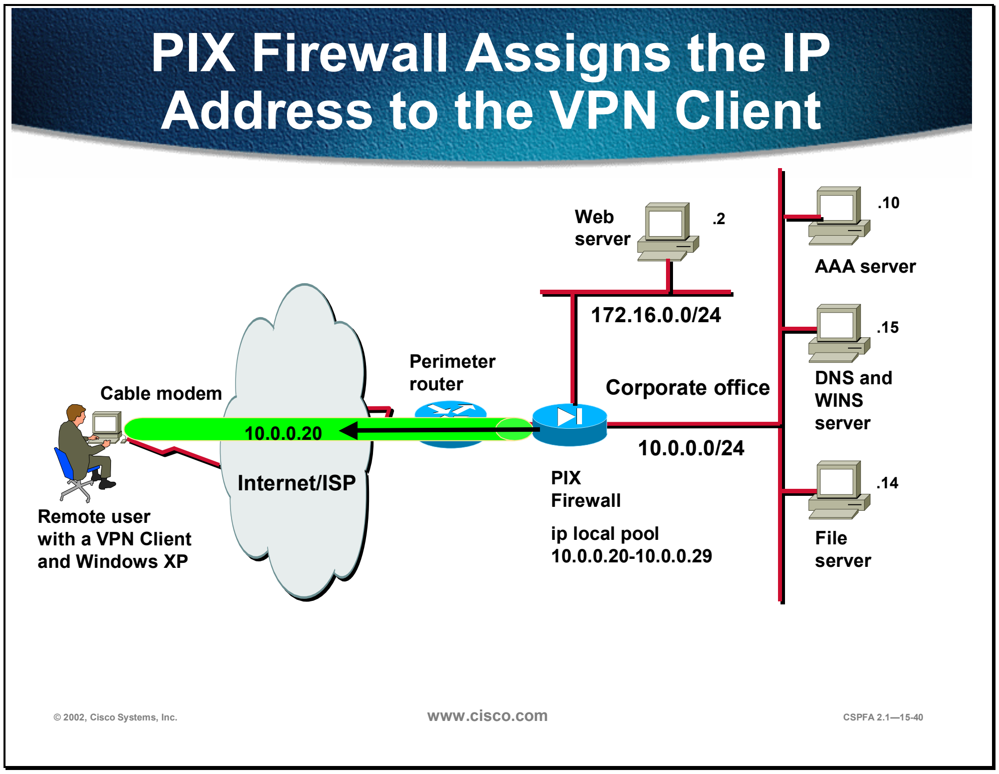
Confirm Password: *****

○ Certificate

Name: No Certificates Installed

www.cisco.com CSPFA 2.1—15-39

If you are using pre-shared keys for authentication, you must make sure that the group name (student0, in this case) matches the VPN group name on the PIX Firewall, and that the password (the pre-shared key) matches the VPN group password. You can use digital certificates for authentication instead of pre-shared keys.



## PIX Firewall Assigns the IP Address to the VPN Client

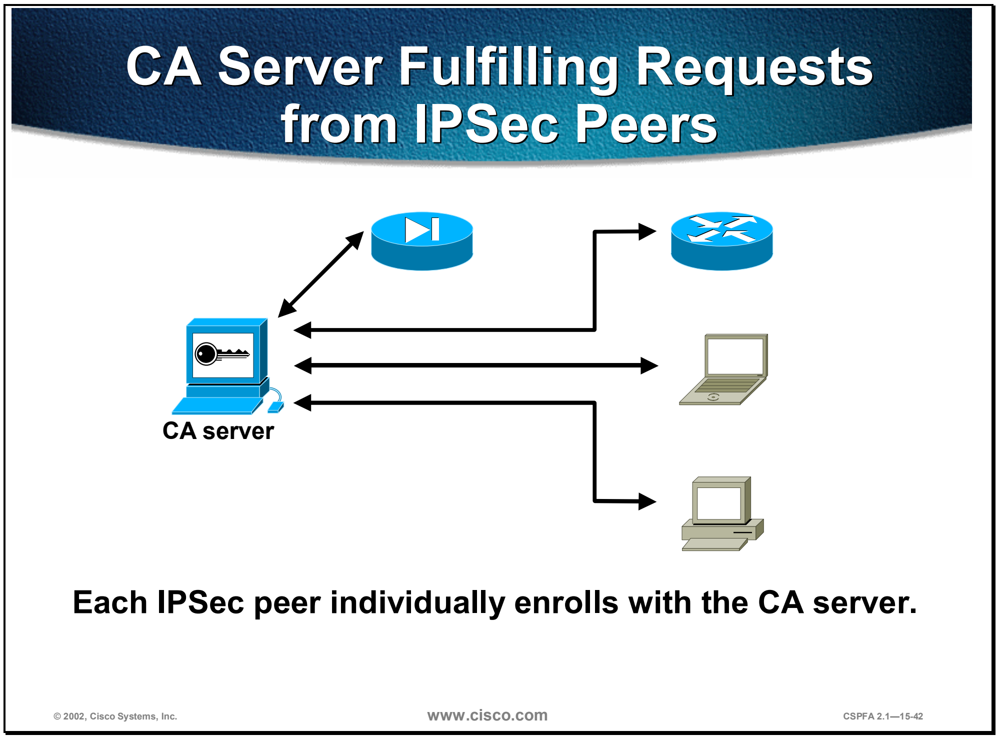
© 2002, Cisco Systems, Inc. www.cisco.com CSPFA 2.1—15-40

When the VPN Client initiates ISAKMP with the PIX Firewall, the VPN group name and pre-shared key are sent to the PIX Firewall. The PIX Firewall then uses the group name to look up the configured VPN Client policy attributes for the given VPN Client and downloads the matching policy attributes to the VPN Client during the IKE negotiation.

In the figure, the PIX Firewall uses IKE mode configuration to push IPSec policy defined with the **vpngroup** command to the VPN Client. The VPN Client IP address (10.0.0.20, in this example) is set with the PIX Firewall **ip local pool** and **vpngroup** commands.

# Scale PIX Firewall VPNs

This section explains how to scale PIX Firewall VPNs using Certificate Authorities (CAs).

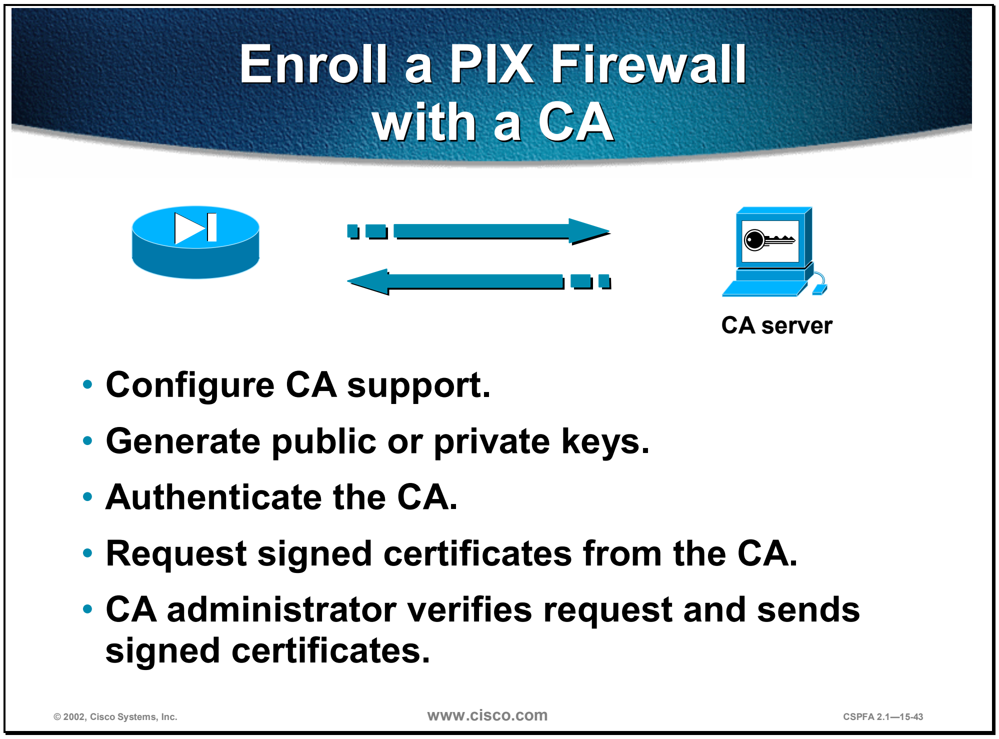


CA Server Fulfilling Requests from IPSec Peers

Each IPSec peer individually enrolls with the CA server.

© 2002, Cisco Systems, Inc. www.cisco.com CSPFA 2.1—15-42

The use of pre-shared keys for IKE authentication works only when you have a few IPSec peers. CAs enable scaling to a large number of IPSec peers. Although there are a number of methods, using a CA server is the most scalable solution. Other IKE authentication methods require manual intervention to generate and distribute the keys on a per-peer basis. The CA server enrollment process can be largely automated so that it scales well to large deployments. Each IPSec peer individually enrolls with the CA server and obtains public and private encryption keys compatible with other peers enrolled with the server.



## Enroll a PIX Firewall with a CA

- Configure CA support.
- Generate public or private keys.
- Authenticate the CA.
- Request signed certificates from the CA.
- CA administrator verifies request and sends signed certificates.

© 2002, Cisco Systems, Inc. www.cisco.com CSPFA 2.1—15-43

Peers enroll with a CA server in a series of steps in which specific keys are generated and then exchanged by the PIX Firewall and the CA server to ultimately form a signed certificate. The enrollment steps can be summarized as follows:

**Step 1** The PIX Firewall generates an RSA key pair.

**Step 2** The PIX Firewall obtains a public key and its certificate from the CA server.

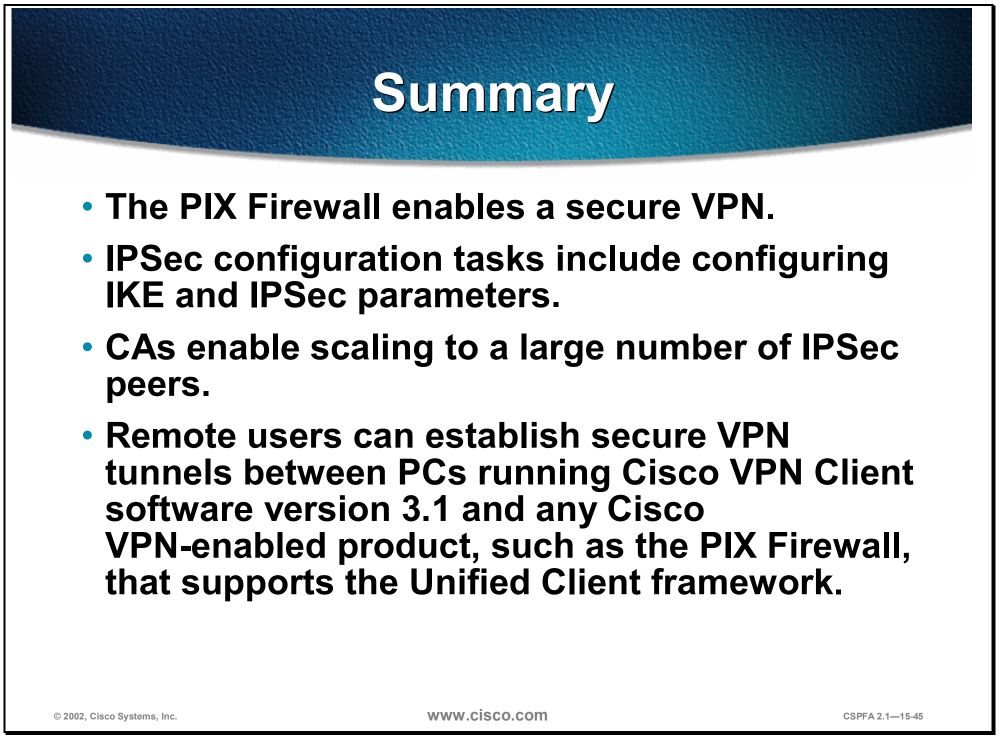
**Step 3** The PIX Firewall requests a signed certificate from the CA using the generated RSA keys and the public key certificate from the CA server.

**Step 4** The CA administrator verifies the request and sends a signed certificate.

**Note** See the “About CA” and “Configuring CA” sections in the “Configuring IPSec” chapter of the *Configuration Guide for the Cisco PIX Firewall* for more details on how CA servers work and how to configure the PIX Firewall for CA support.

# Summary

This section summarizes the tasks you learned to complete in this chapter.



## Summary

- The PIX Firewall enables a secure VPN.
- IPSec configuration tasks include configuring IKE and IPSec parameters.
- CAs enable scaling to a large number of IPSec peers.
- Remote users can establish secure VPN tunnels between PCs running Cisco VPN Client software version 3.1 and any Cisco VPN-enabled product, such as the PIX Firewall, that supports the Unified Client framework.

© 2002, Cisco Systems, Inc. www.cisco.com CSPFA 2.1—15-45

# Lab Exercise—Configure PIX Firewall VPNs

Complete the following lab exercise to practice what you learned in this chapter.

There are two lab exercise sub-sections:

- Configure a secure VPN gateway using IPSec between two PIX Firewalls
- Configure a secure VPN using IPSec between a PIX Firewall and a VPN Client

## Configure a Secure VPN Gateway Using IPSec Between Two PIX Firewalls

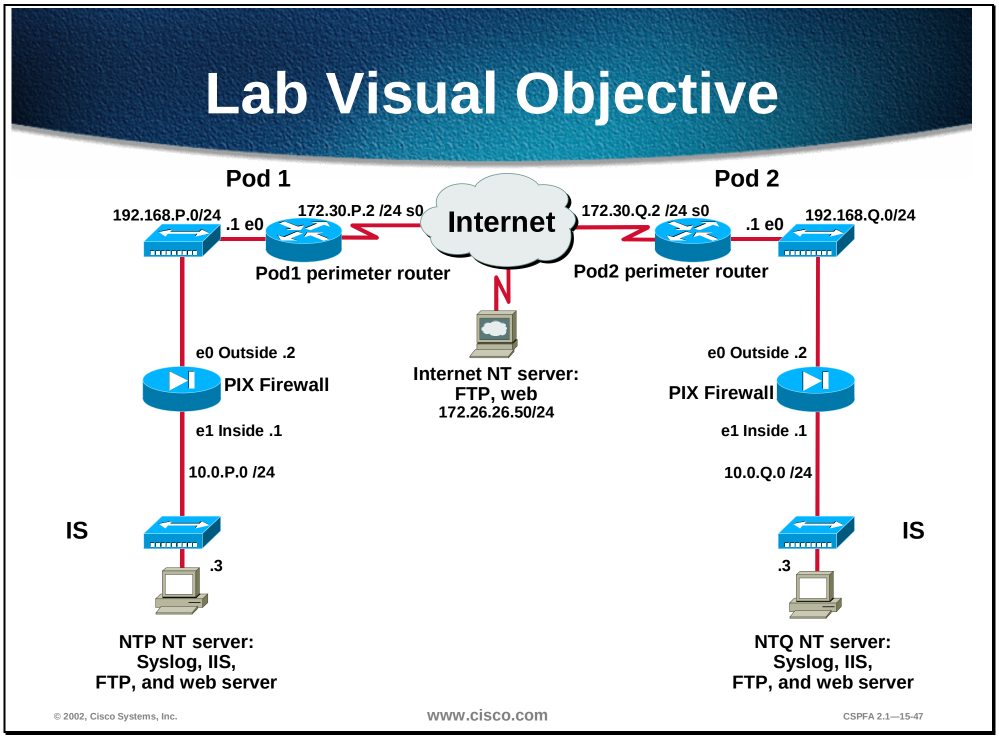
## Objectives

In this lab exercise you will complete the following tasks:

- Prepare to configure VPN support.
- Configure IKE parameters.
- Configure IPSec parameters.
- Test and verify IPSec configuration.

## Visual Objective

The following figure displays the configuration you will complete in this lab exercise.



Lab Visual Objective

© 2002, Cisco Systems, Inc. www.cisco.com CSPFA 2.1—15-47

## Scenario

The XYZ Company has purchased PIX Firewalls to create a secure VPN over the Internet between sites. The company wants you to configure a secure VPN gateway using IPSec between two PIX Firewalls.

## Setup

Before starting this lab exercise, set up your equipment as follows:

- Ensure your Windows NT server is turned on.
- Access the PIX Firewall console port. You may wish to save the PIX Firewall configuration to a text file for later analysis.
- Make sure the PIX Firewall is turned on.
- Ensure you can ping from your internal Windows NT server to the opposite pod group’s Windows NT server.
- Ensure the web server is running on your own internal Windows NT server.
- Ensure you can establish a Web connection from a web browser on your internal Windows NT server to the opposite pod group’s Windows NT server.

## Task 1—Prepare to Configure VPN Support

Perform the following lab steps to prepare for the IKE and IPSec configuration. In this exercise, you will use default values except when you are directed to enter a specific value. Your IKE policy will use pre-shared keys. Your IPSec policy will use ESP mode with DES encryption.

**Step 1** Verify that a static translation is configured from a global IP address on the outside interface to the internal Windows NT server:

```
pixP(config)# show static
static (inside,outside) 192.168.P.10 10.0.P.3 netmask 255.255.255.255 0 0
```

(where P = pod number)

**Step 2** Verify a conduit permitting Web access to your internal Windows NT server has been configured:

```
pixP(config)# show conduit
conduit permit tcp host 192.168.P.10 eq www any
```

(where P = pod number)

**Step 3** Ensure you can establish a Web connection between pods from the Windows NT server using the static and conduit.

**Step 4** Enable the PIX Firewall to implicitly permit any packet from an IPSec tunnel and bypass checking with an associated conduit or **access-group** command for IPSec connections:

```
pixP(config)# sysopt connection permit-ipsec
```

## Task 2—Configure IKE Parameters

Perform the following steps to configure IKE on your PIX Firewall:

**Step 1** Ensure IKE is enabled on the outside interface:

```
pixP(config)# isakmp enable outside
```

**Step 2** Configure a basic IKE policy using pre-shared keys for authentication:

```
pixP(config)# isakmp policy 10 authentication pre-share
```

**Step 3** Set the IKE identity:

```
pixP(config)# isakmp identity address
```

**Step 4** Configure the ISAKMP pre-shared key to point to the outside IP address of the peer PIX Firewall:

```
pixP(config)# isakmp key cisco123 address 192.168.Q.2 netmask 255.255.255.255
```

(where P = pod number, and Q = peer pod number)

## Task 3—Configure IPSec Parameters

Perform the following steps to configure IPSec (IKE phase two) parameters:

**Step 1** Create an ACL to select traffic to protect. The ACL should protect IP traffic between Windows NT servers on the peer PIX Firewalls:

```
pixP(config)# access-list 101 permit ip host 192.168.P.10 host 192.168.Q.10
```

(where P = pod number, and Q = peer pod number)

Consider the example ACL for PIX1 peering to PIX2:

```
pix1(config)# show access-list
access-list 101 permit ip host 192.168.P.10 host 192.168.Q.10
```

(where P = pod number, and Q = peer pod number)

**Step 2** Configure an IPSec transform set (IKE phase two parameters) to use ESP and DES. Use a *transform-set-name* of pixQ:

```
pixP(config)# crypto ipsec transform-set pixQ esp-des
```

(where Q = peer pod number)

**Step 3** Create a crypto map by performing the following sub-steps:

1. Create a crypto map entry. Use a *map-name* of peer Q:

```
pixP(config)# crypto map peerQ 10 ipsec-isakmp
```

(where Q = peer pod number)

2. Look at the crypto map and observe the defaults:

```
pixP(config)# show crypto map
Crypto Map “peerQ” 10 ipsec-isakmp
        No matching address list set.
        Current peer: 0.0.0.0
        Security association lifetime: 4608000 kilobytes/28800 seconds
        PFS (Y/N): N
        Transform sets={ }
```

Q 1) What is the default SA lifetime?

A) 4608000 kilobytes/28800 seconds

3. Assign the ACL to the crypto map:

```
pixP(config)# crypto map peerQ 10 match address 101
```

(where Q = peer pod number)

4. Define the peer. The peer IP address should be set to the peer’s outside interface IP address:

```
pixP(config)# crypto map peerQ 10 set peer 192.168.Q.2
```

(where Q = peer pod number)

5. Specify the transform set used to reach the peer. Use the transform set name you configured in sub-step 2.

```
pixP(config)# crypto map peerQ 10 set transform-set pixQ
```

(where Q = peer pod number)

6. Apply the crypto map set to the outside interface:

```
pixP(config)# crypto map peerQ interface outside
```

(where Q = peer pod number)

## Task 4—Test and Verify IPSec Configuration

Perform the following steps to test and verify VPN configuration:

**Step 1** Verify the IKE policy you just created. Note the default values.

```
pixP(config)# show isakmp
isakmp enable outside
isakmp key ******** address 192.168.Q.2 netmask 255.255.255.255
isakmp policy 10 authentication pre-share
isakmp policy 10 encryption des
isakmp policy 10 hash sha
isakmp policy 10 group 1
isakmp policy 10 lifetime 86400
```

Q 2) What five policy items are configured in an IKE policy?

A) authentication method, encryption algorithm, hash algorithm, DH group, and ISAKMP SA lifetime.

Q 3) Which IKE policy value did you configure in a previous step?

B) authentication method as pre-share

Q 4) Which IKE policy values had defaults?

C) encryption algorithm=des, hash algorithm=sha, D-H group=group 1, and ISAKMP SA lifetime=86400.

**Step 2** Examine the IKE policies in your PIX Firewall.

```
pixP(config)# show isakmp policy
Protection suite of priority 10
        encryption algorithm:   DES - Data Encryption Standard (56 bit keys).
        hash algorithm:         Secure Hash Standard
        authentication method:  Pre-Shared Key
        Diffie-Hellman group:   #1 (768 bit)
        lifetime:               86400 seconds, no volume limit
Default protection suite
        encryption algorithm:   DES - Data Encryption Standard (56 bit keys)
        hash algorithm:         Secure Hash Standard
        authentication method: Rivest-Shamir-Adleman Signature
        Diffie-Hellman group:   #1 (768 bit)
        lifetime:               86400 seconds, no volume limit
```

Q 5) How did the default protection suite get configured?

D) It is part of the default configuration.

**Step 3** Verify the crypto ACL. The list shown is for PIX2 connecting to PIX1:

```
pix2(config)# show access-list
access-list 101 permit ip host 192.168.P.10 host 192.168.Q.10 (hitcnt=0)
```

**Step 4** Verify correct IPSec parameters (IKE phase two):

```
pixP(config)# show crypto ipsec transform-set
Transform set pixQ: { esp-des  }
   will negotiate = { Tunnel,  },
```

**Step 5** Verify correct crypto map configuration. The crypto map shown is for PIX1:

```
pix1(config)# show crypto map
Crypto Map: “peerQ” interfaces: { outside }
Crypto Map “peerQ” 10 ipsec-isakmp
  Peer = 192.168.Q.2
  access-list 101 permit ip host 192.168.P.10 host 192.168.Q.10 (hitcnt=0)

  Current peer: 192.168.Q.2
  Security association lifetime: 4608000 kilobytes/28800 seconds
  PFS (Y/N): N
  Transform sets={ pixQ, }
```

**Step 6** Turn on debugging for IPSec and ISAKMP:

```
pixP(config)# debug crypto ipsec
pixP(config)# debug crypto isakmp
```

**Step 7** Clear the IPSec SA, by using the following command:

```
pixP(config)# clear crypto ipsec sa
```

**Step 8** Initiate a Web session from your internal Windows NT Server to the internal Windows NT server 192.168.Q.10 of an opposite pod group. Observe the debug output and verify the Web session was established. The debug should state the following status indicating that IPSec was successful:

```
return status is IKMP_NO_ERROR
```

**Step 9** Ensure that traffic between peers is being encrypted by performing the following sub-steps:

1. Examine the IPSec SAs. Note the number of packets encrypted and decrypted:

```
pix1(config)# show crypto ipsec sa
interface: outside
    Crypto map tag: peerQ, local addr. 192.168.P.2

   local  ident (addr/mask/prot/port): (192.168.P.10/255.255.255.255/0/0)
   remote ident (addr/mask/prot/port): (192.168.Q.10/255.255.255.255/0/0)
   current_peer: 192.168.Q.2
     PERMIT, flags={origin_is_acl,}
    #pkts encaps: 30, #pkts encrypt: 30, #pkts digest 0
    #pkts decaps: 27, #pkts decrypt: 27, #pkts verify 0
    #pkts compressed: 0, #pkts decompressed: 0
    #pkts not compressed: 0, #pkts compr. failed: 0, #pkts decompress failed: 0
    #send errors 0, #recv errors 0

     local crypto endpt.: 192.168.P.2, remote crypto endpt.: 192.168.Q.2
     path mtu 1500, ipsec overhead 44, media mtu 1500
     current outbound spi: 381fb0b1


     inbound esp sas:
      spi: 0xf690ef14(4136693524)
        transform: esp-des ,
        in use settings ={Tunnel, }
        slot: 0, conn id: 1, crypto map: peerQ
        sa timing: remaining key lifetime (k/sec): (4607996/28250)
        IV size: 8 bytes
        replay detection support: N

     inbound ah sas:



     inbound pcp sas:



     outbound esp sas:
      spi: 0x381fb0b1(941600945)
        transform: esp-des ,
        in use settings ={Tunnel, }
        slot: 0, conn id: 2, crypto map: peerQ
        sa timing: remaining key lifetime (k/sec): (4607996/28241)
        IV size: 8 bytes
        replay detection support: N


     outbound ah sas:


     outbound pcp sas:
```

2. Generate additional traffic by clicking the **Reload** button of your web browser.
3. Examine the IPSec SAs again. Note that the packet counters have increased incrementally.

```
pix2(config)# show cry ipsec sa
interface: outside
    Crypto map tag: peerQ, local addr. 192.168.P.2

   local  ident (addr/mask/prot/port): (192.168.P.10/255.255.255.255/0/0)
   remote ident (addr/mask/prot/port): (192.168.Q.10/255.255.255.255/0/0)
   current_peer: 192.168.Q.2
     PERMIT, flags={origin_is_acl,}
    #pkts encaps: 54, #pkts encrypt: 54, #pkts digest 0
    #pkts decaps: 47, #pkts decrypt: 47, #pkts verify 0
    #pkts compressed: 0, #pkts decompressed: 0
    #pkts not compressed: 0, #pkts compr. failed: 0, #pkts decompress failed: 0
    #send errors 0, #recv errors 0


     local crypto endpt.: 192.168.P.2, remote crypto endpt.: 192.168.Q.2
     path mtu 1500, ipsec overhead 44, media mtu 1500
     current outbound spi: 381fb0b1

     inbound esp sas:
      spi: 0xf690ef14(4136693524)
        transform: esp-des ,
        in use settings ={Tunnel, }
        slot: 0, conn id: 1, crypto map: peerQ
        sa timing: remaining key lifetime (k/sec): (4607993/27908)
```

```
         IV size: 8 bytes
         replay detection support: N


      inbound ah sas:


      inbound pcp sas:


      outbound esp sas:
       spi: 0x381fb0b1(941600945)
         transform: esp-des ,
         in use settings ={Tunnel, }
         slot: 0, conn id: 2, crypto map: peerQ
         sa timing: remaining key lifetime (k/sec): (4607993/27908)
         IV size: 8 bytes
         replay detection support: N


      outbound ah sas:


      outbound pcp sas:
```

**Step 10** Clear your IPSec SAs with the **clear crypto sa** command:

```
pixP(config)# clear crypto sa
```

**Step 11** Remove all isakmp command statements from your configuration with the **clear isakmp** command:

```
pixP(config)# clear isakmp
```

**Step 12** Remove all parameters entered through the crypto map command with the **clear crypto map** command:

```
pixP(config)# clear crypto map
```

**Step 13** Remove the sysopt command statements from your configuration with the **clear sysopt** command:

```
pixP(config)# clear sysopt
```

**Step 14** Remove ACL 101 from your configuration:

```
pixP(config)# clear access-list 101
```

# Configure a Secure VPN Using IPSec Between a PIX Firewall and a VPN Client

## Objectives

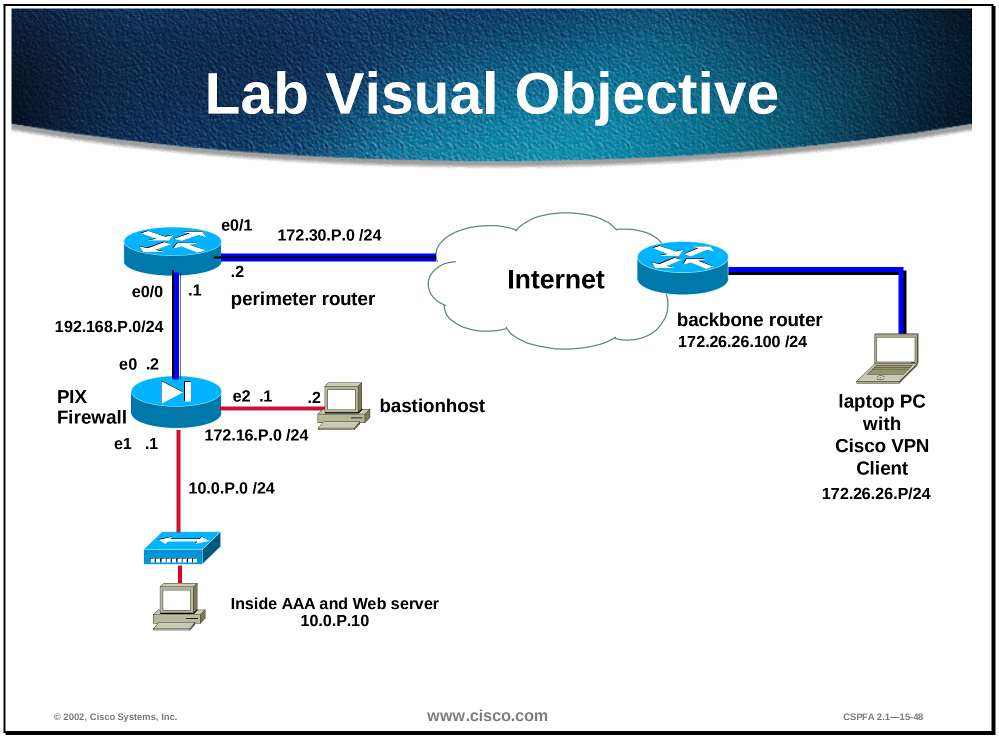
In this lab exercise, you will complete the following tasks:

- Install and configure the Cisco VPN Client, Release 3.1 on an MS-Windows end-user PC.

- Configure the PIX Firewall for Cisco VPN Client remote access.

## Visual Objective

The following figure displays the configuration you will complete in this lab exercise.



Lab Visual Objective

© 2002, Cisco Systems, Inc. www.cisco.com CSPFA 2.1—15-48

## Scenario

Your company wants to implement a virtual private network (VPN) using remotely located Cisco VPN Clients terminating at your PIX Firewall. You must configure both the remote VPN Client and the PIX Firewall for remote access using pre-shared keys for authentication.

## Setup

Before starting this lab exercise, set up your equipment as follows:

- Ensure that your perimeter router is powered on.
- Ensure that your PIX Firewall is powered on.
- Ensure that your laptop PC is powered on and Windows 2000 is operational. Your instructor will provide you with the correct username and password to log into the laptop PC.

## Directions

Your task in this lab exercise is to install and configure the Cisco VPN Client and configure the PIX Firewall to enable IPSec encrypted tunnels using pre-shared keys. Work with your lab partner to perform the following tasks:

- Configure the laptop PC networking parameters.
- Configure the PIX Firewall.
- Verify your configuration.
- Install the Cisco VPN Client.
- Configure the Cisco VPN Client.
- Verify the Cisco VPN Client properties.
- Launch the Cisco VPN Client.
- Verify the VPN connection.

## Task 1—Configure the Laptop PC Networking Parameters

Certain networking parameters must be configured before your student laptop PC will operate in the lab environment. Complete the following steps to configure your student laptop networking parameters. This procedure assumes that Windows 2000 Server is operating with an active Network Interface Card (NIC):

**Step 1** Right-click the **My Network Places** desktop icon and choose **Properties**. The Network and Dial-up Connections window opens.

**Step 2** Double-click the **Local Area Connection** icon. The Local Area Connection Status window opens.

**Step 3** Click **Properties**. The Local Area Connection Properties window opens.

**Step 4** Select the Internet Protocol (TCP/IP) component and click **Properties**. The Internet Protocol (TCP/IP) Properties window opens.

**Step 5** Select the **Use the following IP address** option.

**Step 6** Enter the laptop primary IP address, **172.26.26.*P***, in the IP Address field.

(where P = pod number)

**Step 7** Enter the laptop primary subnet mask, **255.255.255.0**, in the Subnet Mask field.

**Step 8** Enter the backbone router IP address, **172.26.26.100**, in the Default Gateway field.

**Step 9** Click **OK** to close the Internet Protocol (TCP/IP) Properties window.

**Step 10** Click **OK** to close the Local Area Connection Properties window. It may take a few moments for the window to close.

**Step 11** Close the Local Area Connection Status window.

**Step 12** Close the Network and Dial-up Connections window.

**Step 13** Open a command prompt session by double-clicking the **Command Prompt** desktop icon.

**Step 14** Ping the backbone router IP address, **172.26.26.100**. If you cannot ping the backbone router, review the steps in Task 1.

**Step 15** Close the Command Prompt session window.

## Task 2—Configure the PIX Firewall

The instructor will provide you with the procedures for access to the PIX Firewall console port. After you access the PIX Firewall console port, enter configuration mode, and complete the following steps to configure the PIX Firewall:

**Step 1** Set up a pool of IP addresses that will dynamically be assigned to the VPN clients via IKE mode configuration:

```
pixP(config)# ip local pool dealer 10.0.P.20-10.0.P.29
```

(where P = pod number)

**Step 2** Create a static translation that maps addresses from the local pool to addresses on the internal network:

```
pixP(config)# static (inside,outside) 10.0.P.0 10.0.P.0 netmask 255.255.255.0
```

(where P = pod number)

**Step 3** Configure the PIX Firewall for TACACS by completing the following sub-steps:

1. Create a group tag called MYTACACS and assign the TACACS+ protocol to it.

```
pixP(config)# aaa-server MYTACACS protocol tacacs+
```

2. Assign the CS ACS IP address and the encryption key secretkey.

```
pixP(config)# aaa-server MYTACACS (inside) host 10.0.P.10 secretkey timeout 5
```

(where P = pod number)

**Step 4** Enable the PIX Firewall to implicitly permit any packet from an IPSec tunnel and bypass checking with an associated conduit or access-group command for IPSec connections:

```
pixP(config)# sysopt connection permit-ipsec
```

**Step 5** Set up a transform set that will be used for the VPN Clients:

```
pixP(config)# crypto ipsec transform-set AAADES esp-des esp-md5-hmac
```

**Step 6** Set up a dynamic crypto map to enable the VPN Clients to connect to the PIX Firewall:

```
pixP(config)# crypto dynamic-map DYNOMAP 10 set transform-set AAADES
```

**Step 7** Create a crypto map, and assign the dynamic crypto map to it:

```
pixP(config)# crypto map VPNPEER 20 ipsec-isakmp dynamic DYNOMAP
```

**Step 8** Configure Xauth to point to the TACACS+ server:

```
pixP(config)# crypto map VPNPEER client authentication MYTACACS
```

**Step 9** Apply the crypto map to the PIX Firewall interface:

```
pixP(config)# crypto map VPNPEER interface outside
```

**Step 10** Enable IKE on the outside interface:

```
pixP(config)# isakmp enable outside
```

**Step 11** Configure IKE mode configuration related parameters:

```
pixP(config)# isakmp client configuration address-pool local dealer outside
```

**Step 12** Configure the ISAKMP policy by completing the following sub-steps:

1. Configure a basic IKE policy using pre-shared keys for authentication:

```
pixP(config)# isakmp policy 10 authentication pre-share
```

2. Specify the encryption algorithm:

```
pixP(config)# isakmp policy 10 encryption des
```

Specify the hash algorithm:

```
pixP(config)# isakmp policy 10 hash md5
```

3. Specify the Diffie-Hellman (DH) by group identifier:

```
pixP(config)# isakmp policy 10 group 2
```

4. Specify the IKE Security Association's (SA's) lifetime:

```
pixP(config)# isakmp policy 10 lifetime 86400
```

**Step 13** Configure the VPN group to support pushing mode configuration parameters to the VPN Client. The VPN group name of *training* must match the group name in the VPN Client. The VPN group password must match the password in the VPN Client. Complete the following sub-steps:

1. Configure the IP address pool name:

```
pixP(config)# vpngroup training address-pool dealer
```

2. Configure the inactivity timeout in seconds:

```
pixP(config)# vpngroup training idle-time 1800
```

3. Configure the VPN group password:

```
pixP(config)# vpngroup training password training
```

## Task 3—Verify Your Configuration

Complete the following steps to verify your PIX Firewall's configuration:

**Step 1** Verify your ip local pool:

```
pixP(config)# sh ip local pool
Pool            Begin           End               Free    In use
dealer          10.0.P.20       10.0.P.29           10         0
Available Addresses:
10.0.P.20
10.0.P.21
10.0.P.22
10.0.P.23
10.0.P.24
10.0.P.25
10.0.P.26
10.0.P.27
10.0.P.28
10.0.P.29
```

**Step 2** Verify your Network Address Translation (NAT) configuration:

```
pixP(config)# sh nat
nat (inside) 1 0.0.0.0 0.0.0.0 0 0
```

**Step 3** Verify your statics:

```
pixP(config)# show static
static (dmz,outside) 192.168.P.11 172.16.P.2 netmask 255.255.255.255 0 0
static (inside,outside) 10.0.P.0 10.0.P.0 netmask 255.255.255.0 0 0
```

**Step 4** Verify your authentication, authorization, and accounting (AAA) server configuration:

```
pixP(config)# show aaa-server
aaa-server TACACS+ protocol tacacs+
aaa-server RADIUS protocol radius
aaa-server MYTACACS protocol tacacs+
aaa-server MYTACACS (inside) host 10.0.P.10 secretkey timeout 5
```

**Step 5** Verify your crypto map:

```
pixP(config)# show crypto map
Crypto Map: "VPNPEER" interfaces: { outside }
        client authentication MYTACACS


Crypto Map "VPNPEER" 20 ipsec-isakmp
        Dynamic map template tag: dynomap
```

**Step 6** Verify your transform set:

```
pixP(config)# show crypto ipsec transform-set
Transform set AAADES: { esp-des esp-md5-hmac  }
   will negotiate = { Tunnel,  },
```

**Step 7** Verify your IKE policy:

```
pixfirewall(config)# show isakmp policy
Protection suite of priority 10
        encryption algorithm:   DES - Data Encryption Standard (56 bit keys).
        hash algorithm:         Message Digest 5
        authentication method:  Pre-Shared Key
        Diffie-Hellman group:   #2 (1024 bit)
        lifetime:               86400 seconds, no volume limit
Default protection suite
        encryption algorithm:   DES - Data Encryption Standard (56 bit keys).
        hash algorithm:         Secure Hash Standard
        authentication method:  Rivest-Shamir-Adleman Signature
        Diffie-Hellman group:   #1 (768 bit)
        lifetime:               86400 seconds, no volume limit
```

**Step 8** Verify your VPN group configuration:

```
pixfirewall(config)# show vpn group
vpngroup training address-pool dealer
vpngroup training idle-time 1800
vpngroup training password ********
```

## Task 4—Install the Cisco VPN Client

In this lab exercise, the source files for the Cisco VPN Client already reside on the hard disk drive of the laptop PC. Complete the following steps to install the Cisco VPN Client:

**Step 1** Open the Cisco VPN Client folder found on the laptop PC desktop.

**Step 2** Double-click the setup.exe file from the Cisco VPN Client folder. The Cisco Systems VPN Client Setup window opens.

**Step 3** Click **Next**. The License Agreement window opens.

**Step 4** Read the license agreement and click **Yes**. You are prompted to choose a destination location.

**Step 5** Accept the default destination folder by clicking **Next**. The Select Program Folder window opens.

**Step 6** Accept the defaults by clicking **Next**. The Start Copying Files window opens.

**Step 7** The files are copied to the hard disk drive of the Windows 2000 PC and the InstallShield Wizard Complete window opens.

**Step 8** Select **Yes, I want to restart my computer now** and click **Finish**. The Windows 2000 laptop restarts.

**Step 9** Log into the laptop PC.

## Task 5—Configure the Cisco VPN Client

Use the following procedure to configure the networking parameters of the new Cisco VPN Client. This procedure assumes Windows 2000 is already running.

**Step 1** Choose **Start>Programs>Cisco Systems VPN Client>VPN Dialer**. The Cisco Systems VPN Client window opens.

**Step 2** Click **New**. The New Connection Entry wizard opens.

**Step 3** Enter **vpnpeer*P*** as the name for the new connection entry in the Name of the New Connection Entry field.

(where P = pod number)

**Step 4** Click **Next**.

**Step 5** Enter the PIX Firewall’s public interface IP address, **192.168.P.2**, as the IP address of the server.

(where P = pod number)

**Step 6** Click **Next**.

**Step 7** Select **Group Access Information** and complete the following sub-steps. The following entries are always case-sensitive. Use lower-case characters for this lab exercise.

1. Enter a group name: **training**.
2. Enter a group password: **training**.
3. Confirm the password: **training**.

**Step 8** Click **Next**.

**Step 9** Click **Finish** and leave the Cisco Systems VPN Client window open.

## Task 6—Verify the Cisco VPN Client Properties

Complete the following steps to verify the Cisco VPN Client parameters you just configured:

**Step 1** Ensure that the Cisco Systems VPN Client window is open. If the Cisco Systems VPN Client window is not open, choose **Start>Programs>Cisco Systems VPN Client>VPN Dialer**.

**Step 2** Select **vpnpeer*P*** within the Connection Entry group box.

(where P = your pod number)

**Step 3** Verify that the IP address of the remote server is set to the PIX Firewall's public interface IP address, 192.168.P.2.

**Step 4** Click **Options**. A popup menu opens.

**Step 5** Choose **Properties**. The Properties for vpnpeer*P* window opens.

(where P = pod number)

**Step 6** Select the **General** tab and view the available options. Do not make any changes to the default settings.

**Step 7** Select the **Authentication** tab and verify the spelling of the group name. If you needed to, you can edit the group name and password here.

**Step 8** Select the **Connections** tab and view the available options. Do not make any changes to the default settings.

**Step 9** Click **OK**.

**Step 10** Close the Cisco Systems VPN Client window.

## Task 7—Launch the Cisco VPN Client

Complete the following steps to launch the Cisco VPN Client on your laptop PC:

**Step 1** Choose Start>Programs>Cisco Systems VPN Client>VPN Dialer.

**Step 2** Verify that the Connection Entry is vpnpeer*P*.

(where P = pod number)

**Step 3** Verify that the IP address of remote server is set to the PIX Firewall's public interface IP address, 192.168.P.2.

**Step 4** Click **Connect**. The Connection History window opens, and several messages flash by quickly. Complete the following sub-steps:

1. When prompted for a username, enter: **student*P***.

(where P = pod number)

2. When prompted to enter a password, enter **training**.

**Step 5** Click **OK**. The following messages flash by quickly:

- Initializing the connection

- Contacting the security gateway at
- Authenticating user

The window disappears and a VPN (lock) icon appears in the system tray. You have successfully launched the Cisco VPN Client.

## Task 8—Verify the VPN Connection

Complete the following steps to verify the IPSec connection:

**Step 1** Test access to the inside Web server from the remote client by completing the following sub-steps:

1. Open a web browser on the VPN Client PC.
2. Use the web browser to access the inside Web server by entering: **http://10.0.P.10**

(where P = pod number)

3. The web server's home page should display.

**Step 2** Double-click the VPN Dialer icon in the system tray and answer the following questions:

Q 1) What window opened?

E) Cisco Systems VPN Client Connection Status

Q 2) What encryption scheme was used?

F) 56-bit DES

Q 3) What authentication method was used?

G) HMAC-MD5

Q 4) What IP Address was assigned to you?

H) 10.0.P.20

**Step 3** Click the **Statistics** tab and view the information provided. Notice the number of packets encrypted and decrypted.

**Step 4** Refresh your browser.

**Step 5** Return to the Cisco Systems VPN Client Connection Status window and notice that the number of packets encrypted and decrypted has incremented.

**Step 6** Click **OK**.

**Step 7** Right-click the Cisco icon on the system tray. Answer the following question:

Q 5) What do you see in the window?

I) Status, Notifications, Disconnect, About

**Step 8** Select **Status**. Answer the following question:

Q 6) What window opens?

J) Cisco Systems VPN Client Connection Status

**Step 9** Click **OK** to close the window.

**Step 10** Disconnect your VPN dialer session using the system tray VPN dialer icon.

**Step 11** Remove the crypto map from the PIX Firewall's outside interface:

```
pixP(config)# no crypto map vpnpeer interface outside
```

# Completion Criteria

You completed this lab exercise if you were able to do the following:

- Cause an IPSec tunnel to be established between PIX Firewalls
- Closely match your PIX Firewall configuration with the example configuration at the end of this lab exercise

# Example Configurations

The following tables show an example configuration for PIX1 and PIX2. You may experience differences between the example configuration and your own configuration.

## PIX1 Example Configuration

The example in the following table is a summary of the configuration for PIX1.

**Table 12-1. PIX1 Example Configuration**

| Example Configuration | Description |
|---|---|
| **ip address outside 192.168.1.2 255.255.255.0**<br>**ip address inside 10.0.1.1 255.255.255.0**<br>**ip address dmz 172.16.1.1 255.255.0.0** | Configures the IP addresses for each PIX Firewall interface. |
| **global (outside) 1 192.168.1.10-192.168.1.254 netmask 255.255.255.0** | Creates a global pool on the outside interface. |
| **nat (inside) 1 10.0.0.0 0.0.0.0 0 0** | Enables NAT for the inside interface. |
| **static (inside,outside) 192.168.1.10 10.0.1.3 netmask 255.255.255.255 0 0** | Creates a static translation between the global IP address of 192.168.1.10 and the inside Windows NT server at address 10.0.1.3. |
| **access-list 101 permit ip host 192.168.1.10 host 192.168.2.10** | The crypto ACL specifies that traffic between the internal Windows NT servers of PIX1 and PIX2 be encrypted. The source and destination IP addresses are the global IP addresses of the static translations. Note that the ACLs for PIX1 and PIX2 are mirror images of each other. |
| **conduit permit icmp any any**<br>**conduit permit tcp host 192.168.1.10 eq www any** | The conduits permit ICMP and Web access for testing. |
| **route outside 0.0.0.0 0.0.0.0 192.168.1.1 1** | Specifies the router on the outside interface for the default route. |
| **sysopt connection permit-ipsec** | Enables IPSec to bypass ACL, access, and conduit restrictions. |

| Example Configuration | Description |
|---|---|
| **crypto ipsec transform-set pix2 esp-des** | Defines a crypto map transform set named pix2 to use esp-des. |
| **crypto map peer2 10 ipsec-isakmp** | Defines the crypto map named peer2 with a priority of 10 to use ISAKMP access. The crypto map defines IPSec (IKE phase two) parameters. |
| **crypto map peer2 10 match address 101** | Defines the crypto map named peer2 to use ACL 101 for crypto traffic selection. |
| **crypto map peer2 10 set peer 192.168.2.2** | Defines the crypto map named peer2 to point to the peer (pix2) by specifying the peer PIX Firewall's outside interface IP address. |
| **crypto map peer2 10 set transform-set pix2** | Defines the crypto map named peer2 to use the transform set named pix2. |
| **crypto map peer2 interface outside** | Assigns the crypto map set named peer2 to the outside PIX Firewall interface. As soon as the crypto map is assigned to the interface, the IKE and IPSec policy is active. |
| **isakmp enable outside** | Enables ISAKMP (IKE) on the outside interface. |
| **isakmp key cisco123 address 192.168.2.2 netmask 255.255.255.255** | Defines the pre-shared IKE key of cisco123 to work with the IPSec peer at address 192.168.2.2. The address points to the peer's outside interface. A wildcard address of 0.0.0.0 with a netmask of 0.0.0.0 could also have been used. |
| **isakmp policy 10 authentication pre-share** | Defines the ISAKMP (IKE) policy of 10 to use pre-shared keys for authentication. |
| **isakmp policy 10 encryption des** | Defines the ISAKMP (IKE) policy of 10 to use DES encryption. Could have used 3DES for stronger encryption. |
| **isakmp policy 10 hash sha** | Defines the ISAKMP (IKE) policy of 10 to use the SHA-1 hashing algorithm for encryption. |
| **isakmp policy 10 group 1** | Specifies use of DH group 1. Could have used DH group 2 for stronger security, but requires more CPU time to execute. |
| **isakmp policy 10 lifetime 86400** | Specifies an ISAKMP (IKE) lifetime of 86,400 seconds. |

## PIX2 Example Configuration

The example in the following table is a summary of the configuration for PIX2.

**Table 12-2. PIX2 Example Configuration**

| Example Configuration | Description |
|---|---|
| **ip address outside 192.168.2.2 255.255.255.0**<br>**ip address inside 10.0.2.1 255.255.255.0**<br>**ip address dmz 172.16.2.1 255.255.0.0** | Configures the IP addresses for each PIX Firewall interface. |
| **global (outside) 1 192.168.2.10-192.168.2.254 netmask 255.255.255.0** | Creates a global pool on the outside interface. |
| **nat (inside) 1 10.0.0.0 0.0.0.0 0 0** | Enables NAT for the inside interface. |

| Example Configuration | Description |
|---|---|
| **static (inside,outside) 192.168.2.10 10.0.2.3 netmask 255.255.255.255 0 0** | Creates a static translation between the global IP address of 192.168.2.10 and the inside Windows NT server at address 10.0.2.3. |
| **access-list 101 permit ip host 192.168.2.10 host 192.168.1.10** | The crypto ACL specifies that traffic between the internal Windows NT servers of PIX1 and PIX2 be encrypted. The source and destination IP addresses are the global IP addresses of the static translations. Note that the ACLs for PIX1 and PIX2 are mirror images of each other. |
| **conduit permit icmp any any**<br>**conduit permit tcp host 192.168.2.10 eq www any** | The conduits permit ICMP and Web access for testing. |
| **route outside 0.0.0.0 0.0.0.0 192.168.2.1 1** | Specifies the router on the outside interface for the default route. |
| **sysopt connection permit-ipsec** | Enables IPSec to bypass ACL, access, and conduit restrictions. |
| **crypto ipsec transform-set pix1 esp-des** | Defines a crypto map transform set named pix1 to use esp-des. |
| **crypto map peer1 10 ipsec-isakmp** | Defines the crypto map named peer1 with a priority of 10 to use ISAKMP access. The crypto map defines IPSec (IKE phase two) parameters. |
| **crypto map peer1 10 match address 101** | Defines the crypto map named peer1 to use ACL 101 for crypto traffic selection. |
| **crypto map peer1 10 set peer 192.168.1.2** | Defines the crypto map named peer1 to point to the peer (pix1) by specifying the peer PIX Firewall's outside interface IP address. |
| **crypto map peer1 10 set transform-set pix1** | Defines the crypto map named peer1 to use the transform set named pix1. |
| **crypto map peer1 interface outside** | Assigns the crypto map set named peer1 to the outside PIX Firewall interface. As soon as the crypto map is assigned to the interface, the IKE and IPSec policy is active. |
| **isakmp enable outside** | Enables ISAKMP (IKE) on the outside interface. |
| **isakmp key cisco123 address 192.168.1.2 netmask 255.255.255.255** | Defines the pre-shared IKE key of cisco123 to work with the IPSec peer at address 192.168.1.2. The address points to the peer's outside interface. A wildcard address of 0.0.0.0 with a netmask of 0.0.0.0 could also have been used. |
| **isakmp policy 10 authentication pre-share** | Defines the ISAKMP (IKE) policy of 10 to use pre-shared keys for authentication. |
| **isakmp policy 10 encryption des** | Defines the ISAKMP (IKE) policy of 10 to use DES encryption. Could have used 3DES for stronger encryption. |
| **isakmp policy 10 hash sha** | Defines the ISAKMP (IKE) policy of 10 to use the SHA-1 hashing algorithm for encryption. |
| **isakmp policy 10 group 1** | Specifies use of DH group 1. Could have used DH group 2 for stronger security, but requires more CPU time to execute. |
| **isakmp policy 10 lifetime 86400** | Specifies an ISAKMP (IKE) lifetime of 86,400 seconds. |

16

# System Maintenance

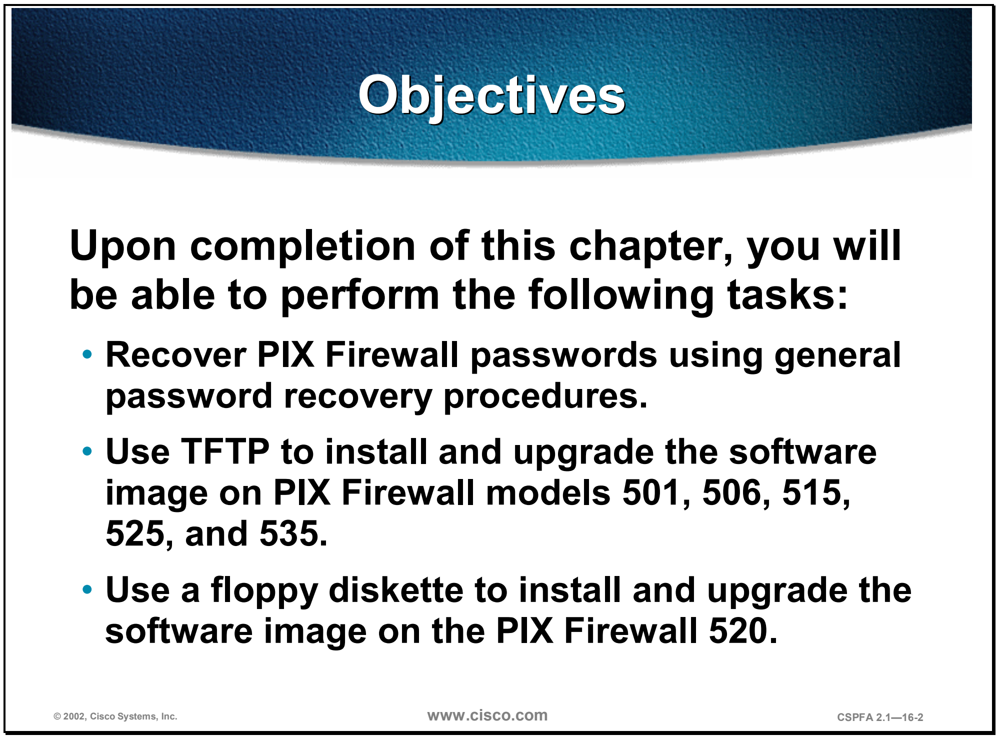
## Overview

This chapter includes the following topics:

- Objectives
- Password recovery
- Image upgrade
- Summary
- Lab exercise

# Objectives

This section lists the chapter's objectives.



## Objectives

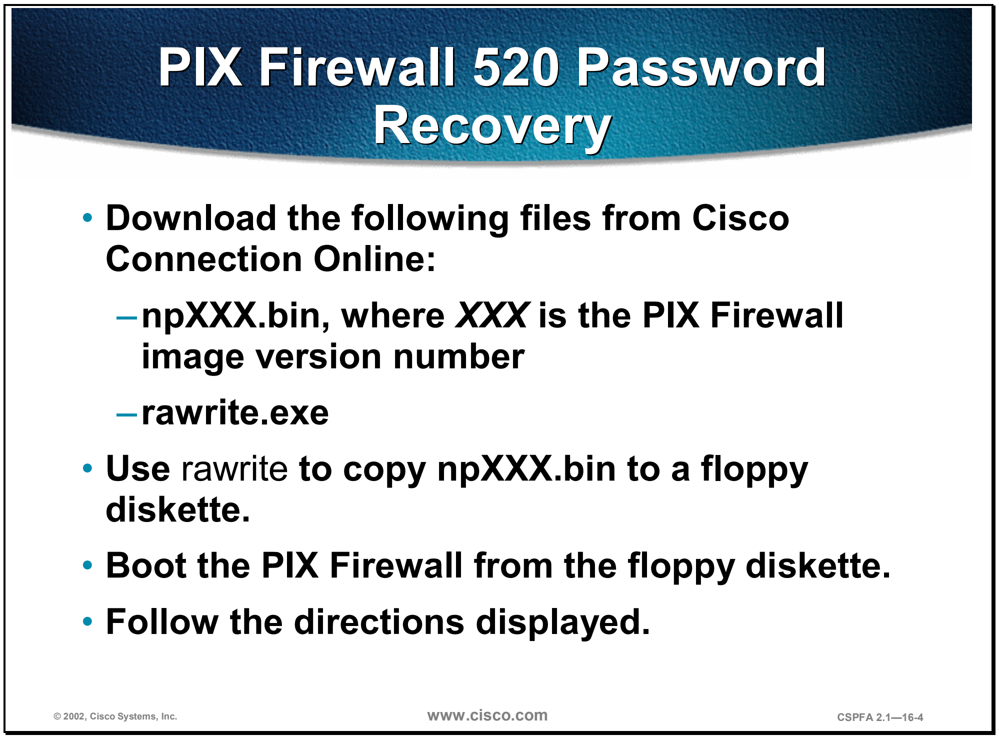
**Upon completion of this chapter, you will be able to perform the following tasks:**

- **Recover PIX Firewall passwords using general password recovery procedures.**
- **Use TFTP to install and upgrade the software image on PIX Firewall models 501, 506, 515, 525, and 535.**
- **Use a floppy diskette to install and upgrade the software image on the PIX Firewall 520.**

www.cisco.com CSPFA 2.1—16-2

# Password Recovery

This section explains the how to perform password recovery on various Cisco Secure PIX™ Firewall models.



## PIX Firewall 520 Password Recovery

- **Download the following files from Cisco Connection Online:**
  - **npXXX.bin, where *XXX* is the PIX Firewall image version number**
  - **rawrite.exe**
- **Use** rawrite **to copy npXXX.bin to a floppy diskette.**
- **Boot the PIX Firewall from the floppy diskette.**
- **Follow the directions displayed.**

© 2002, Cisco Systems, Inc. www.cisco.com CSPFA 2.1—16-4

The password recovery for the PIX Firewall 520 requires writing a special image to a floppy diskette. Use this diskette to boot PIX Firewall 520. To perform a PIX Firewall 520 password recovery, complete the following:

**Step 1** Download the file for the PIX Firewall software version you are running from the Cisco Connection Online (each version requires a different file): ftp://ftp.cisco.com /cisco/internet/pix/special/your version. You will need a Cisco Connection Online login to download this data.

**Step 2** Download the rawrite.exe file into the same directory as the password version you downloaded previously.

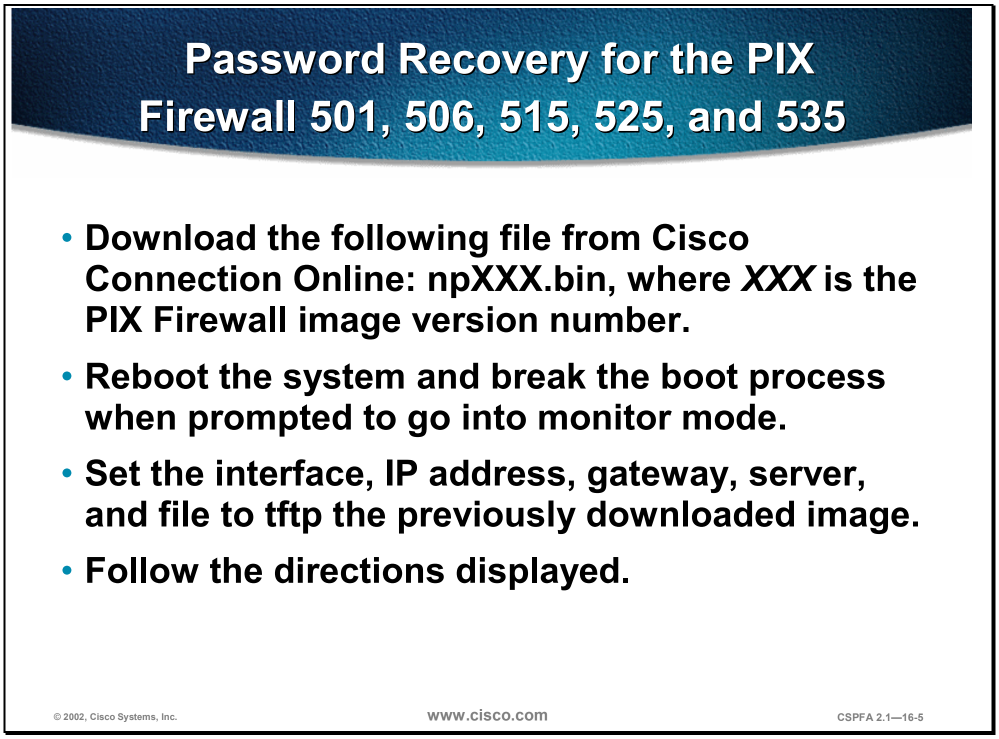
**Step 3** After you have retrieved the two files, execute the rawrite.exe file as follows and enter the information when prompted:

```
C:\> rawrite
RaWrite 1.2 - Write disk file to a floppy diskette
Enter the source file name: npXXX.bin
 (where XXX=version number)
Enter the destination drive: a:
Please insert a formatted diskette into drive A: and press -ENTER- : <Enter>
Number of sectors per track for this disk is 18
Writing image to drive A:. Press ^C to abort.
Track: 78 Head: 1 Sector: 16
Done.
C:\>
```

**Step 4** Reboot your PIX Firewall with the diskette you just created. When prompted, press **y** to erase the password:

```
Do you wish to erase the passwords? [yn] y
Passwords have been erased
```

The system automatically erases the password and starts rebooting.



## Password Recovery for the PIX Firewall 501, 506, 515, 525, and 535

- Download the following file from Cisco Connection Online: npXXX.bin, where *XXX* is the PIX Firewall image version number.
- Reboot the system and break the boot process when prompted to go into monitor mode.
- Set the interface, IP address, gateway, server, and file to tftp the previously downloaded image.
- Follow the directions displayed.

© 2002, Cisco Systems, Inc. www.cisco.com CSPFA 2.1—16-5

The password recovery for PIX Firewall models 501, 506, 515, 525, and 535 requires a TFTP sever. To perform a password recovery using TFTP, complete the following steps:

**Step 1** Download the file for the PIX Firewall software version you are running from the Cisco Connection Online (each version requires a different file): ftp://ftp.cisco.com /cisco/internet/pix/special/your version. You will need a Cisco Connection Online login to download this data.

**Step 2** Move the binary file you just downloaded to the TFTP home folder on your TFTP server.

**Step 3** Reboot your PIX Firewall and interrupt the boot process to enter monitor mode. To do this, you must press the Escape key or send a break character.

**Step 4** Specify the PIX Firewall interface to use for TFT:

```
monitor> interface [num]
```

**Step 5** Specify the PIX Firewall interface's IP address:

```
monitor> address [IP_address]
```

**Step 6** Specify the default gateway (if needed):

```
monitor> gateway [IP_address]
```

**Step 7** Verify connectivity to the TFTP server:

```
monitor> ping [server_address]
```

**Step 8** Name the server:

```
monitor> server [IP_address]
```

**Step 9** Name the image filename:

```
monitor> file [name]
```

**Step 10** Start the TFTP process:

```
monitor> tftp
```

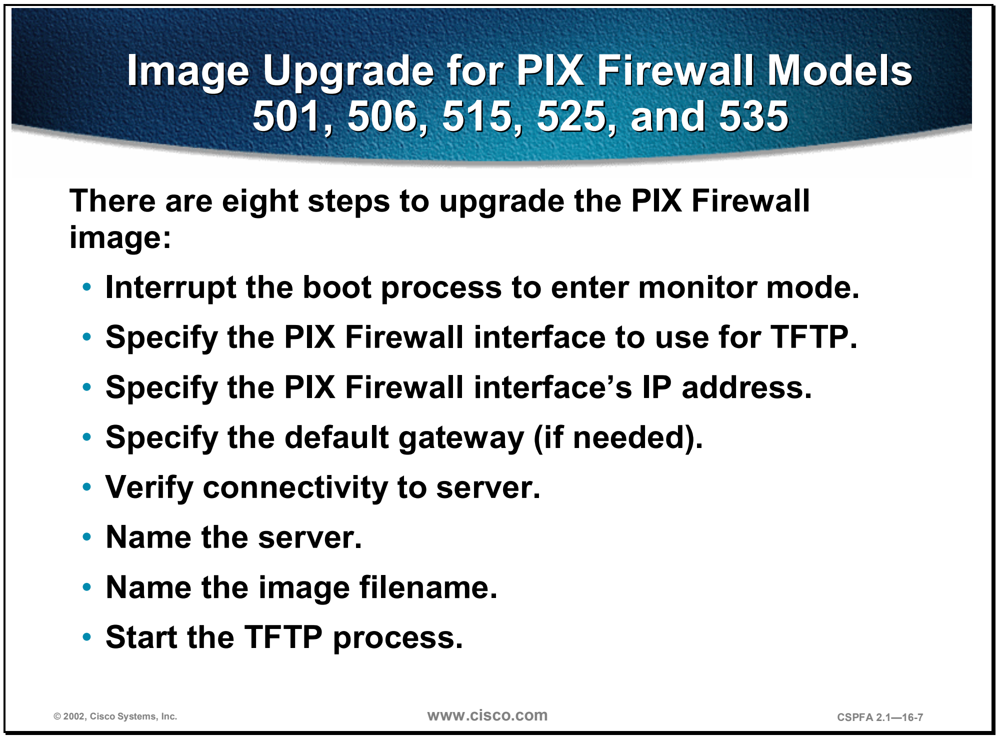
**Step 11** When prompted, press **y** to erase the password:

```
Do you wish to erase the passwords? [yn] y
Passwords have been erased
```

The system automatically erases the password and starts rebooting.

# Image Upgrade

This section explains how to upgrade your PIX Firewall image.



## Image Upgrade for PIX Firewall Models 501, 506, 515, 525, and 535

There are eight steps to upgrade the PIX Firewall image:

- Interrupt the boot process to enter monitor mode.
- Specify the PIX Firewall interface to use for TFTP.
- Specify the PIX Firewall interface's IP address.
- Specify the default gateway (if needed).
- Verify connectivity to server.
- Name the server.
- Name the image filename.
- Start the TFTP process.

© 2002, Cisco Systems, Inc. www.cisco.com CSPFA 2.1—16-7

There are eight steps to upgrade the PIX Firewall image:

**Step 1** Interrupt the boot process to enter monitor mode. To do this, you must press the **Escape** key or send a break character.

**Step 2** Specify the PIX Firewall interface to use for TFTP:

```
monitor> interface [num]
```

**Step 3** Specify the PIX Firewall interface's IP address:

```
monitor> address [IP_address]
```

**Step 4** Specify the default gateway (if needed):

```
monitor> gateway [IP_address]
```

**Step 5** Verify connectivity to the TFTP server:

```
monitor> ping [server_address]
```

**Step 6** Name the server:

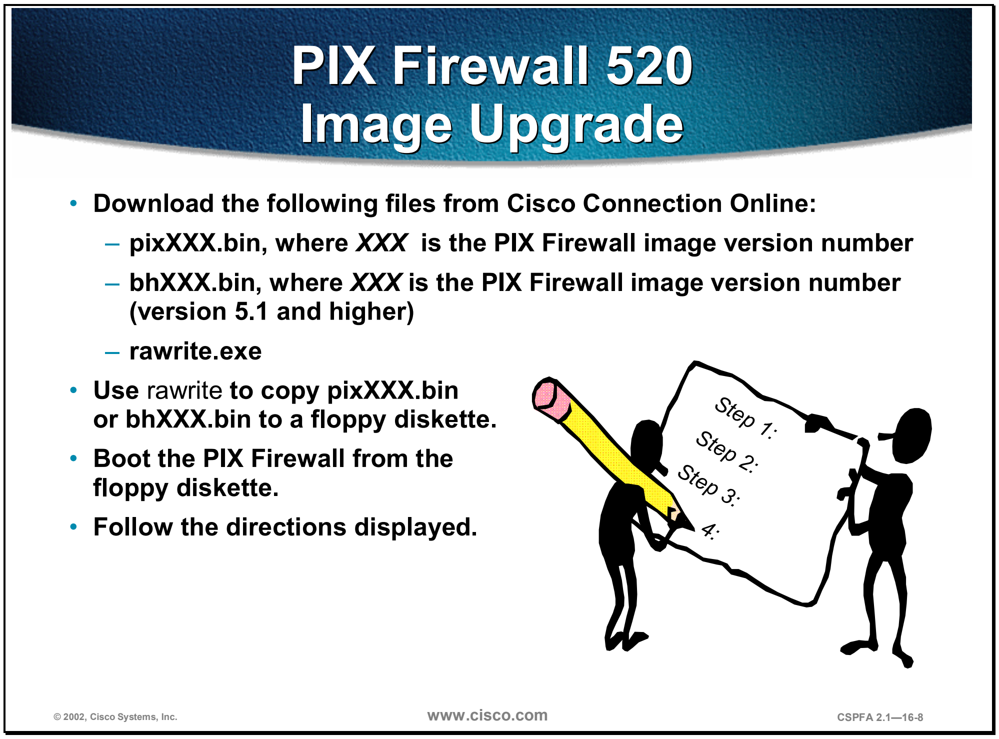
```
monitor> server [IP_address]
```

**Step 7** Name the image filename:

```
monitor> file [name]
```

**Step 8** Start the TFTP process:

```
monitor> tftp
```



## PIX Firewall 520 Image Upgrade

- **Download the following files from Cisco Connection Online:**
  - **pixXXX.bin, where *XXX* is the PIX Firewall image version number**
  - **bhXXX.bin, where *XXX* is the PIX Firewall image version number (version 5.1 and higher)**
  - **rawrite.exe**
- **Use** rawrite **to copy pixXXX.bin or bhXXX.bin to a floppy diskette.**
- **Boot the PIX Firewall from the floppy diskette.**
- **Follow the directions displayed.**

© 2002, Cisco Systems, Inc. www.cisco.com CSPFA 2.1—16-8

For the image upgrade of the PIX Firewall 520 to versions lower than 5.1, complete the following steps:

**Step 9** Download the file for the PIX Firewall software version you are running from the Cisco Connection Online (each version requires a different file).Enter **ftp://ftp.cisco.com /cisco/internet/pix/special/your version** in the URL field of your web browser. You will need a CCO login to download this data.

**Step 10** Download the rawrite.exe file into the same directory as the password version you downloaded previously.

**Step 11** Execute the rawrite.exe file as follows and enter the information when prompted:

```
C:\> rawrite
RaWrite 1.2 - Write disk file to a floppy diskette

Enter the source file name: pixXXX.bin (where XXX=version number)
Enter the destination drive: a:
Please insert a formatted diskette into drive A: and press -ENTER- : <Enter>
Number of sectors per track for this disk is 18
Writing image to drive A:. Press ^C to abort.
Track: 78 Head: 1 Sector: 16
Done.
C:\>
```

**Step 12** Reboot your PIX Firewall with the diskette you just created. The system automatically loads the new image into Flash memory.

**Step 13** Remove the disk after rebooting. You are finished with the upgrade.

For the image upgrade of the PIX Firewall 520 to versions 5.1 and higher, complete the following steps:

**Step 1** Download the file for the PIX Firewall software version you are running from the Cisco Connection Online (each version requires a different file). Enter **ftp://ftp.cisco.com/cisco/internet/pix/special/your version** in the URL field of your web browser. You will need a CCO login to download this data.

**Step 2** Download the boothelper utility file for the software version to which you are upgrading from the Cisco Connection Online (each version requires a different file): Enter **ftp://ftp.cisco.com /cisco/internet/pix/special/your version** in the URL field of your web browser.

**Step 3** Download the rawrite.exe file into the same directory as the password version you downloaded previously.

**Step 4** Execute the rawrite.exe file as follows and enter the information when prompted:

```
C:\> rawrite
RaWrite 1.2 - Write disk file to a floppy diskette

Enter the source file name: bhXXX.bin (where XXX=version number)
Enter the destination drive: a:
Please insert a formatted diskette into drive A: and press -ENTER- : <Enter>
Number of sectors per track for this disk is 18
Writing image to drive A:. Press ^C to abort.
Track: 78 Head: 1 Sector: 16
Done.
C:\>
```

**Step 5** Reboot your PIX Firewall with the diskette you just created. The system automatically loads the boothelper utility from which you will tftp the new image to Flash memory.

**Step 6** At the boothelper prompt, specify the PIX Firewall interface to use for tftp:

```
boothelper> interface [num]
```

**Step 7** Specify the PIX Firewall interface's IP address. To do this, you must enter the following command at the monitor prompt:

```
boothelper> address [IP_address]
```

**Step 8** Specify the default gateway (if needed):

```
boothelper> gateway [IP_address]
```

**Step 9** Verify connectivity to the TFTP server:

```
boothelper> ping [server_address]
```

**Step 10** Name the server:

```
boothelper> server [IP_address]
```

**Step 11** Name the image filename:

```
boothelper> file [name]
```

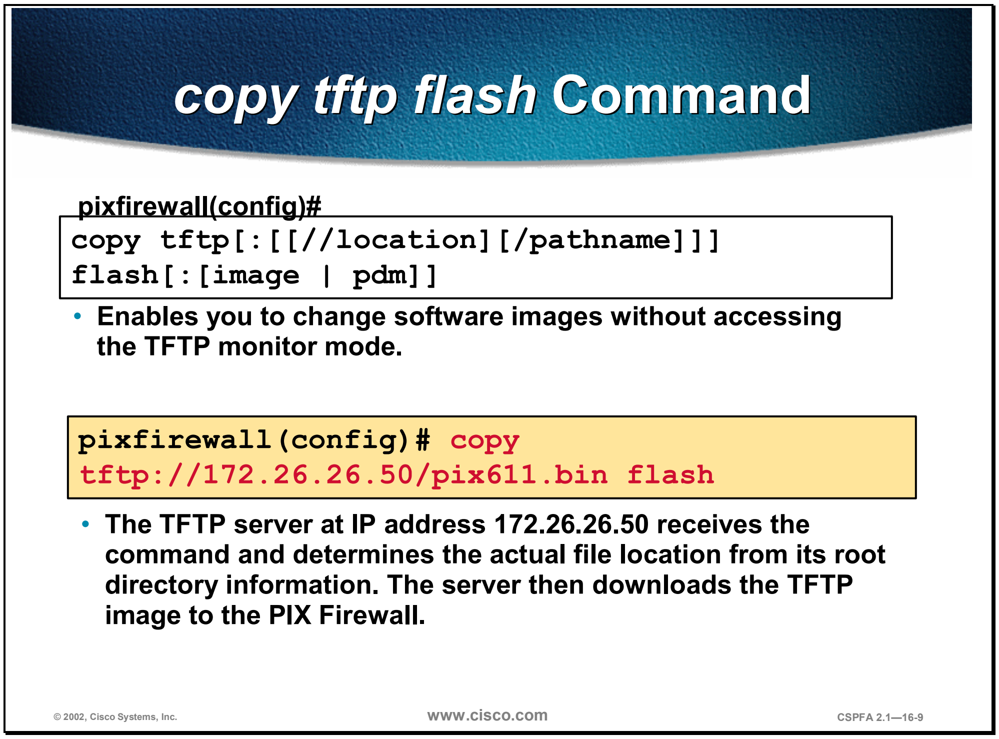
**Step 12** Start the TFTP process:

```
boothelper> tftp
```

**Step 13** The system automatically reboots with the new image in Flash memory. Quickly remove the boothelper diskette when prompted to do so. You are finished with the upgrade.

**Note** The boothelper file is needed because the PIX Firewall software images 5.1 and higher do not fit on a diskette. The boothelper file enables you to use TFTP to load the image. Once your PIX Firewall has been upgraded to version 5.1 or later, it is no longer necessary to use a floppy diskette to load new images. Starting with software version 5.1, the **copy tftp flash** command enables you to TFTP your new image directly to the PIX Firewall from a TFTP server. This command can be used with any PIX Firewall model running PIX Firewall software version 5.1.1 or later.

**Note** Fast Ethernet cards in 64-bit slots are not visible in monitor mode. This problem means that the TFTP server cannot reside on one of these interfaces. Use the **copy tftp flash** command to download the PIX Firewall image file via TFTP.



## *copy tftp flash* Command

pixfirewall(config)#

```
copy tftp[:[[//location][/pathname]]]
flash[:[image | pdm]]
```

- Enables you to change software images without accessing the TFTP monitor mode.

```
pixfirewall(config)# copy
tftp://172.26.26.50/pix611.bin flash
```

- The TFTP server at IP address 172.26.26.50 receives the command and determines the actual file location from its root directory information. The server then downloads the TFTP image to the PIX Firewall.

© 2002, Cisco Systems, Inc. www.cisco.com CSPFA 2.1—16-9

The **copy tftp flash** command enables you to change software images without accessing the TFTP monitor mode. You can use this command to download a software image via TFTP with any PIX Firewall model running version 5.1 or later. The image you download is made available to the PIX Firewall on the next reload.

Be sure to configure your TFTP server to point to the image you wish to download. For example, to download the pix611.bin file from the D: partition on a Windows system whose IP address is 172.26.26.50, you would access the Cisco TFTP Server View>Options menu and enter the filename path in the TFTP server root directory edit box; for example, D:\pix_images. Then, to copy the file to the PIX Firewall, use the following command.

```
copy tftp://172.26.26.50/pix611.bin flash
```

The TFTP server receives the command and determines the actual file location from its root directory information. The server then downloads the TFTP image to the PIX Firewall.

**Note** Your TFTP server must be open when you enter the **copy tftp** command on the PIX Firewall.

The syntax of the **copy tftp flash** command is as follows:

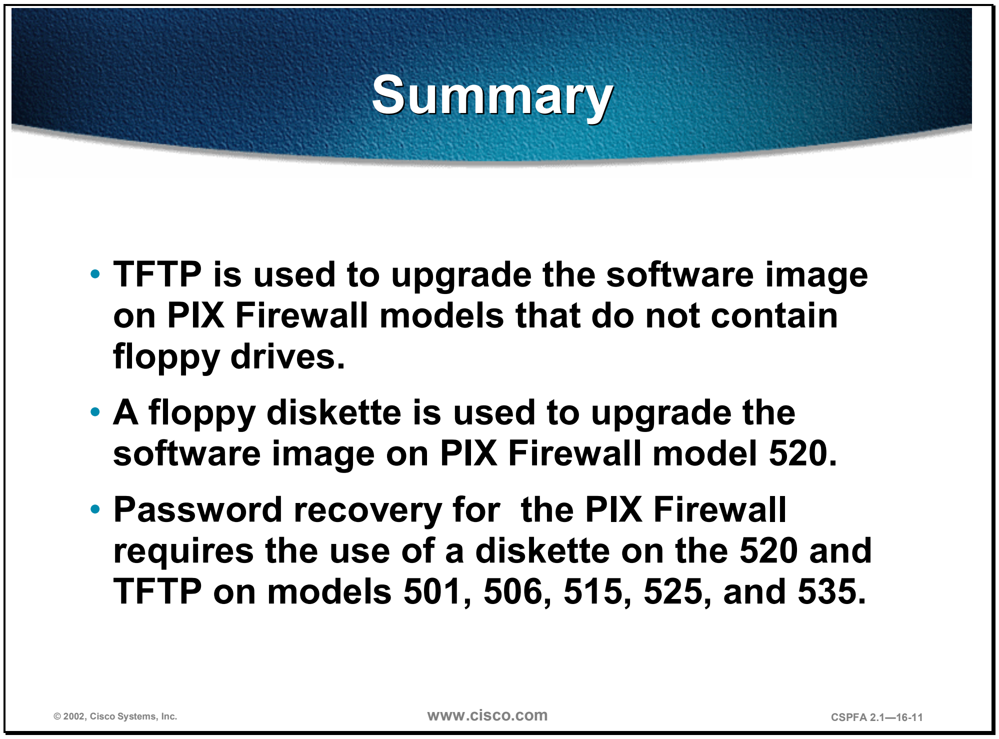
```
copy tftp[:[[//location] [/pathname]]] flash[:[image | pdm]]
```

| | |
|---|---|
| **copy tftp flash** | Enables you to download Flash memory software images via TFTP without using monitor mode. |
| ***location*** | Specifies the IP address or name of the server on which the TFTP server resides. |
| ***pathname*** | Specifies any directory names in the path to the file as well as the actual file name. The |

| | |
|---|---|
| | PIX Firewall must know how to reach this location via its routing table information, which is determined by the **ip address** command, the **route** command, or  RIP, depending upon your configuration. |
| **image** | Allows you to download the selected PIX Firewall image to Flash memory. An image you download is made available to the PIX Firewall on the next reboot. |
| **pdm** | Allows you to download the selected PDM image files to Flash memory. These files are available to the PIX Firewall immediately, without a reboot. |

# Summary

This section summarizes the information you learned and the tasks you completed in this chapter.



## Summary

- TFTP is used to upgrade the software image on PIX Firewall models that do not contain floppy drives.
- A floppy diskette is used to upgrade the software image on PIX Firewall model 520.
- Password recovery for  the PIX Firewall requires the use of a diskette on the 520 and TFTP on models 501, 506, 515, 525, and 535.

www.cisco.com

CSPFA 2.1—16-11

# Lab Exercise—Upgrade the PIX Firewall Image

Complete the following lab exercise to practice what you have learned in this chapter.

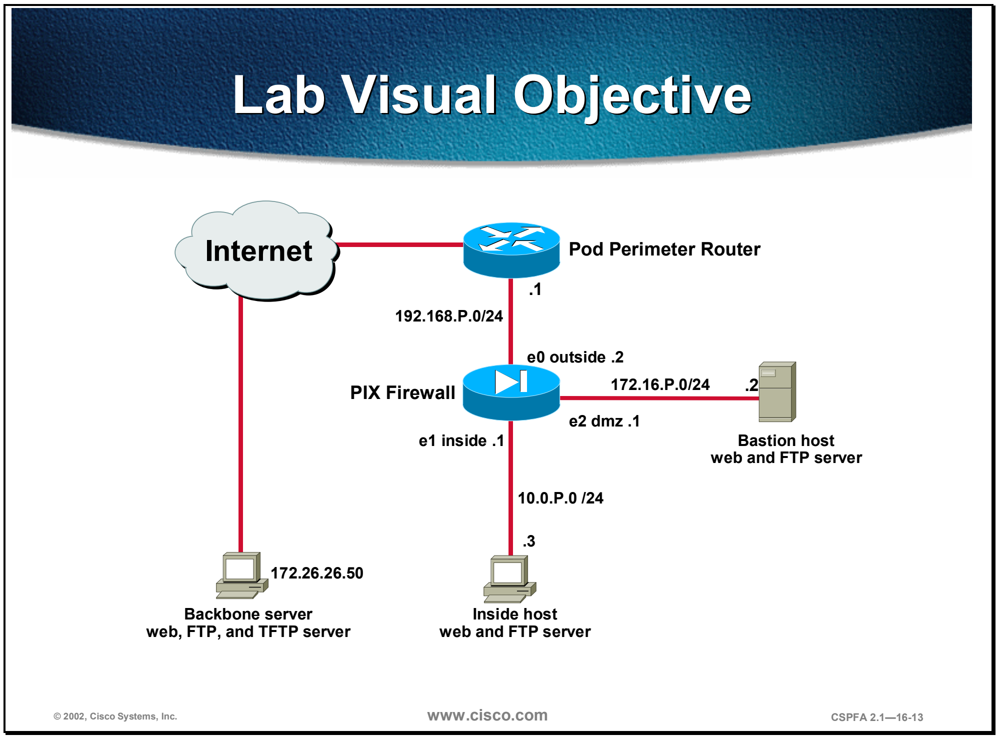
## Objectives

In this lab exercise you will initialize the PIX Firewall by loading the latest software image.

- Perform password recovery for the PIX Firewall 520.
- Perform password recovery for PIX Firewall 515.
- Update the image of the PIX Firewall 515.
- Update the image of the PIX Firewall 520.

## Visual Objective

The following figure displays the lab topology for your classroom environment. You will use the IP addresses in this visual objective for the remainder of the course.



Lab Visual Objective

© 2002, Cisco Systems, Inc. www.cisco.com CSPFA 2.1—16-13

# Setup

Before starting this lab exercise, complete the following:

- Make sure the PIX Firewall is turned on and that your PC is connected to the PIX Firewall.
- Restore the original IP addresses to your PC.
  - Primary IP address: 10.0.P.3

(where P = pod number)

  - Netmask: 255.255.255.0
  - Secondary IP address: 10.1.P.3

(where P = pod number)

  - Netmask: 255.255.255.0

- If you are using PIX Firewall 520 models, make sure that you have a floppy diskette created with the following files:
  - pix5nn.bin
  - rawrite.exe
  - readme.txt
- If you are using PIX Firewall 520 models, insert the floppy diskette with the latest version of the PIX Firewall software into the PIX Firewall.

# Directions

Your task for this exercise is to load either the PIX Firewall 515 or the PIX Firewall 520—depending on which firewall you have set up for your lab exercise—with the most recent version of the PIX Firewall software, and save it to Flash memory. This requires the following tasks.

- Access the PIX Firewall console.
- Enter the enable mode.
- Erase the configuration in Flash memory.
- Load the PIX Firewall image.
- Write the new image to Flash memory.
- Verify the installation of the new image.

## Task 1—Perform Password Recovery for the PIX Firewall 520

To perform a password recovery for the PIX Firewall model 520 and earlier models with 3.5-inch floppy drives, complete the following steps:

**Step 1** Get the correct image file and rawrite.exe file as directed by the instructor.

**Step 2** After retrieving the two files, execute the rawrite.exe file from your Windows host as follows, and enter the information when prompted:

```
C:\> rawrite
RaWrite 1.2 - Write disk file to a floppy diskette
```

```
Enter the source file name: npXXX.bin
Enter the destination drive: a:
Please insert a formatted diskette into drive A: and press -ENTER- : <Enter>
Number of sectors per track for this disk is 18
Writing image to drive A:. Press ^C to abort.
Track: 78 Head: 1 Sector: 16
Done.
C:\>
```

(where XXX = version number)

Reboot your PIX Firewall with the diskette you just created. When prompted, press **y** to erase the password.

```
Do you wish to erase the passwords? [yn] y
Passwords have been erased
```

The system automatically erases the password and starts rebooting.

## Task 2—Perform Password Recovery for PIX Firewall 515

The instructor will provide you with the procedures for access to the PIX Firewall console port. After you access the console port, enter privileged mode. Then, perform a password recovery for the PIX Firewall model 515 by completing the following steps. This procedure requires a TFTP sever.

**Step 1** Clear the translation table:

```
pixP# clear xlate
```

**Step 2** Create an enable password for entering into privileged mode:

```
pixP# enable password badpassword
```

**Step 3** Save your configuration

```
pixP# write memory
```

**Step 4** Reboot your PIX Firewall and interrupt the boot process to enter monitor mode. To do this, press the Escape key or send a break character.

```
pixP# reload
```

**Step 5** Specify the PIX Firewall interface to use for TFTP:

```
monitor> int 0
```

**Step 6** Specify the PIX Firewall interface's IP address:

```
monitor> address 192.168.P.2
```

(where P = pod number)

**Step 7** Enter **gateway** to specify the IP address of the gateway.

```
monitor> gateway 192.168.P.1
```

**Step 8** Verify connectivity to the TFTP server:

```
monitor> ping 172.26.26.50
```

**Step 9** Name the server:

```
monitor> server 172.26.26.50
```

**Step 10** Name the image filename:

```
monitor> file np53.bin
```

**Step 11** Start the TFTP process:

```
monitor> tftp
```

**Step 12** When prompted, press **y** to erase the password:

```
Do you wish to erase the passwords? [yn] y
Passwords have been erased
```

The system automatically erases the password and starts rebooting.

**Step 13** Verify that the password badpassword has been erased by entering privileged mode on your PIX Firewall:

```
pix> en
password: <Enter>
pixP#
```

## Task 3—Update the Image of the PIX Firewall 515

**Note** You will complete this task only if your lab is set up with PIX Firewall 515s.

To load the PIX Firewall 515 image using TFTP, complete the following steps:

**Step 1** After you reload the PIX Firewall 515 and the startup messages appear, press **Esc** to interrupt the boot process. The monitor prompt appears.

```
Use BREAK or ESC to interrupt flash boot.
Flash boot in 5 sec.
Esc
```

**Note** If you are using HyperTerminal with Windows 95, you can press the control and break keys simultaneously to activate a break. Depending on which service pack is installed, Windows NT HyperTerminal may not be able to send a break character. Refer to the Windows NT documentation for more information.

**Step 2** You can enter **?** to list the available commands:

```
monitor> ?
?            this help message
address      [addr]  set ip address
file         [name]  set boot file name
gateway      [addr]  set IP gateway
help         this help message
interface    [num]   select TFTP interface
ping         <addr>  send ICMP echo
reload       halt and reload system
server       [addr]  set server IP address
tftp         TFTP download
timeout      TFTP timeout
trace        toggle packet tracing
```

**Step 3** Specify the PIX Firewall interface to use for TFTP:

```
monitor> int 0
```

**Step 4** Enter **address** to specify the IP address of the PIX Firewall's interface:

```
monitor> address 192.168.P.2
```

(where P = pod number)

**Step 5** Enter **gateway** to specify the IP address of the gateway.

```
monitor> gateway 192.168.P.1
```

(where P = pod number)

**Step 6** Ping the TFTP server to verify connectivity:

```
monitor> ping 172.26.26.50
```

**Step 7** Enter **server** to specify the IP address of the TFTP server:

```
monitor> server 172.26.26.50
```

**Step 8** Enter **file** to specify the filename of the PIX Firewall image (the instructor will provide you with the name of the file):

```
monitor> file pixNNN.bin
```

(where NNN = release number)

**Step 9** Enter **tftp** to start downloading the PIX Firewall 515 image:

```
monitor> tftp
```

**Step 10** After the PIX Firewall has received the image from the TFTP server, it will then reboot itself. During the reboot process, you are prompted for a new activation key. Enter **n** for no.

**Step 11** Enter the **show version** command to verify that you have loaded PIX Firewall software version 6.1.

```
pixP> show version
```

## Task 4—Update the Image of the PIX Firewall 520

**Note** You will complete this task only if your lab is set up with PIX Firewall 520s.

To load the PIX Firewall 520 image, complete the following steps:

**Step 1** At the Windows command prompt, execute the rawrite program. You are prompted for the name of the binary file and the output device, and to insert a formatted diskette into the drive.

```
C:\> rawrite
rawrite 1.2 - write disk file to raw floppy diskette
enter source file name: pixNNN.bin
enter destination drive: a:\
please insert a formatted diskette into drive a: and press -enter-:
writing image to drive a: press "control c" to abort
track: 31 head: 1 sector: 16
done
C:\>
```

(where NNN = revision number)

**Step 2** Remove the diskette from the drive and place it in the PIX Firewall 520 diskette drive. Power cycle the unit, or use the **reload** command from the console to reboot the machine. The PIX Firewall 520 then boots from the new diskette.

**Step 3** For the new configuration to permanently load into memory, you have to save the configuration to system memory. Use the **write memory** command to complete this task.

```
pixfirewall> write memory
```

17

# Cisco PIX Device Manager

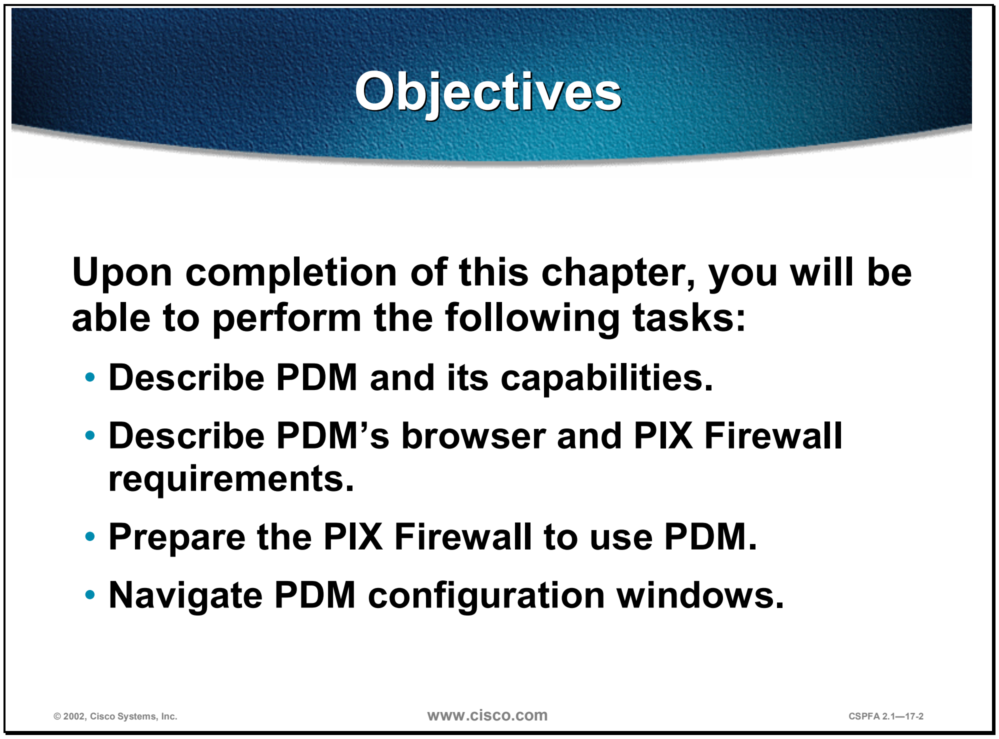
## Overview

This chapter includes the following topics:

- Objectives
- PDM overview
- PDM operating requirements
- Prepare for PDM
- Using PDM
- Other tools
- Summary
- Lab exercise

# Objectives

This section lists the chapter’s objectives.

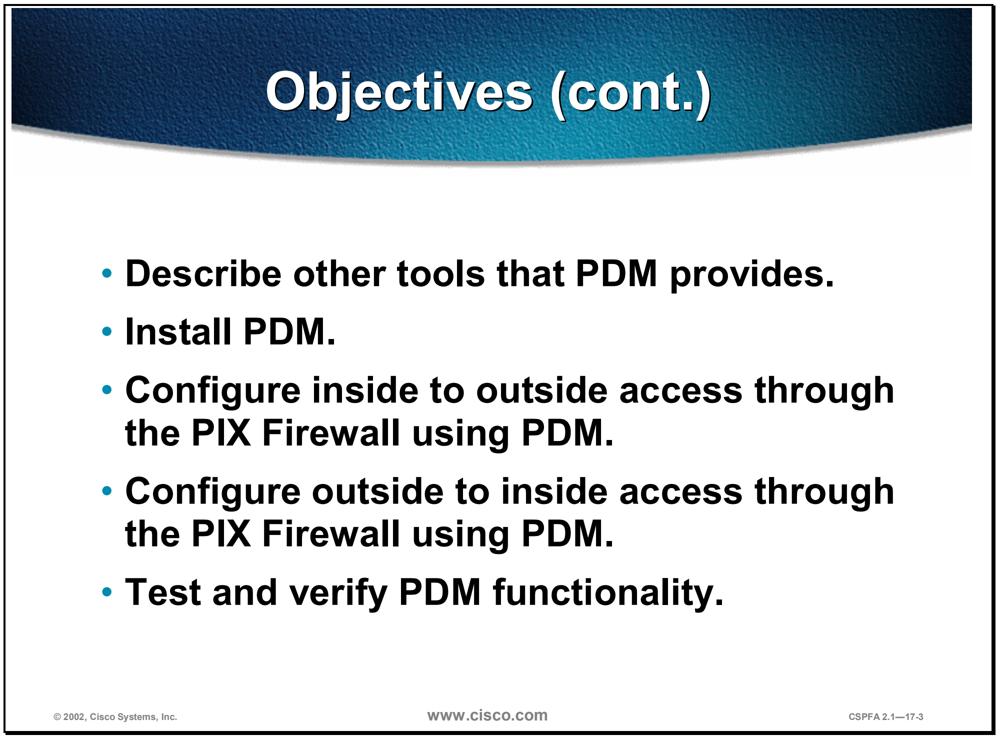


## Objectives

**Upon completion of this chapter, you will be able to perform the following tasks:**

- **Describe PDM and its capabilities.**
- **Describe PDM’s browser and PIX Firewall requirements.**
- **Prepare the PIX Firewall to use PDM.**
- **Navigate PDM configuration windows.**

© 2002, Cisco Systems, Inc. www.cisco.com CSPFA 2.1—17-2



## Objectives (cont.)

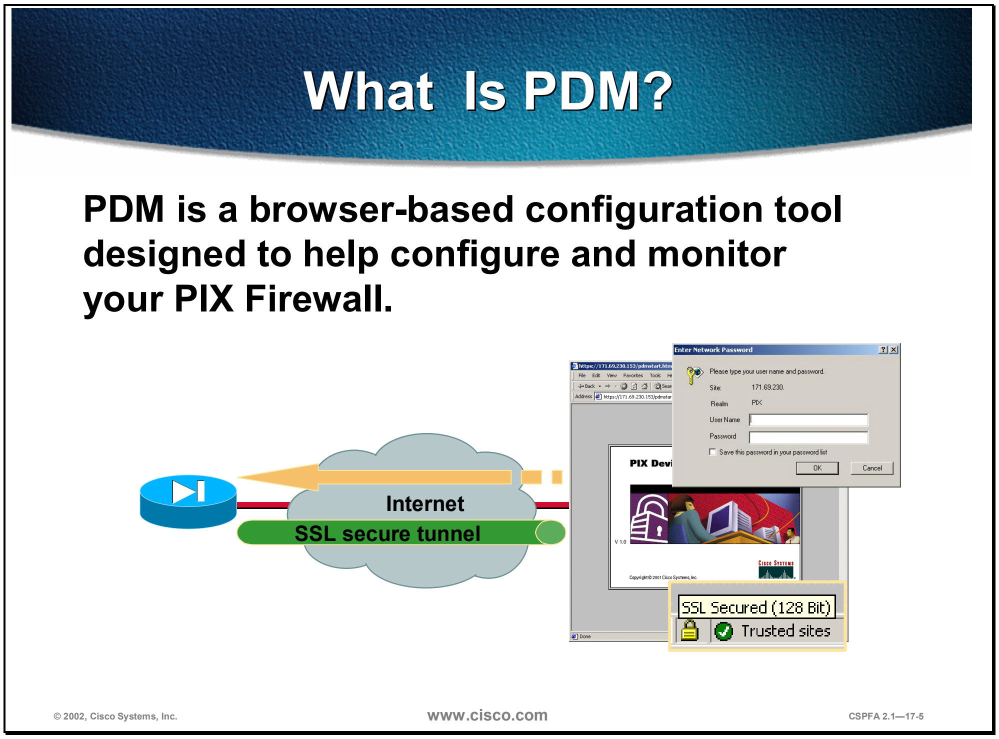
- Describe other tools that PDM provides.
- Install PDM.
- Configure inside to outside access through the PIX Firewall using PDM.
- Configure outside to inside access through the PIX Firewall using PDM.
- Test and verify PDM functionality.

www.cisco.com

CSPFA 2.1—17-3

# PDM Overview

This section provides an overview of the Cisco PIX Device Manager (PDM) and its limitations.



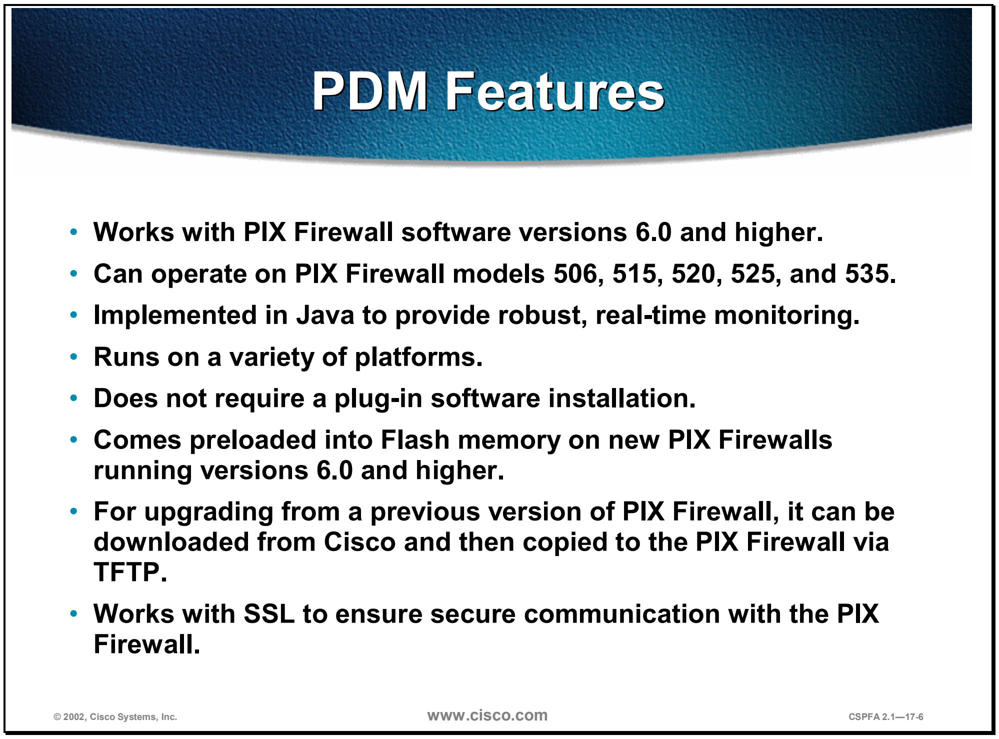
## What Is PDM?

**PDM is a browser-based configuration tool designed to help configure and monitor your PIX Firewall.**

© 2002, Cisco Systems, Inc. www.cisco.com CSPFA 2.1—17-5

PDM is a browser-based configuration tool designed to help you set up, configure, and monitor your Cisco PIX Firewall graphically, without requiring an extensive knowledge of the PIX Firewall command line interface (CLI).

PDM monitors and configures a single PIX Firewall. You can use PDM to create a new configuration, or you can use PDM in addition to a configuration you create or maintain from the PIX Firewall console or with Cisco Secure Policy Manager (CSPM). You can point your browser to more than one PIX Firewall and administer several PIX Firewall from a single workstation.



## PDM Features

- Works with PIX Firewall software versions 6.0 and higher.
- Can operate on PIX Firewall models 506, 515, 520, 525, and 535.
- Implemented in Java to provide robust, real-time monitoring.
- Runs on a variety of platforms.
- Does not require a plug-in software installation.
- Comes preloaded into Flash memory on new PIX Firewalls running versions 6.0 and higher.
- For upgrading from a previous version of PIX Firewall, it can be downloaded from Cisco and then copied to the PIX Firewall via TFTP.
- Works with SSL to ensure secure communication with the PIX Firewall.

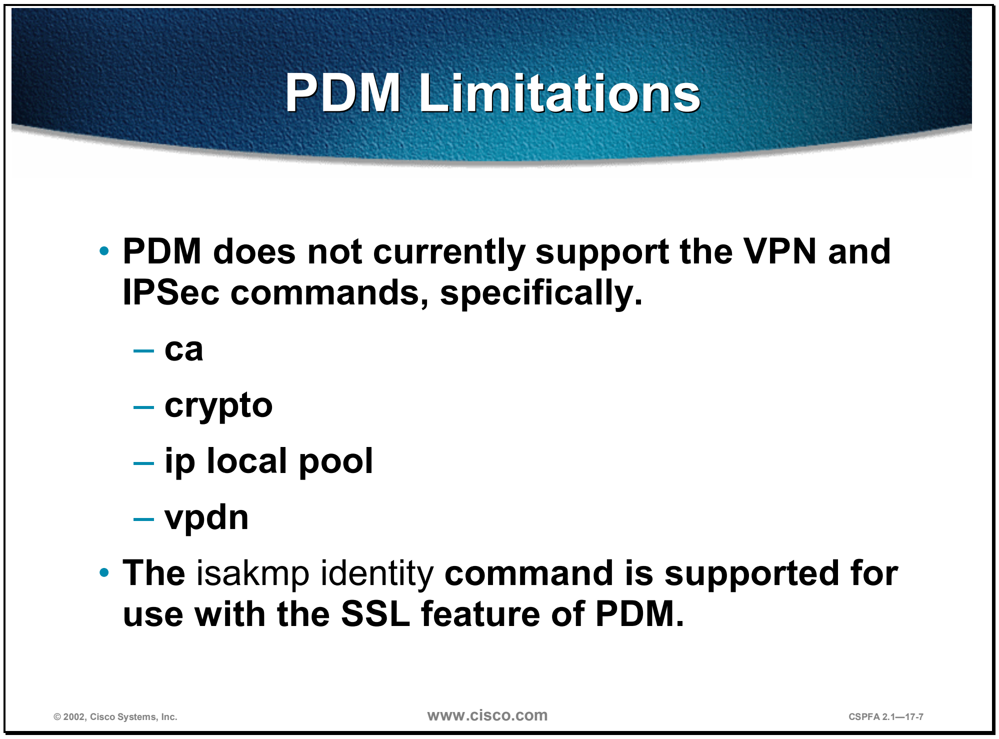
© 2002, Cisco Systems, Inc. www.cisco.com CSPFA 2.1—17-6

PDM is secure, versatile, and easy to use. It works with most PIX Firewall models and runs on a variety of platforms.

PDM enables you to securely configure and monitor your PIX Firewall remotely. Its ability to work with the Secure Socket Layer (SSL) protocol ensures that communication with the PIX Firewall is secure, and because it is implemented in Java, it is able to provide robust, real-time monitoring.

PDM works with PIX Firewall software versions 6.0 and higher and can operate on PIX Firewall models 506, 515, 520, 525, and 535. It comes preloaded into Flash memory on new PIX Firewalls running software versions 6.0 and higher. If you are upgrading from a previous version of PIX Firewall, PDM can be downloaded from Cisco and then copied to the PIX Firewall via TFTP.

PDM runs on Windows, Sun Solaris, and Linux platforms and requires no plug-ins or complex software installations. The PDM applet uploads to your workstation when you access the PIX Firewall from your browser.



## PDM Limitations

- **PDM does not currently support the VPN and IPSec commands, specifically.**
  - **ca**
  - **crypto**
  - **ip local pool**
  - **vpdn**
- **The** isakmp identity **command is supported for use with the SSL feature of PDM.**

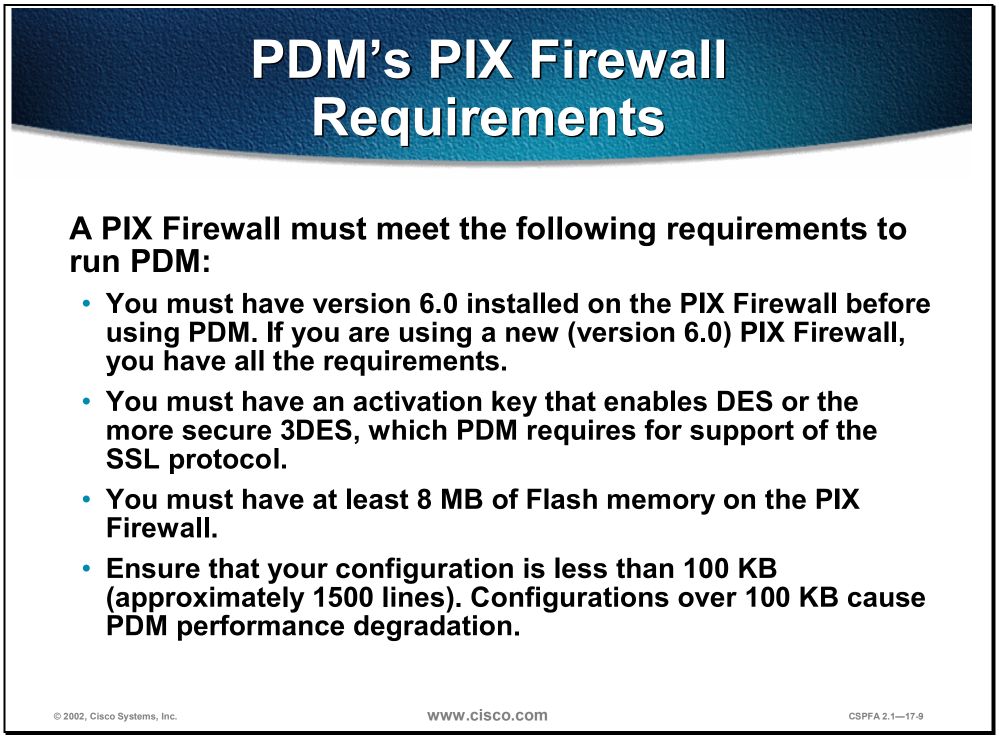
www.cisco.com

CSPFA 2.1—17-7

PDM does not currently support the VPN and IPSec command sets. These are the **ca**, **crypto**, **ip local pool**, and **vpdn** commands. The **isakmp identity** command is supported for use with the SSL feature of PDM.

# PDM Operating Requirements

This section discusses the system requirements for PDM.



## PDM's PIX Firewall Requirements

**A PIX Firewall must meet the following requirements to run PDM:**

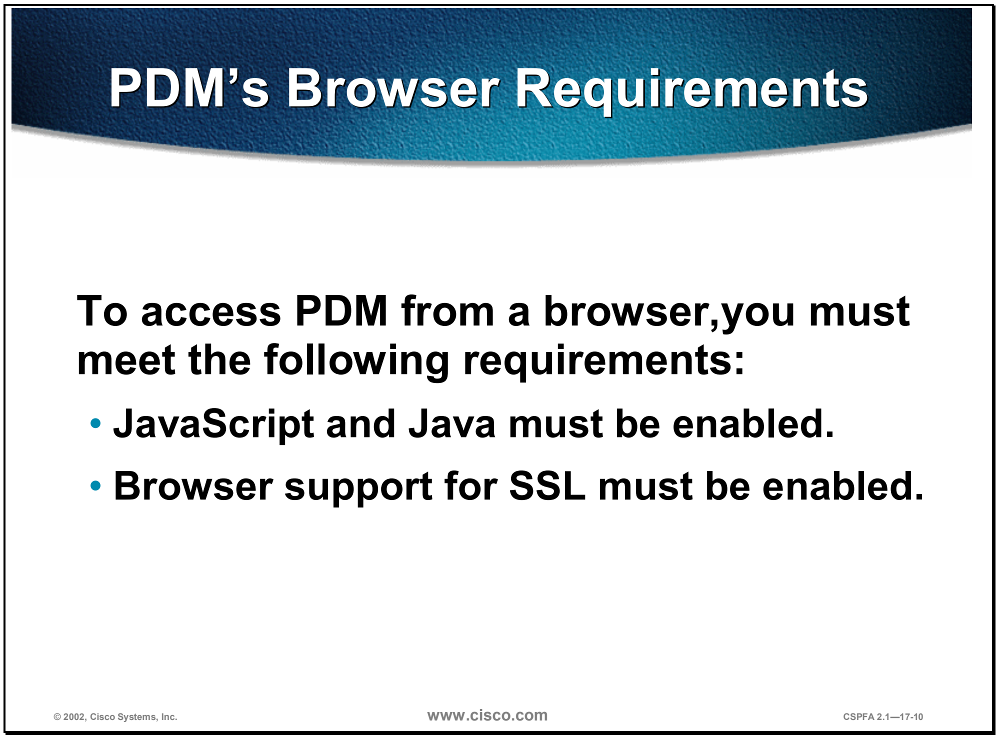
- **You must have version 6.0 installed on the PIX Firewall before using PDM. If you are using a new (version 6.0) PIX Firewall, you have all the requirements.**
- **You must have an activation key that enables DES or the more secure 3DES, which PDM requires for support of the SSL protocol.**
- **You must have at least 8 MB of Flash memory on the PIX Firewall.**
- **Ensure that your configuration is less than 100 KB (approximately 1500 lines). Configurations over 100 KB cause PDM performance degradation.**

© 2002, Cisco Systems, Inc. www.cisco.com CSPFA 2.1—17-9

A PIX Firewall must meet the following requirements to run PDM:

**Note** New PIX Firewalls that contain version 6.0 also have a pre-installed DES activation key. If you are using a new PIX Firewall, you have all the requirements discussed in this section and you can continue to the next section.

- You must have an activation key that enables Data Encryption Standard (DES) or the more secure Triplet-Data Encryption Standard (3DES), which PDM requires for support of the SSL protocol. If your PIX Firewall is not enabled for DES, you can have a new activation key sent to you by completing the form at the following web site: www.cisco.com/kobayashi/sw-center/internet/pix-56bit-license-request.shtml
- Verify that your PIX Firewall meets all version 6.0 requirements listed in the release notes for the PIX Firewall version 6.0. You must have version 6.0 installed on the PIX Firewall before using PDM. You can download version 6.0 and the PDM software from the following web site: www.cisco.com/cgi-bin/tablebuild.pl/pix.
- You must have at least 8 MB of Flash memory on the PIX Firewall.
- Ensure that your configuration is less than 100 KB (approximately 1500 lines). Configurations over 100 KB cause PDM performance degradation.



## PDM's Browser Requirements

**To access PDM from a browser,you must meet the following requirements:**

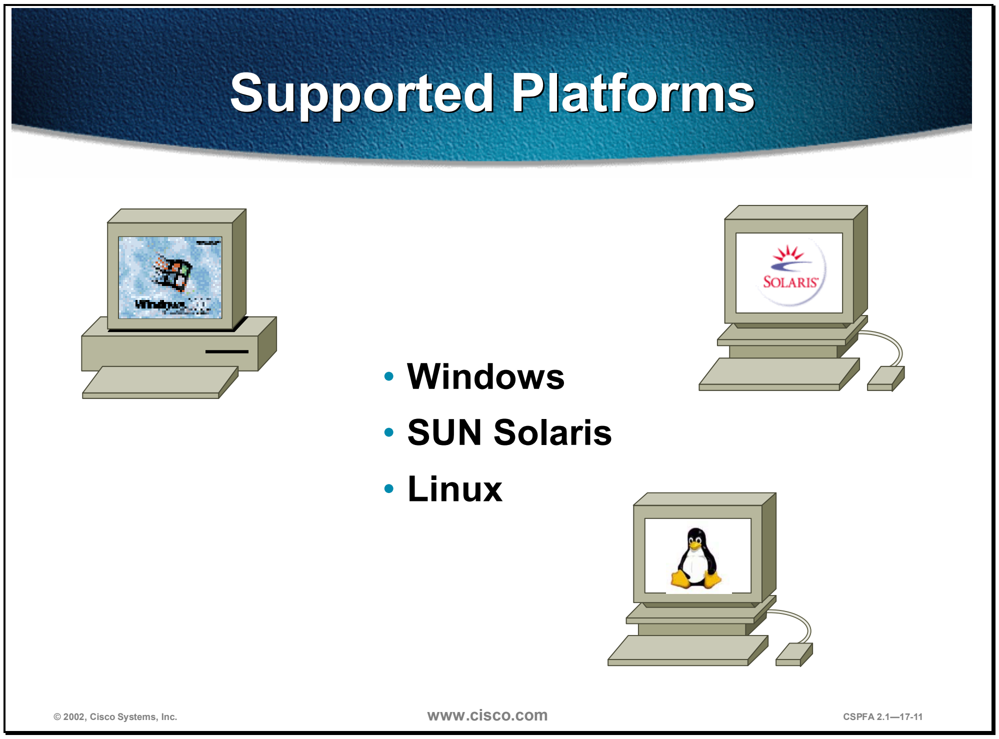
- **JavaScript and Java must be enabled.**
- **Browser support for SSL must be enabled.**

© 2002, Cisco Systems, Inc. www.cisco.com CSPFA 2.1—17-10

To access PDM from a browser, you must meet the following requirements:

- JavaScript and Java must be enabled. If these are not enabled, PDM helps you enable them. If you are using Microsoft Internet Explorer, your Java Development Kit (JDK) version should be 1.1.4 or higher. To check which version you have, launch PDM. When the PDM information window opens, the field JDK Version indicates your JDK version. If you have an older JDK version, you can get the latest Java Virtual Machine (JVM) from Microsoft by downloading the product called Virtual Machine.
- Browser support for SSL must be enabled. The supported versions of Internet Explorer and Netscape Navigator support SSL without requiring additional configuration.

**Note** PIX Firewall version 6.0 supports SSL 2.0, SSL 3.0, and TLS 1.0 in the browser. PIX Firewall supports all browser encryption levels.



## Supported Platforms

- Windows
- SUN Solaris
- Linux

© 2002, Cisco Systems, Inc. www.cisco.com CSPFA 2.1—17-11

PDM can operate in browsers running on Windows, SUN, Solaris, or Linux operating systems. The requirements for each operating system follow:

## Windows Requirements

The following requirements apply to the use of PDM with Windows:

- Windows 2000 (Service Pack 1), Windows NT 4.0 (Service Pack 4 and higher), Windows 98, or Windows ME
- The supported browsers are Internet Explorer 5.0 (Service Pack 1) or higher (version 5.5 is recommended), and Netscape Communicator 4.51 or higher (version 4.76 is recommended)
- Any Pentium or Pentium-compatible processor running at 350 MHz or higher
- At least 128 MB of RAM (192 MB or more is recommended)
- An 800 x 600 pixel display with at least 256 colors (a 1024 x 768 pixel display and at least 16-bit colors is recommended)
- PDM does not support use on Windows 3.1 or Windows 95

**Note** The use of virus checking software may dramatically increase the time required to start PDM. This is especially true for Netscape Communicator on any Windows platform or Windows 2000 running any browser.

## SUN Solaris Requirements

The following requirements apply to the use of PDM with Sun SPARC:

- Sun Solaris 2.6 or later running CDE or OpenWindows window manager
- SPARC microprocessor
- The supported browser is Netscape Communicator 4.51 or higher (version 4.76 is recommended)
- At least 128 MB of RAM
- An 800 x 600 pixel display with at least 256 colors (a 1024 x 768 pixel display and at least 16-bit colors is recommended)

**Note** PDM does not support Solaris on IBM PCs.

## Linux Requirements

The following requirements apply to the use of PDM with Linux:

- Red Hat Linux 7.0 running the GNOME or KDE 2.0 desktop environment
- The supported browser is Netscape Communicator 4.75 or later
- At least 64 MB of RAM
- An 800 x 600 pixel display with at least 256 colors (a 1024 x 768 pixel display and at least 16-bit colors is recommended)

## General Guidelines

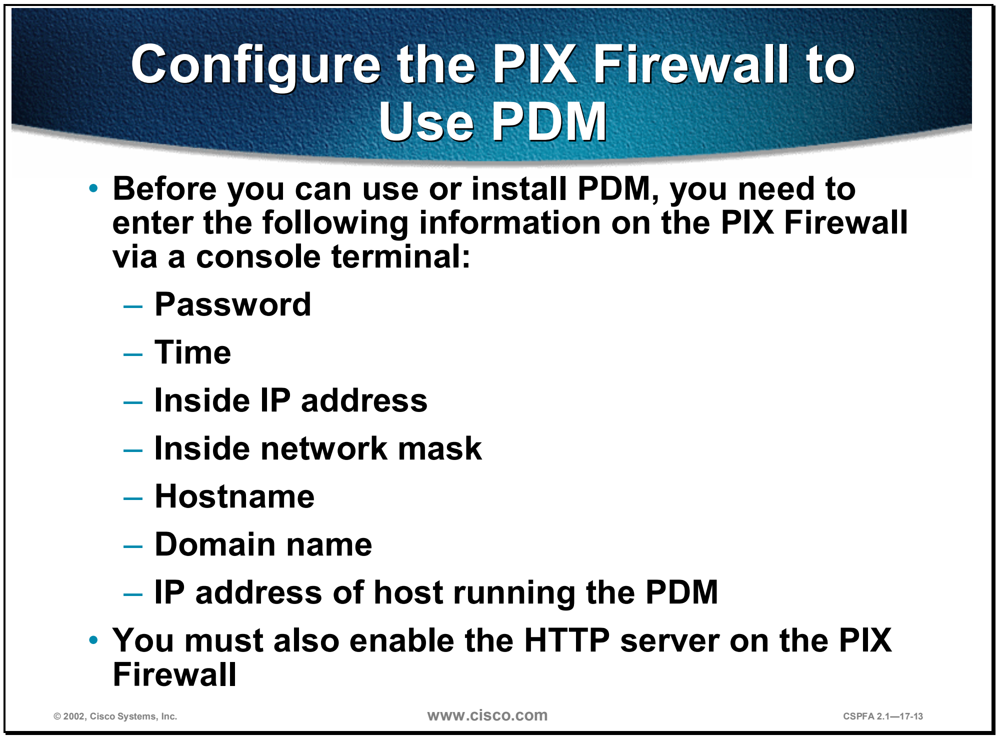
The following are a few general guidelines for workstations running PDM:

- You can run several PDM sessions on a single workstation. The maximum number of PDM sessions you can run varies depending on your workstation's resources such as memory, CPU speed, and browser type.
- The time required to download the PDM applet can be greatly affected by the speed of the link between your workstation and the PIX Firewall. A minimum of 56 Kbps link speed is required; however, 1.5 Mbps or higher is recommended. Once the PDM applet is loaded on your workstation, the link speed impact on PDM operation is negligible.
- The use of virus checking software may dramatically increase the time required to start PDM. This is especially true for Netscape Communicator on any Windows platform or Windows 2000 running any browser.

If your workstation's resources are running low, you should close and re-open your browser before launching PDM.

# Prepare for PDM

This section details the configuration of the PIX Firewall to enable the use of PDM.



## Configure the PIX Firewall to Use PDM

- Before you can use or install PDM, you need to enter the following information on the PIX Firewall via a console terminal:
  - Password
  - Time
  - Inside IP address
  - Inside network mask
  - Hostname
  - Domain name
  - IP address of host running the PDM
- You must also enable the HTTP server on the PIX Firewall

© 2002, Cisco Systems, Inc. www.cisco.com CSPFA 2.1—17-13

The PIX Firewall must be configured with the following information before you can install or use PDM. You can either pre-configure a new PIX Firewall through the interactive prompts, which appear after the PIX Firewall boots, or you can enter the commands shown below each information item.

- Enable Password—Enter an alphanumeric password to protect the PIX Firewall's privileged mode. The alphanumeric password can be up to 16 characters in length. You must use this password to log in to PDM. The command syntax for enabling a password is as follows:

  ```
  enable password password [encrypted]
  ```

- Time—Set the PIX Firewall clock to Universal Coordinated Time (UTC, also known as Greenwich Mean Time, or GMT). For example, if you are in the Pacific Daylight Savings time zone, set the clock 7 hours ahead of your local time to set the clock to UTC. Enter the year, month, day, and time. Enter the UTC time in 24-hour time as hour:minutes:seconds. The command syntax for setting the clock is as follows:

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

- Inside IP address—Specify the IP address of the PIX Firewall's inside interface. Ensure that this IP address is unique on the network and not used by any other computer or network device, such as a router. The command syntax for setting an inside IP address is as follows:

  ```
  ip address if_name ip_address [netmask]
  ```

- Inside network mask—Specify the network mask for the inside interface. An example mask is 255.255.255.0. You can also specify a subnetted mask, for example: 255.255.255.224. Do not use all 255s, such as 255.255.255.255; this prevents traffic from passing on the interface. Use the **ip address** command shown above to set the inside network mask.
- Host name—Specify up to 16 characters as a name for the PIX Firewall unit. The command syntax for setting a host name is as follows:

```
hostname newname
```

- Domain name—Specify the domain name for the PIX Firewall. The command syntax for enabling domain name is as follows:

```
domain-name name
```

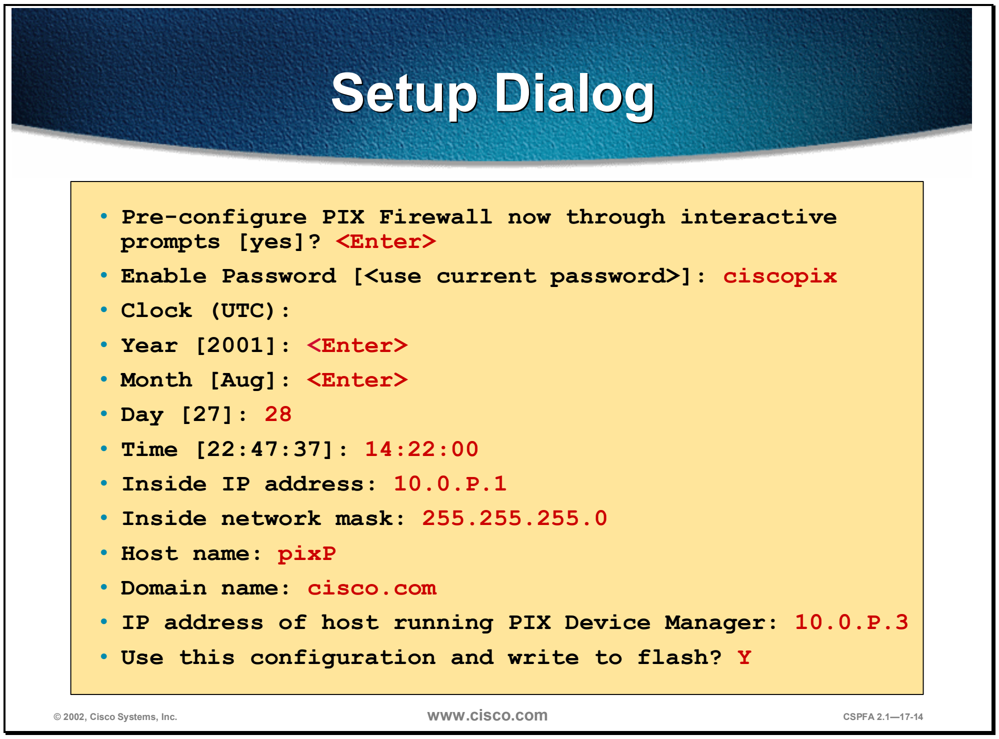
- IP address of the host running PDM—Specify the IP address of the workstation that will access PDM from its browser. The command syntax for granting permission for a host to connect to the PIX Firewall with SSL is as follows:

```
http ip_address [netmask] [if_name]
```

- HTTP Server—Enable the HTTP server on the PIX Firewall with the **http server enable** command. You must also use the **http ip_address** command to specify the host or network authorized to initiate an HTTP connection to the PIX Firewall.

If you are installing PDM on a PIX Firewall with an existing configuration, you may need to restructure your configuration from the PIX Firewall CLI before installing PDM in order to obtain full PDM capability. There are certain commands that PDM does not support in a configuration. If these commands are present in your configuration, you will only have access to the Monitoring tab. This is because PDM handles each PIX Firewall command in one of the following ways, each of which is explained in detail in the document titled “PDM Support for PIX Firewall CLI Commands” on CCO:

- Parse and allow changes (supported commands)
- Parse and only permit access to the Monitoring tab (unsupported commands)
- Parse without allowing changes (commands PDM does not understand but handles without preventing further configuration)
- Only display in the unparseable command list (commands PDM does not understand but handles without preventing further configuration)



## Setup Dialog

- Pre-configure PIX Firewall now through interactive prompts [yes]? <Enter>
- Enable Password [<use current password>]: ciscopix
- Clock (UTC):
- Year [2001]: <Enter>
- Month [Aug]: <Enter>
- Day [27]: 28
- Time [22:47:37]: 14:22:00
- Inside IP address: 10.0.P.1
- Inside network mask: 255.255.255.0
- Host name: pixP
- Domain name: cisco.com
- IP address of host running PIX Device Manager: 10.0.P.3
- Use this configuration and write to flash? Y

© 2002, Cisco Systems, Inc. www.cisco.com CSPFA 2.1—17-14

An unconfigured PIX Firewall will start up in an interactive setup dialog to enable you to perform the initial configuration required to use PDM. You can also access the setup dialog by entering **setup** at the configuration mode prompt.

The dialog asks for several responses, including the inside IP address, network mask, host name, domain name and PDM host. The host and domain names are used to generate the default certificate for the SSL connection.

The example in the figure shows how to respond to the **setup** command prompts. Pressing the Enter key instead of entering a value at the prompt accepts the default value within the brackets. You must fill in any fields that show no default values, and change default values as necessary. After the configuration is written to Flash memory, your PIX Firewall is ready to start PDM.

***CAUTION*** *The clock must be set for PDM to generate a valid certification. Set the PIX Firewall clock to Universal Coordinated Time (also known as Greenwich Mean Time).*

The following table explains each prompt in the setup dialog.

| Setup Dialog Prompts | Description |
|---|---|
| **Enable password** | Enables you to specify an enable password for this PIX Firewall. |
| **Clock (UTC)** | Enables you to set the PIX Firewall clock to Universal Coordinated Time (also known as Greenwich Mean Time). |
| **Year [system year]:** | Enables you to specify the current year, or return to the default year stored in the host computer. |

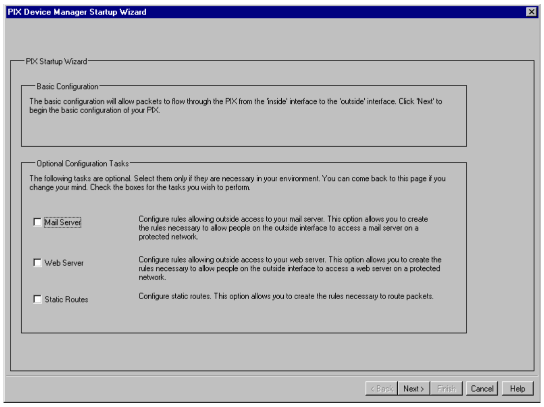
| Setup Dialog Prompts | Description |
|---|---|
| **Month [system month]:** | Enables you to specify the current month, or return to the default month stored in the host computer. |
| **Day [system day]:** | Enables you to specify the current day, or return to the default day stored in the host computer. |
| **Time [system time]:** | Enables you to specify the current time in hh:mm:ss format, or return to the default time stored in the host computer. |
| **Inside IP address:** | The network interface IP address of the PIX Firewall. |
| **Inside network mask:** | A network mask that applies to the inside IP address. Use 0.0.0.0 to specify a default route. The 0.0.0.0 netmask can be abbreviated as 0. |
| **Host name:** | The host name you want to display in the PIX Firewall command line prompt. |
| **Domain name:** | The DNS domain name of the network on which the PIX Firewall runs (for example, cisco.com). |
| **IP address of host running PIX Device Manager:** | IP address on which PDM connects to the PIX Firewall. |
| **Use this configuration and write to flash?** | Enables you to store the new configuration to Flash memory. It is the same as the **write memory** command. If the answer is yes, the inside interface is enabled and the requested configuration is written to Flash memory. If the user answers anything else, the setup dialog repeats using the values already entered as the defaults for the questions. |

# Using PDM

This section discusses getting started with PDM and the layout of the product.

## Startup Wizard

**The PDM Startup Wizard enables you to easily perform basic configuration of the PIX Firewall.**



PIX Startup Wizard

Basic Configuration

The basic configuration will allow packets to flow through the PIX from the 'inside' interface to the 'outside' interface. Click 'Next' to begin the basic configuration of your PIX.

Optional Configuration Tasks

The following tasks are optional. Select them only if they are necessary in your environment. You can come back to this page if you change your mind. Check the boxes for the tasks you wish to perform.

| | |
|---|---|
| □ Mail Server | Configure rules allowing outside access to your mail server. This option allows you to create the rules necessary to allow people on the outside interface to access a mail server on a protected network. |
| □ Web Server | Configure rules allowing outside access to your web server. This option allows you to create the rules necessary to allow people on the outside interface to access a web server on a protected network. |
| □ Static Routes | Configure static routes. This option allows you to create the rules necessary to route packets. |

www.cisco.com CSPFA 2.1—17-16

The PDM Startup Wizard is an easy way to begin the process of configuring your PIX Firewall. The wizard steps you through such tasks as the following:

- Enabling the PIX Firewall interfaces
- Assigning IP addresses to the interfaces
- Configuring static routes
- Configuring NAT
- Assigning a global pool of addresses to be used for NAT

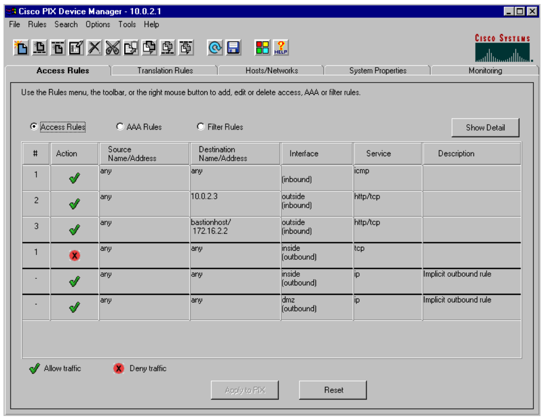
The Startup Wizard runs only if the PIX Firewall has not previously been configured. The user cannot choose to run the wizard; PDM determines whether it runs. PDM assumes that the PIX Firewall has not been configured if both of the following are true:

- None of the PIX Firewall interfaces, except the inside interface, has an IP address configured
- None of the following commands exist in the configuration: **access-list, conduit**, **global**, **nat**, **outbound**, and **static**

# Overall Layout

**PDM consists of five major configuration areas:**

- Access rules
- Translation rules
- Hosts or networks
- System properties
- Monitoring


Cisco PIX Device Manager - 10.0.2.1
File Rules Search Options Tools Help
Cisco Systems
Access Rules | Translation Rules | Hosts/Networks | System Properties | Monitoring

Use the Rules menu, the toolbar, or the right mouse button to add, edit or delete access, AAA or filter rules.

● Access Rules ○ AAA Rules ○ Filter Rules [Show Detail]

| # | Action | Source Name/Address | Destination Name/Address | Interface | Service | Description |
|---|---|---|---|---|---|---|
| 1 | ✓ | any | any | (inbound) | icmp | |
| 2 | ✓ | any | 10.0.2.3 | outside<br>(inbound) | http/tcp | |
| 3 | ✓ | any | bastionhost/<br>172.16.2.2 | outside<br>(inbound) | http/tcp | |
| 1 | ☒ | any | any | inside<br>(outbound) | tcp | |
| - | ✓ | any | any | inside<br>(outbound) | ip | Implicit outbound rule |
| - | ✓ | any | any | dmz<br>(outbound) | ip | Implicit outbound rule |

✓ Allow traffic ☒ Deny traffic

www.cisco.com

CSPFA 2.1—17-17

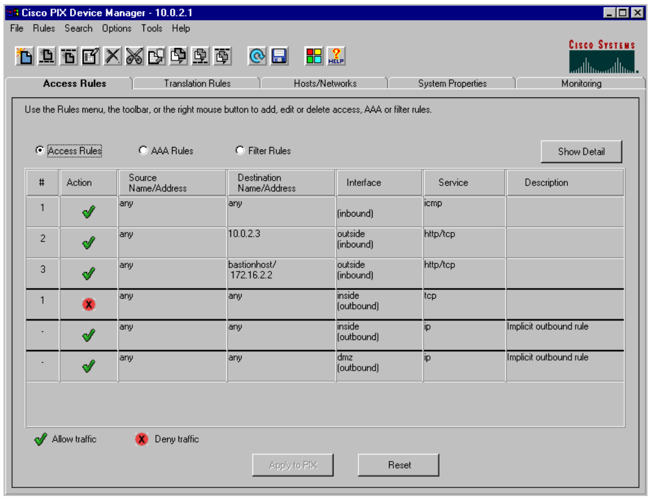
PDM consists of five tabs, which enable you to configure various aspects of the product:

- Access Rules—Shows your entire network security policy
- Translation Rules—Enables you to view all the address translation rules applied to your network
- Hosts/Networks—Enables you to view, edit, add to, or delete from the list of hosts and networks defined for the selected interface
- System Properties—Enables you to configure many aspects of the PIX Firewall
- Monitoring—Enables you to access the various monitoring features of PDM

The remainder of the section covers each configuration area in detail.

# Access Rules

**The Access Rules tab shows your entire network security policy.**



Use the Rules menu, the toolbar, or the right mouse button to add, edit or delete access, AAA or filter rules.

● Access Rules ○ AAA Rules ○ Filter Rules

| # | Action | Source Name/Address | Destination Name/Address | Interface | Service | Description |
|---|---|---|---|---|---|---|
| 1 | ✓ | any | any | (inbound) | icmp | |
| 2 | ✓ | any | 10.0.2.3 | outside (inbound) | http/tcp | |
| 3 | ✓ | any | bastionhost/ 172.16.2.2 | outside (inbound) | http/tcp | |
| 1 | ☒ | any | any | inside (outbound) | tcp | |
| - | ✓ | any | any | inside (outbound) | ip | Implicit outbound rule |
| - | ✓ | any | any | dmz (outbound) | ip | Implicit outbound rule |

✓ Allow traffic ☒ Deny traffic

www.cisco.com CSPFA 2.1—17-18

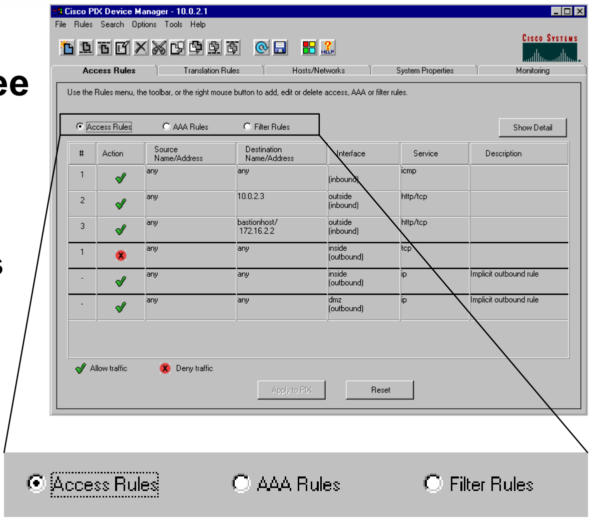
The Access Rules tab combines the concepts of access control lists (ACLs), outbound lists, and conduits to describe how an entire subnet or specific network host, interacts with another to permit or deny a specific service, protocol, or both. PDM does not support the use of ACLs, conduits, and outbounds together. Only one of the three can be used, ACLs being the preferred choice. The choice you make continues to be used by PDM. If you attempt to use more than one of these choices in your configuration, you will only be able to do monitoring.

This tab also enables you to define authentication, authorization, or accounting rules, and filter rules for ActiveX and Java. The configuration edits you perform on the Access Rules tab are captured by PDM but are not sent to the PIX Firewall until you click **Apply to PIX**. This applies to all configuration performed with PDM, including those performed in the Translation Rules tab, the Hosts/Networks tab, and the System Properties tab. Always click **Apply to PIX** to send your configuration edits to the PIX Firewall. Also remember, it is very important to save your configuration to Flash memory by choosing **File>Write Configuration to Flash** from the main menu or clicking the **Save** icon in the toolbar.

## Access Rule Types

There are three rule types on the Access Rules tab:

- Access Rules
- AAA Rules
- Filter Rules



Use the Rules menu, the toolbar, or the right mouse button to add, edit or delete access, AAA or filter rules.

● Access Rules ○ AAA Rules ○ Filter Rules

| # | Action | Source Name/Address | Destination Name/Address | Interface | Service | Description |
|---|---|---|---|---|---|---|
| 1 | ✓ | any | any | (inbound) | icmp | |
| 2 | ✓ | any | 10.0.2.3 | outside (inbound) | http/tcp | |
| 3 | ✓ | any | bastionhost/ 172.16.2.2 | outside (inbound) | http/tcp | |
| 1 | ✗ | any | any | inside (outbound) | tcp | |
| - | ✓ | any | any | inside (outbound) | ip | Implicit outbound rule |
| - | ✓ | any | any | dmz (outbound) | ip | Implicit outbound rule |

✓ Allow traffic ✗ Deny traffic

● Access Rules ○ AAA Rules ○ Filter Rules

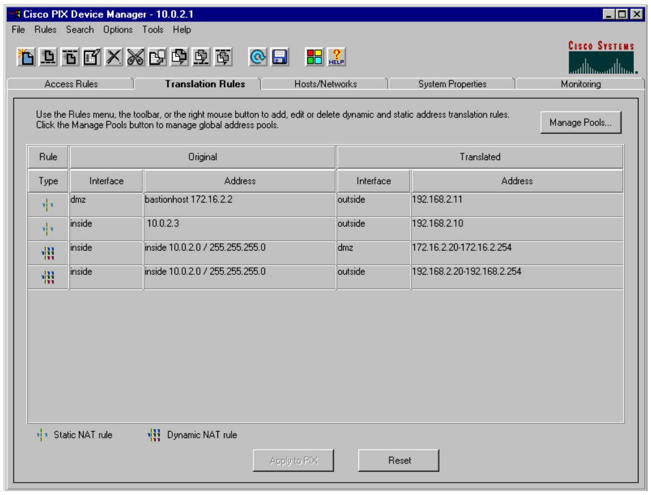
www.cisco.com CSPFA 2.1—17-19

The Access Rules tab has three types of rules:

- Access Rules—Govern which hosts can communicate with other hosts using protocols and services
- AAA Rules—Govern which connections between which hosts will be subjected to authentication, authorization, or accounting
- Filter Rules—Govern which connections between which hosts will be subjected to content or URL filtering

# Translation Rules

**The Translation Rules tab enables you to view all the address translation rules applied to your network.**



Cisco PIX Device Manager - 10.0.2.1

Use the Rules menu, the toolbar, or the right mouse button to add, edit or delete dynamic and static address translation rules.
Click the Manage Pools button to manage global address pools.

| Rule | Original | | Translated | |
|---|---|---|---|---|
| Type | Interface | Address | Interface | Address |
| ✛ | dmz | bastionhost 172.16.2.2 | outside | 192.168.2.11 |
| ✛ | inside | 10.0.2.3 | outside | 192.168.2.10 |
| ▦ | inside | inside 10.0.2.0 / 255.255.255.0 | dmz | 172.16.2.20-172.16.2.254 |
| ▦ | inside | inside 10.0.2.0 / 255.255.255.0 | outside | 192.168.2.20-192.168.2.254 |

✛ Static NAT rule ▦ Dynamic NAT rule

www.cisco.com CSPFA 2.1—17-20

The Translations Rules tab enables you to create and view static and dynamic address translation rules for your network. Before you can designate access and translation rules for your network, you must first define each host or server for which a rule will apply. To do this, click the **Hosts/Networks** tab to define hosts and networks.

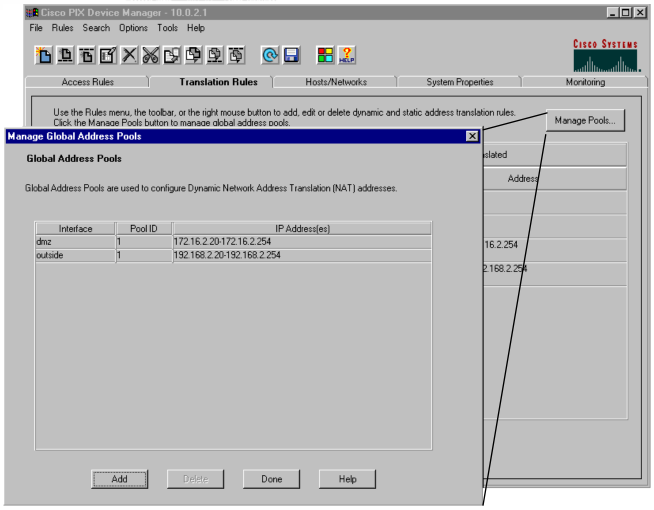
When you are working in either the Access Rules or the Translation Rules tabs, you can access the task menus used for modifying rules three ways:

- The PDM toolbar
- The Rules menu
- Right-clicking anywhere in the rules table

**Note** The order in which you apply translation rules can affect the way the rules operate. PDM lists the static translations first and then the dynamic translations. When processing NAT, the PIX Firewall first translates the static translations in the order they are configured. You can use the Insert Before or Insert After command from the Rules menu to determine the order in which static translations are processed. Because dynamically translated rules are processed on a best-match basis, the option to insert a rule before or after a dynamic translation is disabled.

## Manage Global Address Pools

**In the Manage Global Address Pools window, you can view, define new, or delete existing global address pools used in dynamic NAT rules.**



Use the Rules menu, the toolbar, or the right mouse button to add, edit or delete dynamic and static address translation rules.
Click the Manage Pools button to manage global address pools.

**Global Address Pools**

Global Address Pools are used to configure Dynamic Network Address Translation (NAT) addresses.

| Interface | Pool ID | IP Address(es) |
|---|---|---|
| dmz | 1 | 172.16.2.20-172.16.2.254 |
| outside | 1 | 192.168.2.20-192.168.2.254 |

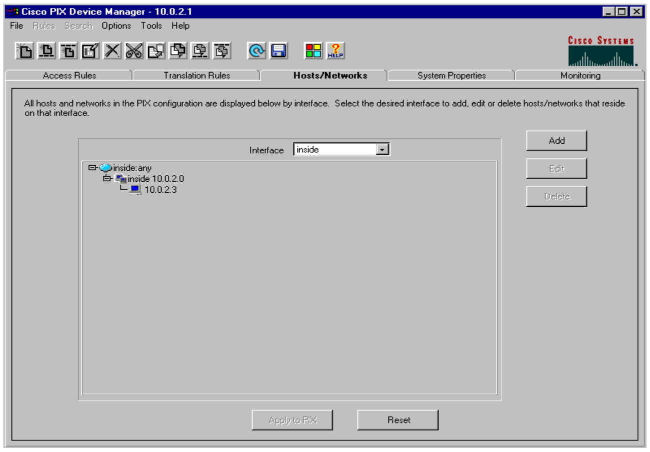
www.cisco.com CSPFA 2.1—17-21

The Manage Global Address Pools window enables you to create global address pools to be used by NAT. From this window, you can also view or delete existing global pools. You can access the Manage Global Address Pools window from the Manage Pools button on the Translation Rules tab.

Remember that it is necessary to run NAT even if your have routable IP addresses on your secure networks. This is a unique feature of the PIX Firewall. You can do this by translating the IP address to itself on the outside.

# Hosts and Networks

On the Hosts/Networks tab you can view, edit, add to, or delete from the list of hosts and networks defined for the selected interface.



All hosts and networks in the PIX configuration are displayed below by interface. Select the desired interface to add, edit or delete hosts/networks that reside on that interface.

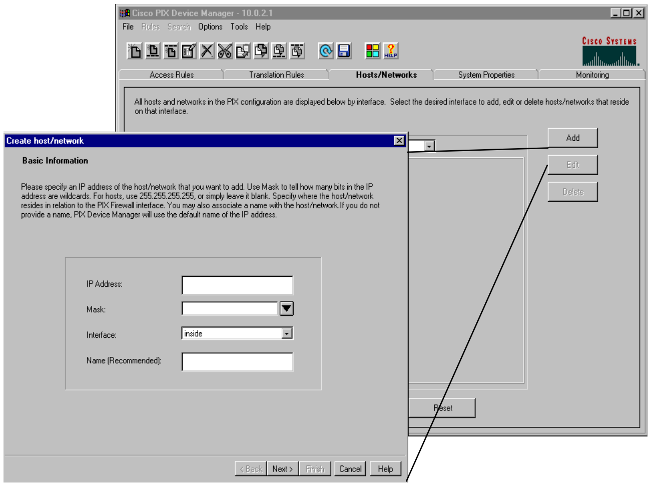
© 2002, Cisco Systems, Inc. www.cisco.com CSPFA 2.1—17-22

The PDM requires that you define any host or network that you intend to use in ACLs and translation rules. These hosts or networks are organized below the interface from which they are reachable. When defining either type of rule, you can reference a host or network by clicking the **Browse** button in the appropriate add or edit rule window. Additionally, you can reference the host or network by name if a name is defined for that host or network. It is recommended that you name all hosts and networks.

In addition to defining the basic information for these hosts or networks, you can define route settings and translation rules (NAT) for any host or network. You can also configure route settings in the Static Route panel on the System Properties tab and translation rules on the Translation Rules tab. These different configuration options accomplish the same results. The Hosts/Networks tab provides another view to modify these settings on a per host and per network basis.

## Create Host and Networks

**Within basic information of the Create host/network window, you specify values for the IP address, netmask, interface, and name of a host or network.**



All hosts and networks in the PIX configuration are displayed below by interface. Select the desired interface to add, edit or delete hosts/networks that reside on that interface.

Basic Information

Please specify an IP address of the host/network that you want to add. Use Mask to tell how many bits in the IP address are wildcards. For hosts, use 255.255.255.255, or simply leave it blank. Specify where the host/network resides in relation to the PIX Firewall interface. You may also associate a name with the host/network.If you do not provide a name, PIX Device Manager will use the default name of the IP address.

IP Address:

Mask:

Interface: inside

Name (Recommended):

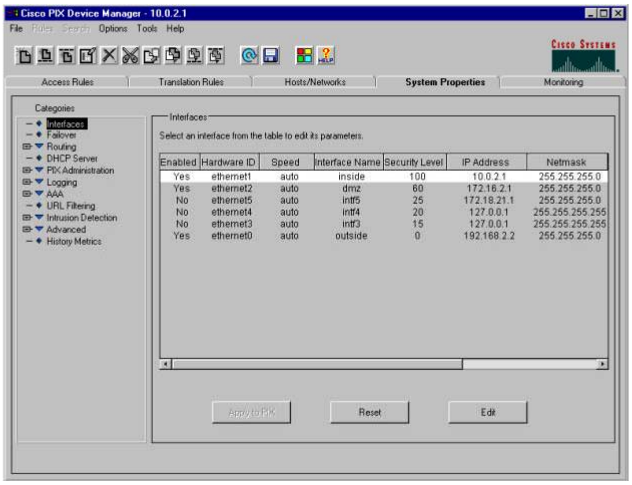
© 2002, Cisco Systems, Inc. www.cisco.com CSPFA 2.1—17-23

The information provided in this window enables the basic identification information for that host or network. This includes values for the IP address, netmask, interface, and name of the host or network. PDM uses the name and IP address and netmask pair to resolve references to this host or network in the source and destination conditions of access rules and in translation rules. PDM uses the interface value to apply access and translation rules that reference this host or network to the correct interface. The interface delivers network packets to the host or network; therefore, it enforces the rules that reference that host or network.

# System Properties

On the System Properties tab, you can configure the following:

- Interfaces
- Failover
- Routing
- DHCP servers
- PIX Firewall administration
- Logging
- AAA
- URL filtering
- Intrusion detection



Select an interface from the table to edit its parameters.

| Enabled | Hardware ID | Speed | Interface Name | Security Level | IP Address | Netmask |
|---|---|---|---|---|---|---|
| Yes | ethernet1 | auto | inside | 100 | 10.0.2.1 | 255.255.255.0 |
| Yes | ethernet2 | auto | dmz | 60 | 172.16.2.1 | 255.255.255.0 |
| No | ethernet5 | auto | intf5 | 25 | 172.18.21.1 | 255.255.255.0 |
| No | ethernet4 | auto | intf4 | 20 | 127.0.0.1 | 255.255.255.255 |
| No | ethernet3 | auto | intf3 | 15 | 127.0.0.1 | 255.255.255.255 |
| Yes | ethernet0 | auto | outside | 0 | 192.168.2.2 | 255.255.255.0 |

www.cisco.com
CSPFA 2.1—17-24

The System Properties tab enables you to configure many aspects of the PIX Firewall, including the following:

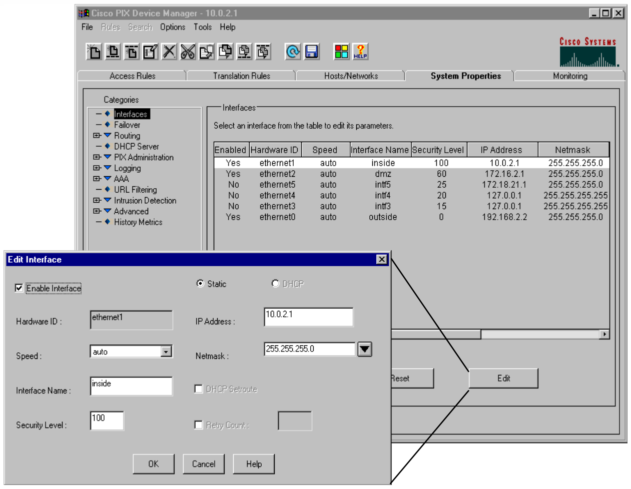
- Interfaces—In addition to their names, the Interfaces panel displays and enables you to edit additional configuration information required for each interface. Your configuration edits are captured by PDM but not sent to the PIX Firewall until **Apply to PIX** is clicked.
- Failover—Enables you to enable, disable, and configure failover and stateful failover.
- Routing—The routing panel is divided into the following three sections dealing with different routing configurations:
  - RIP
  - Static Routes
  - Proxy ARPs
- PIX Administration—This panel consists of several sections which contain the following administration configuration options:
  - Authentication
  - Password
  - PDM/HTTPS
  - Telnet
  - Secure Shell
  - SNMP
  - ICMP
  - TFTP Server

- Logging—This panel is divided into the following sections:
  - Logging Setup
  - PDM Logging
  - Syslog
  - Others
- AAA—This panel contains the following sections:
  - AAA Server Groups
  - AAA Servers
  - Auth. Prompt
- URL Filtering—This panel enables you prevent users from accessing external WWW URLs that you designative using the Websense URL filtering server.
- Intrusion Detection—This panel is divided into the following two sections:
  - IDS Policy
  - IDS Signatures
- Advanced—This panel is made up of the five panels listed below, with the FixUp panel having further selections nested beneath it.
  - Fixup
    - FTP
    - H.232
    - HTTP
    - RSH
    - RTSP
    - SIP
    - Skinny
    - SMTP
    - SQL*Net
  - Anti-Spoofing
  - Fragment
  - TCP Options
  - Timeout
- History Metrics—This panel enables the PIX Firewall to keep a history of many statistics, which can be displayed by PDM through the Monitoring tab.

**Note** If PDM History Metrics is not enabled, the only view available in the Monitoring tab is the "Real-time" view. PDM History Metrics is enabled by default.

# Interface Panel

**The Interfaces panel enables you to enable, disable, and edit the configuration of network interfaces.**



Interfaces

Select an interface from the table to edit its parameters.

| Enabled | Hardware ID | Speed | Interface Name | Security Level | IP Address | Netmask |
|---|---|---|---|---|---|---|
| Yes | ethernet1 | auto | inside | 100 | 10.0.2.1 | 255.255.255.0 |
| Yes | ethernet2 | auto | dmz | 60 | 172.16.2.1 | 255.255.255.0 |
| No | ethernet5 | auto | intf5 | 25 | 172.18.21.1 | 255.255.255.0 |
| No | ethernet4 | auto | intf4 | 20 | 127.0.0.1 | 255.255.255.255 |
| No | ethernet3 | auto | intf3 | 15 | 127.0.0.1 | 255.255.255.255 |
| Yes | ethernet0 | auto | outside | 0 | 192.168.2.2 | 255.255.255.0 |

☑ Enable Interface

● Static ○ DHCP

Hardware ID : ethernet1

IP Address : 10.0.2.1

Speed : auto

Netmask : 255.255.255.0

Interface Name : inside

☐ DHCP Setroute

Security Level : 100

☐ Retry Count :

© 2002, Cisco Systems, Inc. www.cisco.com CSPFA 2.1—17-25

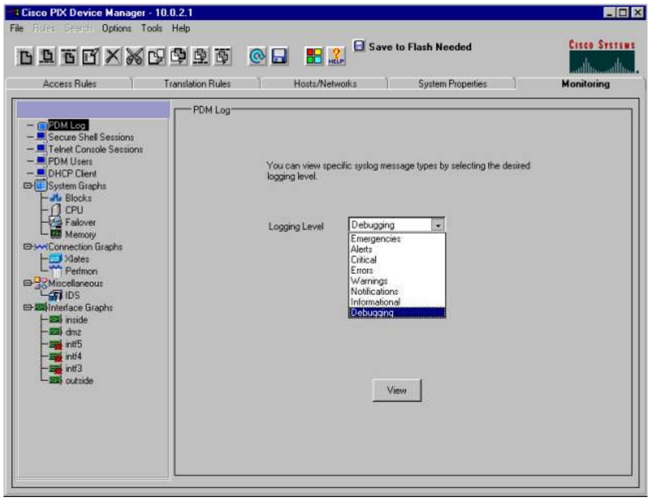
The PIX Firewall requires that you configure and then enable each interface that will be active. Interfaces you are not currently using can be disabled. When disabled, the interface will not transmit or receive data, but the configuration information is retained.

The physical location of each interface and corresponding connector on the PIX Firewall can be identified by its Hardware ID name, such as ethernet0 or ethernet1. The Interface Name is a logical name that relates to how it is used in your network configuration. For example, *inside* connects to your internal network and *outside* connects to an external network or the public Internet.

In addition to their names, this panel displays and enables you to edit additional configuration information required for each interface. Your configuration edits are captured by PDM but not sent to the PIX Firewall until **Apply to PIX** is clicked.

# Monitoring

**The Monitoring tab enables you to access the various monitoring features of PDM.**



You can view specific syslog message types by selecting the desired logging level.

Logging Level: Debugging

www.cisco.com

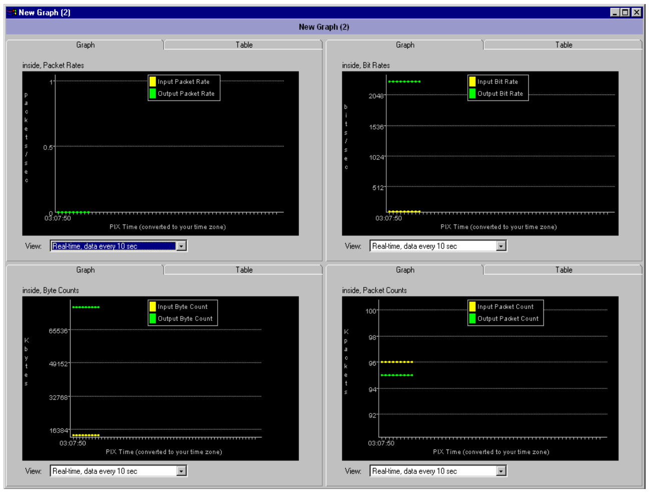
CSPFA 2.1—17-26

Many different items can be monitored using PDM, including but not limited to the following:

- PDM Log
- Secure Shell Sessions
- Telnet Console Settings
- PDM Users
- System Performance Graphs
- Connection statistics

# Interface Graphs Panel

**The Interface Graphs panel enables you to monitor per-interface statistics, such as packet counts and bit rates, for each enabled interface on the PIX Firewall.**



View: Real-time, data every 10 sec

View: Real-time, data every 10 sec

View: Real-time, data every 10 sec

View: Real-time, data every 10 sec

www.cisco.com CSPFA 2.1—17-27

The Interface Graphs panel enables you to monitor per-interface statistics, such as packet counts and bit rates, for each enabled interface on the PIX Firewall.

The list of graphs available is the same for every interface. Each graph can be viewed as a line graph and in table form. Each graph can also be viewed with different time horizons.

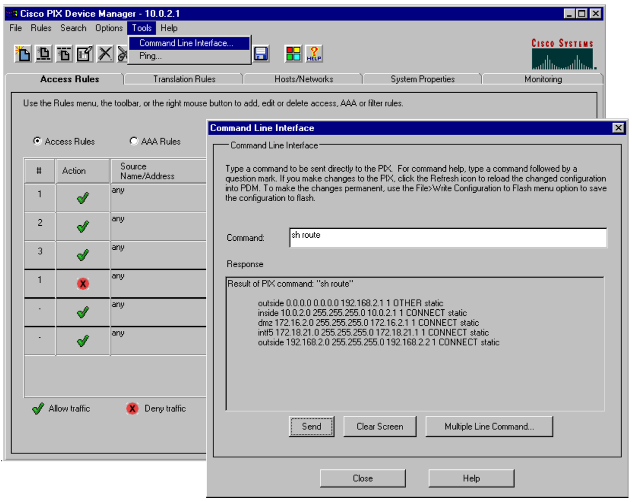
**Note** If an interface is not enabled using the Interfaces panel under the System Properties panel, no graphs will be available for that interface.

# Other Tools

This section details additional tools PDM provides for your use.

## Command Line Interface

**This panel provides a text-based tool for sending CLI commands to the PIX Firewall and displaying responses.**



Use the Rules menu, the toolbar, or the right mouse button to add, edit or delete access, AAA or filter rules.

| # | Action | Source Name/Address |
|---|---|---|
| 1 | ✓ | any |
| 2 | ✓ | any |
| 3 | ✓ | any |
| 1 | ✗ | any |
| - | ✓ | any |
| - | ✓ | any |

Type a command to be sent directly to the PIX. For command help, type a command followed by a question mark. If you make changes to the PIX, click the Refresh icon to reload the changed configuration into PDM. To make the changes permanent, use the File>Write Configuration to Flash menu option to save the configuration to flash.

Command: sh route

Response

```
Result of PIX command: "sh route"

        outside 0.0.0.0 0.0.0.0 192.168.2.1 1 OTHER static
        inside 10.0.2.0 255.255.255.0 10.0.2.1 1 CONNECT static
        dmz 172.16.2.0 255.255.255.0 172.16.2.1 1 CONNECT static
        intf5 172.18.21.0 255.255.255.0 172.18.21.1 1 CONNECT static
        outside 192.168.2.0 255.255.255.0 192.168.2.2 1 CONNECT static
```

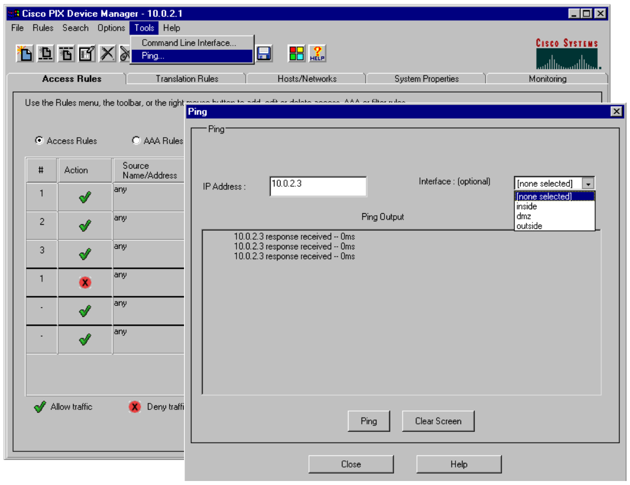
© 2002, Cisco Systems, Inc. www.cisco.com CSPFA 2.1—17-29

PDM panels generate commands and arguments that are sent to the PIX Firewall and applied to the running configuration. PDM receives results, in the form of messages, which provide information about the acceptance and effect of the command. The CLI tool enables administrators to enter those commands directly, send them to the PIX Firewall where they are immediately applied, and view the resulting messages.

Commands are entered as a single line in the Command field. Send, or the keyboard Enter key, transmits the commands to the PIX Firewall and the response is viewed in the Response field. The command and result text is retained in the Response field for a history of the session until erased by clicking the **Clear** button. By clicking **Multiple Line Commands**, multiple lines of commands may also be entered or pasted in from other sources, and then sent as a list of commands using the **Send** button.

# Ping Tool

This panel provides a ping tool which is useful for verifying the configuration and operation of a PIX Firewall and surrounding communications links, as well as the basic testing of other network devices.



Cisco PIX Device Manager - 10.0.2.1

Use the Rules menu, the toolbar, or the right mouse button to add, edit or delete access, AAA or filter rules.

● Access Rules ○ AAA Rules

| # | Action | Source Name/Address |
|---|---|---|
| 1 | ✓ | any |
| 2 | ✓ | any |
| 3 | ✓ | any |
| 1 | ☒ | any |
| - | ✓ | any |
| - | ✓ | any |

✓ Allow traffic ☒ Deny traffi

Ping

IP Address : 10.0.2.3

Interface : (optional) [none selected]

Ping Output

```
10.0.2.3 response received -- 0ms
10.0.2.3 response received -- 0ms
10.0.2.3 response received -- 0ms
```

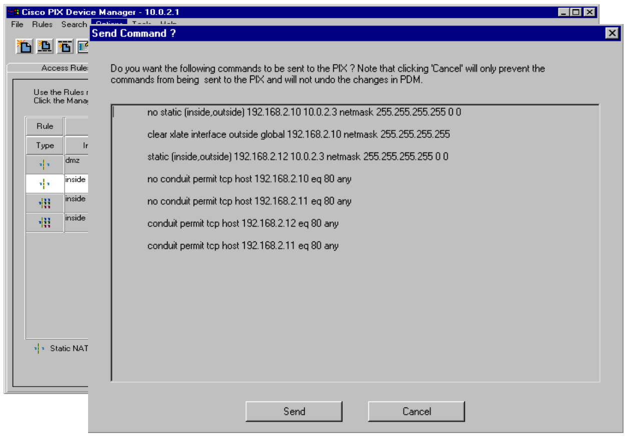
www.cisco.com CSPFA 2.1—17-30

A ping is the network equivalent of sonar for submarines. A ping is sent to an IP address and it returns an "echo." This process enables network devices to test each other.

In PDM's ping panel, you can optionally select an interface from which to ping. This allows you to determine if an IP address is accessible from the PIX Firewall or from a particular interface on the PIX Firewall.

# Preview Command Tool

**The Preview Commands Before Sending to PIX option enables you to preview any proposed configuration changes to the PIX Firewall before they are applied.**



Do you want the following commands to be sent to the PIX ? Note that clicking 'Cancel' will only prevent the commands from being  sent to the PIX and will not undo the changes in PDM.

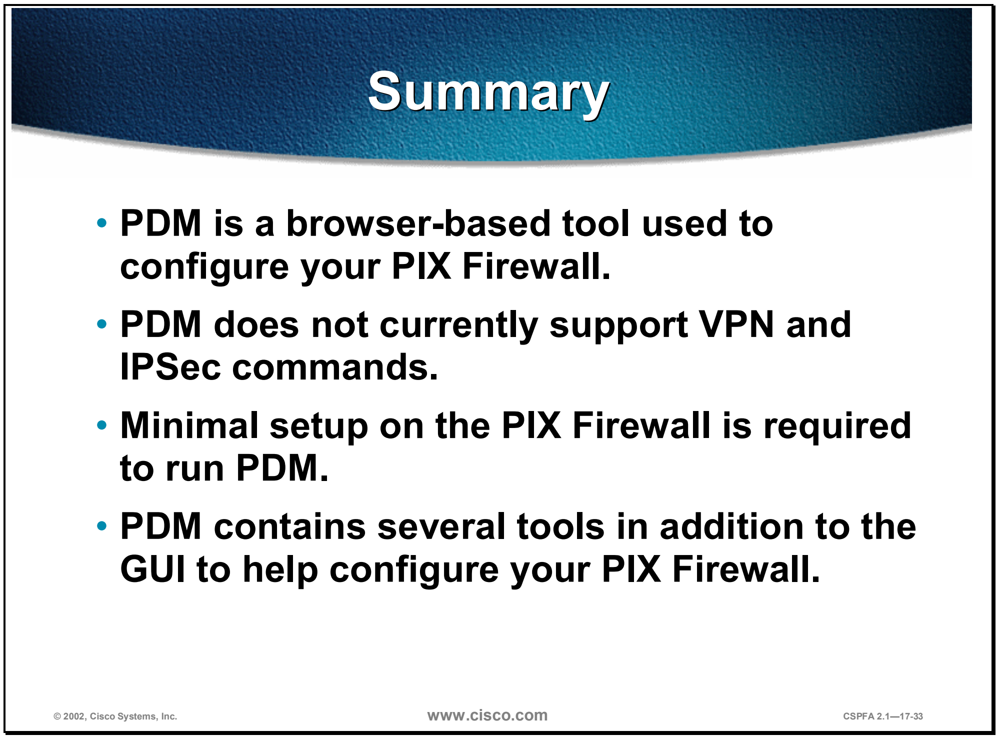
```
no static (inside,outside) 192.168.2.10 10.0.2.3 netmask 255.255.255.255 0 0
clear xlate interface outside global 192.168.2.10 netmask 255.255.255.255
static (inside,outside) 192.168.2.12 10.0.2.3 netmask 255.255.255.255 0 0
no conduit permit tcp host 192.168.2.10 eq 80 any
no conduit permit tcp host 192.168.2.11 eq 80 any
conduit permit tcp host 192.168.2.12 eq 80 any
conduit permit tcp host 192.168.2.11 eq 80 any
```

www.cisco.com CSPFA 2.1—17-31

Choosing **Options> Preview Commands Before Sending to PIX** enables you to preview any commands generated by any panel before they are sent to the PIX Firewall.

# Summary

This section summarizes the information you learned in this chapter.



## Summary

- PDM is a browser-based tool used to configure your PIX Firewall.
- PDM does not currently support VPN and IPSec commands.
- Minimal setup on the PIX Firewall is required to run PDM.
- PDM contains several tools in addition to the GUI to help configure your PIX Firewall.

www.cisco.com

CSPFA 2.1—17-33

# Lab Exercise—Configuring the PIX Firewall with PDM

Complete the following lab exercise to practice what you learned in this chapter.

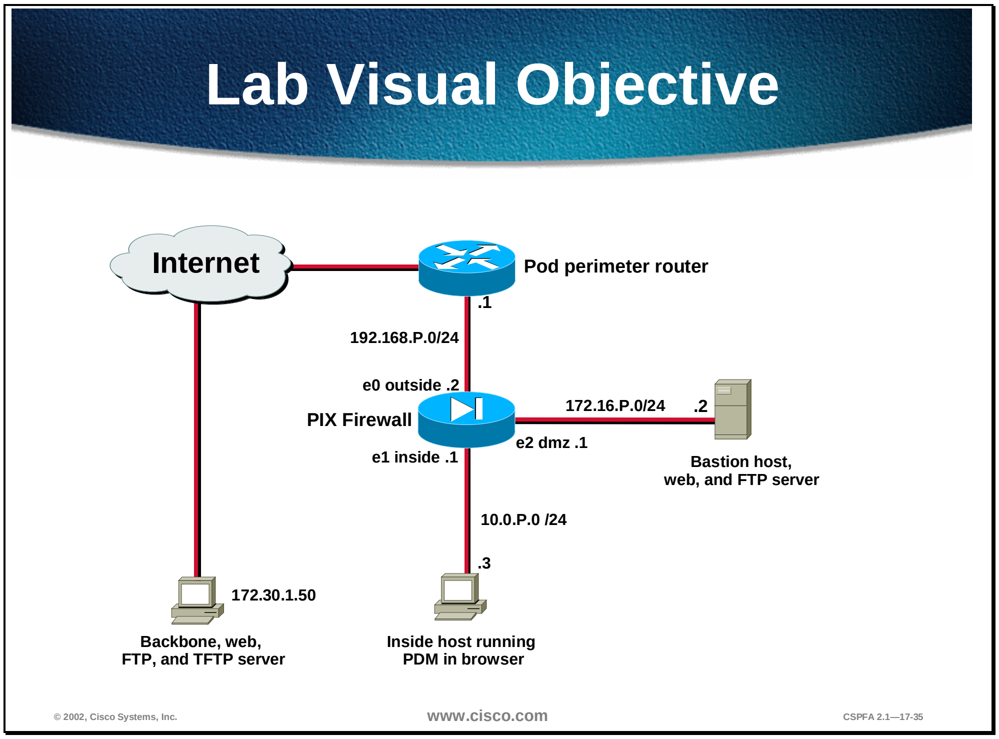
## Objectives

In this lab exercise you will complete the following tasks:

- Install PDM.
- Configure inside to outside access through your PIX Firewall using PDM.
- Configure outside to inside access through the PIX Firewall using PDM.
- Test and verify the PDM operation.

## Visual Objectives

The following figure displays the configuration you will complete in this lab exercise.



Lab Visual Objective

© 2002, Cisco Systems, Inc. www.cisco.com CSPFA 2.1—17-35

**Note** In the following lab exercise, you will bypass the initial security alert regarding the site security certificate. However, remember that when you remotely configure your PIX Firewall with PDM, you can use the security certificate for secure encrypted communication between PDM and the PIX Firewall. To do this, install the certificate by clicking **View Certificate** in the initial Security Alert window and following the prompts. Since the certificate is assigned to your PIX Firewall by name rather than by IP address, you will need to establish the connection with the PIX Firewall by entering its fully qualified domain name, rather than the IP address, in your browser. Using the name rather than an IP address requires that name resolution is enabled through DNS or a hosts file.

## Task 1—Install PDM and Access It from Your Browser

**Step 1** Enter the following commands to install PDM and access it from your browser:

**Step 1** Load the PDM file into the PIX Firewall:

```
pixP(config)# copy tftp://172.26.26.50/pdm-111.bin flash:pdm
```

**Step 2** Enable the HTTP server in the PIX Firewall:

```
pixP(config)# http server enable
```

**Step 3** Grant permission for your inside host to initiate an HTTP connection to the PIX Firewall:

```
pixP(config)# http 10.0.P.3 255.255.255.0 inside
```

(where P = pod number)

**Step 4** Access the PDM console by completing the following sub-steps:

1. Open your browser and enter https://10.0.P.1

(where P = pod number)

2. In the Security Alert window, click **Yes**.
3. When prompted for your username and password, do not enter a username or password. Click **OK** to continue.
4. Click **Yes** in the Security Warning window. If the Update Config window opens, click **Proceed**.

**Step 5** When the PDM console opens, notice that your current PIX Firewall configuration has been imported. Examine the configuration by completing the following sub-steps:

1. Click the **Access Rules** tab. Notice that an access policy has been created to correspond to the following conduits, which you configured earlier in the course:

```
conduit permit tcp host 192.168.P.10 eq www any
conduit permit tcp host 192.168.P.11 eq www any
conduit permit tcp host 192.168.P.11 eq ftp any
conduit permit icmp any any
```

2. Click the **Translation Rules** tab. Notice that your static mappings, NAT, and global pools appear here.

3. Click the **Hosts/Networks** tab and observe your network topology.
4. Click the **System Properties** tab. Notice that the configuration of the PIX Firewall interfaces is displayed.

**Step 6** Close your browser.

**Step 7** The Configuration Modified window appears.

**Step 8** Click **Save** to save the configuration to Flash memory. The Save Successful window opens.

**Step 9** Click **OK**.

## Task 2—Test PDM's Warning for Unsupported Commands

To verify that PDM will warn you of unsupported commands, complete the following steps:

**Step 1** Configure the PIX Firewall with an **access-list** command that permits IP traffic from a peer pod's internal network to your internal network:

```
pixP(config)# access-list ACLNAME permit ip 192.168.Q.0 255.255.255.0 192.168.P.0
255.255.255.0
pixP(config)# access-group ACLNAME in interface outside
```

(where P = pod number, and Q = peer pod number)

**Step 2** Display your ACL:

```
pixP(config)# show access-list
access-list ACLNAME permit ip 192.168.Q.0 255.255.255.0 192.168.P.0 255.255.255.0
```

(where P = pod number, and Q = peer pod number)

**Step 3** Verify that conduits exist in the configuration along with ACLs by displaying your conduits:

```
pixP(config)# show conduit
conduit permit tcp host 192.168.P.10 eq www any (hitcnt=4)
conduit permit tcp host 192.168.P.11 eq www any (hitcnt=4)
conduit permit tcp host 192.168.P.11 eq ftp any (hitcnt=4)
conduit permit icmp any any (hitcnt=8)
```

**Step 4** Access the PDM console by doing the following:

1. In your browser, enter **https://10.0.P.1**.

(where P = pod number)

2. In the Security Alert window, click **Yes**.
3. When prompted for your username and password, do not enter a username or password. Click **OK** to continue.
4. Click **Yes** in the Security Warning window. The Unsupported Command Found window opens. This is because PDM does not support configurations that use both conduits and ACLs.
5. Read the statement in the window.
6. Click **OK** in the Unsupported Command Found window.

7. Click the following tabs to verify that they are disabled: Access Rules, Translation Rules, Hosts/Networks, and System Properties
8. Minimize but do *not* close the PDM console.

**Step 5** In your Telnet window, remove the ACL to restore full PDM capability:

```
pixP(config)# no access-list ACLNAME
```

**Step 6** Verify that the ACL has been removed:

```
pixP(config)# show access-list
```

**Step 7** Show the access group:

```
pixP(config)# show access-group
```

**Step 8** Maximize the PDM console.

**Step 9** Click **File** in the main menu. The drop-down File menu opens.

**Step 10** Choose **Refresh PDM with Current Configuration from PIX**.

**Step 11** Verify that the ACL has been removed by completing the following sub-steps:

1. Choose **Tools**>**Command Line Interface** from the main menu.
2. In the Command box, enter **show access-list**.

**Step 12** Close your browser.

## Task 3—Clear the PIX Firewall's Configuration, and Access the PDM Startup Wizard

Complete the following steps to erase your current PIX Firewall configuration and access the PDM wizard:

**Step 1** In your Telnet window, erase your current PIX Firewall configuration:

```
pixfirewall(config)# write erase
```

**Step 2** In your Telnet window, reload the PIX Firewall:

```
pixP(config)# reload
```

**Step 3** When prompted to pre-configure the PIX Firewall through interactive prompts, press **Enter**.

**Step 4** Agree to use the current password by pressing **Enter**:

```
Enable password [<use current password>]: <Enter>
```

**Step 5** Accept the default year by pressing **Enter**:

```
Clock (UTC):
  Year [2001]: <Enter>
```

**Step 6** Accept the default month by pressing **Enter**:

```
Month [Nov]: <Enter>
```

**Step 7** Accept the default day by pressing **Enter**:

```
Day [14]: <Enter>
```

**Step 8** Accept the default time stored in the host computer by pressing **Enter**:

```
Time [11:21:25]: <Enter>
```

**Step 9** Enter the IP address of your PIX Firewall's inside interface:

```
Inside IP address: 10.0.P.1
```

(where P = pod number)

**Step 10** Enter the network mask that applies to inside IP address:

```
Inside network mask: 255.255.255.0
```

**Step 11** Enter the host name you want to display in the PIX Firewall command line prompt:

```
Host name: pixP
```

(where P = pod number)

**Step 12** Enter the DNS domain name of the network on which the PIX Firewall runs:

```
Domain name: cisco.com
```

**Step 13** Enter the IP address of the host running PDM:

```
IP address of host running PIX Device Manager: 10.0.P.3
```

(where P = pod number)

**Step 14** Enter **y** at the prompt to save the information to the PIX Firewall's Flash memory.

**Step 15** Access the PDM console by doing the following:

1. In your browser, enter **https://10.0.P.1**.

(where P = pod number)

2. In the Security Alert window, click **Yes**.
3. When prompted for your username and password, do not enter a username or password. Click **OK** to continue.
4. Click **Yes** in the Security Warning window. The PIX Device Manager Startup Wizard opens.

## Task 4—Use the PDM Startup Wizard to Perform Basic Configuration Tasks

Complete the following steps to configure the PIX Firewall's outside and DMZ interfaces, establish a default route, configure NAT, and create a global pool of addresses for address translation:

**Step 1** In the PIX Device Manager Startup Wizard window, click **Next**. The Interface Configuration group box appears.

**Step 2** Enable the outside and DMZ interfaces for 100 Mbps Ethernet full duplex communication by completing the following sub-steps:

1. Choose **100full** for ethernet0 and ethernet2 from the drop-down menu in the Speed column.
2. Select the **Enable** boxes for ethernet0 and ethernet2.

**Step 3** Assign the name dmz to ethernet2 by highlighting **intf2** in the Name column, and entering **dmz**.

**Step 4** Assign an IP address and subnet mask to ethernet0 by completing the following sub-steps:

1. Highlight **127.0.0.1** in the Ethernet0 IP address column, and enter **192.168.P.2**.
2. Choose **255.255.255.0** from the drop-down menu next to the IP address you just entered.

**Step 5** Assign an IP address and subnet mask to the DMZ interface by completing the following sub-steps:

1. Highlight **127.0.0.1** in the ethernet2 IP Address column and enter **172.16.P.1**.

(where P = pod number)

2. Choose **255.255.255.0** from the drop-down menu next to the IP address you just entered.

**Step 6** Click **Next** in the Interface Configuration window. The Default Route Configuration screen appears.

**Step 7** Configure a default route by entering **192.168.P.1** in the Internet Address of Router field.

(where P = pod number)

**Step 8** Click **Next**. The Address Translation screen appears.

**Step 9** Verify that Hide Protected Addresses is selected and click **Next**. The Network Address Translation screen appears.

**Step 10** Configure a global pool of addresses to be used for address translation by completing the following sub-steps:

1. Enter **192.168.P.20** in the Starting IP address field.

(where P = pod number)

2. Enter **192.168.P.254** in the Ending IP address field.

(where P = pod number)

3. Select **255.255.255.0** from the drop-down menu.
4. Click **Next**. The Port Address Translation screen appears.

**Step 11** Click **Next**. The PIX Device Manager screen appears informing you that you have entered all the information to perform an initial configuration of the PIX Firewall.

**Step 12** Click **Finish**.

**Step 13** Click **OK** to start the PDM application. The Update Config window opens.

**Step 14** Click **Proceed**.

## Task 5—Verify the Configuration Created by the PDM Wizard and Configure the DMZ Interface

Complete the following steps to verify the configuration of the PIX Firewall's outside and DMZ interfaces, the global address pool, routing, and NAT:

**Step 1** Select the **System Properties** tab.

**Step 2** From the Systems Properties tab, complete the following sub-steps:

1. Verify that ethernet0, ethernet1, and ethernet2 are enabled.
2. Verify that ethernet0, ethernet1, and ethernet2 are set to 100 Mbps Ethernet full-duplex communication.
3. Verify that ethernet0, ethernet1, and ethernet2 are correctly named.
4. Verify that ethernet0 has a security level of 0 and ethernet1 has a security level of 100.
5. Verify the IP addresses and subnet masks of ethernet0, ethernet1, and ethernet2.

**Step 2** Verify the NAT configuration and global address pool you entered earlier by completing the following sub-steps:

1. Select the Translation Rules tab.
2. Click the **Manage Pools** button and verify the global address pool.
3. Click **Done**.

**Step 3** Verify your default route by completing the following sub-steps:

1. Select the System Properties tab
2. Expand **Routing** under Categories.
3. Select **Static Route**.
4. Verify that the gateway under Gateway IP in the Static Route group box is 192.168.P.1.

(where P = pod number)

**Step 4** Configure a privileged mode password by completing the following sub-steps:

1. Expand **PIX Administration** from the Categories tree on the left side of the panel. Password appears under PIX Administration.
2. Select **Password**. The Password group box appears on the right side of the panel.
3. Enter **cisco** in the New Password field in the Enable Password group box.
4. Enter **cisco** in the Confirm New Password field in the Enable password group box.
5. Click **Apply to PIX** in the Enable Password group box. The Information window opens.
6. Click **OK**.

**Step 5** Close your web browser.

**Step 6** Click **Save** in the Config Modified window.

**Step 7** Click **OK** in the Save Successful window.

**Step 8** Access the PDM console by completing the following sub-steps:

1. In your browser, enter **https://10.0.P.1**.

(where P = pod number)

2. Click **Yes** in the Security Alert window.

3. When prompted for your username and password, do not enter a username, but enter a password of **cisco**. Click **OK** to continue.
4. Click **Yes** in the Security Warning window.

**Step 9** Enable command preview by completing the following sub-steps:

1. Choose **Options>Preferences** from the main menu. The Preferences window opens.
2. Select **Preview Commands Before Sending to PIX**.
3. Click **OK**.

**Step 10** Assign the DMZ interface a security level of 50 by completing the following sub-steps:

1. Select the System Properties tab
2. Select the **Interfaces** icon.
3. Select **Ethernet2** in the Interfaces group box.
4. Click **Edit**. The Edit Interface window opens.
5. Change the security level to **50** in the Security Level field of the Edit Interface window.
6. Click **OK**.
7. Click **OK** in the Security Level Change window.
8. Click **Apply to PIX**.
9. Click **Send** in the Send Command window.

**Step 11** Write the configuration to the Flash memory by choosing **File>Write Configuration to Flash**. The Send Command window opens.

**Step 12** Click **Send.** The Save Successful window opens.

**Step 13** Click **OK**.

## Task 6—Test the Inside, Outside, and DMZ Interface Connectivity

Perform the following steps to test NAT and interface connectivity:

**Step 1** Test the operation of the global and NAT you configured by originating connections through the PIX Firewall. To do this, complete the following sub-steps:

1. Open a web browser on the Windows NT server.
2. Use the web browser to access the Super Server at IP address 172.26.26.50 by entering **http://172.26.26.50**.

**Step 2** Observe the translation table by completing the following sub-steps:

1. Choose **Tools> Command Line Interface**. The Command Line Interface window opens.
2. In the Command field, enter **show xlate**.
3. Click **Send**.

4. Observe the output in the Response field. It should appear similar to the following:

```
Result of the PIX command: "show xlate"

1 in use, 1 most used
Global 192.168.P.20 Local 10.0.P.3
```

(where P = pod number)

Note that a global address chosen from the low end of the global range has been mapped to your NT laptop.

**Step 3** Exit the windows by clicking **Close**.

**Step 4** Test interface connectivity by completing the following sub-steps:

1. Choose **Tools> Ping**.
2. In the IP Address field, enter **10.0.P.1**.

(where P = pod number)

3. Click **Ping**.
4. Observe the following output in the Ping Output window. The output should appear similar to the following:

```
10.0.P.1 response received -- 10ms
10.0.P.1 response received -- 10ms
10.0.P.1 response received -- 10ms
```

(where P = pod number)

5. Click **Clear Screen**.

**Step 5** Repeat Step 4 for the following IP addresses. You have successfully completed this task if responses are received for all pings.

- Your inside host: 10.0.P.3

(where P = pod number)

- The outside interface: 192.168.P.2

(where P = pod number)

- Your pod perimeter router: 192.168.P.1

(where P = pod number)

- The DMZ interface: 172.16.P.1

(where P = pod number)

- Your bastion host: 172.16.P.2

(where P = pod number)

**Step 6** Exit the Ping window by clicking **Close**.

## Task 7—Use PDM to Configure Access from Higher Security Level Interfaces to Lower Security Level Interfaces

Perform the following steps to configure NAT for the inside and DMW interfaces:

**Step 1** Remove NAT by completing the following sub-steps:

1. Select the **Translation Rules** tab.
2. Highlight the rule you configured earlier in the lab exercise.
3. Choose **Rules>Delete**.

**Step 2** Configure NAT for the internal network's range of IP addresses by completing the following sub-steps:

1. Click the **Rules** menu.
2. Click **Add**. The Add Address Translation Rule window opens.
3. Verify that the inside interface is selected in the Interface drop-down menu.
4. Click **Browse**. The Select host/network window opens.
5. Verify that the inside interface is selected in the Interface drop-down menu.
6. Click **inside 10.0.P.0**.

(where P = pod number)

7. Click **OK**.
8. Verify that outside is selected in the Translate address on less secure interface drop-down menu.
9. Verify that Dynamic is selected in the Translate Address to group box.
10. Verify that 1 is selected in the Address Pool drop-down menu.
11. Verify that the global pool you configured earlier (192.168.P.20–192.168.P.254) appears under Address.

(where P = pod number)

12. Click **OK** in the Add Address Translation Rule window. Your new rule appears on the Translation Rules tab.
13. Click **Apply to PIX**. The Send Command window opens.
14. Observe the commands that will be sent to the PIX Firewall.
15. Click **Send**.

**Step 3** Configure NAT for the DMZ network's range of IP addresses by completing the following sub-steps:

1. Click the **Rules** menu.
2. Click **Add**. The Add Address Translation Rule window opens.
3. Verify that the dmz interface is selected in the Interface drop-down menu.
4. Click **Browse**. The Select host/network window opens.
5. Verify that the dmz interface is selected in the Interface drop-down menu.

6. Click **dmz 172.16.P.0**.

(where P = pod number)

7. Click **OK**.
8. Verify that outside is selected in the Translate address on less secure interface drop-down menu.
9. Verify that Dynamic is selected in the Translate address to group menu.
10. Verify that 1 appears in the Address Pool drop-down menu.
11. Verify that the global pool you configured earlier (192.168.P.20–192.168.P.254) appears under Address.

(where P = pod number)

12. Click **OK** in the Add Address Translation Rule window. Your new rule appears on the Translation Rules tab.
13. Click **Apply to PIX**. The Send Command window opens.
14. Observe the commands that will be sent to the PIX Firewall.
15. Click **Send**.

**Step 4** Configure the PIX Firewall to allow access to the DMZ from the inside network. To do this, assign one pool of IP addresses for hosts on the public DMZ, and complete the following sub-steps:

1. Click the **Rules** menu.
2. Click **Add**. The Add Address Translation Rule window opens.
3. Verify that the inside interface is selected in the Interface drop-down menu.
4. Click **Browse**. The Select host/network window opens.
5. Verify that the inside interface is selected in the Interface drop-down menu.
6. Click **inside 10.0.P.0**.

(where P = pod number)

7. Click **OK**.
8. Verify that dmz is selected in the Translate address on less secure interface drop-down menu.
9. Verify that Dynamic is selected in the Translate address to group box.
10. Click **Manage Pools**. The Manage Global Address Pools window opens.
11. Select **dmz** under Interface.
12. Click **Add**. The Add Global Pool Item window opens.
13. Verify that dmz is selected in the Interface box.
14. Enter a Pool ID of **1**.
15. Verify that Range is selected in the Add Global Pool Item window.
16. Enter **172.16.P.20** in the first IP Address field.

(where P = pod number)

17. Enter **172.16.P.254** in the second IP Address field.

(where P = pod number)

18. Enter **255.255.255.0** in the Network Mask (optional) field.
19. Click **OK**. The Manage Global Address Pools window opens.
20. Click **Done**. The Add Address Translation Rule window reopens.
21. Choose **1** from the Address pool drop-down menu.
22. Click **OK**. Your new global pool appears in the Translation tab.
23. Click **Apply to PIX**. The Send Command window opens.
24. Click **Send**.

**Step 5** Add your bastion host to PDM and configure NAT for the DMZ by completing the following sub-steps:

1. Choose **dmz** from the Interface drop-down menu.
2. Click **Add**. The Create Host/Network window opens.
3. Enter **172.16.P.2** in the IP Address field.

(where P = pod number)

4. Choose **255.255.255.255** from the Mask drop-down menu.
5. Verify that dmz is selected in the Interface drop-down menu.
6. Enter **bastionhost** in the Name (Recommended) field.
7. Click **Next**. The Create host/network window opens.
8. Verify that Dynamic is selected in the NAT (Network Address Translation) group box.
9. Verify that 1 is selected in the Address Pool(s) drop-down menu.
10. Click **Finish**. Your bastionhost appears in the Hosts/Networks tab.
11. Click **Apply to PIX**. The Send Command window opens.
12. Observe the commands that will be sent to the PIX Firewall.
13. Click **Send**.

**Step 6** Write the current configuration to Flash memory by completing the following sub-steps:

1. Choose **File>Write Configuration to Flash**. The Send Command window opens.
2. Click **Send**.
3. Click **OK** in the Configuration Saved to Flash Memory group box.

**Step 7** Use the **clear xlate** command after configuring with the **nat** and **global** commands to make the global IP addresses available in the translation table by completing the following sub-steps:

1. Choose **Tools>Command Line Interface**. The Command Line Interface window opens.
2. Enter **clear xlate** in the Command field.
3. Click **Send**.

4. Click **Clear Screen**.
5. Enter **show xlate** in the Command field.
6. Click **Send**.
7. Verify that the output in the Response window is similar to the following:

```
Result of PIX command: "show xlate"
0 in use, 1 most used
```

8. Click **Close**.

## Task 8—Test Globals and NAT Configuration

To test the globals and NAT configuration, you must complete the following:

**Step 1** Test the operation of the global and NAT you configured by originating connections through the PIX Firewall. Complete the following sub-steps:

1. Open a web browser on the Windows NT server.
2. Use the web browser to access the super server at IP address 172.26.26.50 by entering **http://172.26.26.50**.

**Step 2** Observe the translation table with the **show xlate** command by completing the following sub-steps:

1. Choose **Tools>Command Line Interface**. The Command Line Interface window opens.
2. Enter **show xlate** in the Command field.
3. Click **Send**.
4. Verify that the output in the Response window is similar to the following:

```
Result of PIX command: "show xlate"
1 in use, 1 most used
Global 192.168.P.21 Local ntP
```

(where P = pod number)

5. Click **Close**.

**Step 3** Test the web access to your bastion host from the Windows NT server by completing the following sub-steps:

1. Open a web browser on the Windows NT server.
2. Use the web browser to access your bastion host by entering **http://172.16.P.2**.

(where P = pod number)

The home page of the bastion host should opens in your web browser.

**Step 4** Observe the transaction by completing the following sub-steps:

1. Open a web browser on the Windows NT server.
2. Choose **Tools>Command Line Interface**. The Command Line Interface window opens.
3. Enter **show arp** in the Command field.
4. Click **Send**.

5. Verify that the output in the Response window is similar to the following:

```
outside 192.168.P.1 00e0.1e41.8762
inside ntP 00e0.b05a.d509
dmz bastionhost 00e0.1eb1.78df
```

(where P = pod number)

6. Click **Clear Screen**.
7. Enter **show xlate** in the Command field.
8. Click **Send**.
9. Verify that the output in the Response window is similar to the following:

```
Result of PIX command: "show xlate"

2 in use, 2 most used
Global 172.16.P.20 Local ntP
Global 192.168.P.21 Local ntP
```

(where P = pod number)

10. Click **Clear Screen**.
11. Enter **show conn** in the Command field.
12. Click **Send**.
13. Verify that the output in the Response window is similar to the following:

```
Result of PIX command: "show conn"

0 in use, 4 most used
```

14. Click **Close**.

**Step 5** Test FTP access to the bastion host from your Windows NT server by completing the following sub-steps:

1. Establish an FTP session to the bastion host by choosing **Start>Run>ftp 172.16.P.2** (where P = pod number). You have reached the bastion host if you receive the message "Connected to 172.16.P.2."
2. Log into the FTP session:

```
User (172.16.P.2(none)): anonymous
331 Anonymous access allowed, send identity (e-mail name) as password.
Password: Cisco
```

(where P = pod number)

3. Quit the FTP session if you were able to connect and log in:

```
ftp> quit
```

## Task 9—Use PDM to Configure Access from Lower to Higher Security Levels

Complete the following steps to configure the PIX Firewall to permit outside access to hosts in the DMZ and to hosts on the inside interface:

**Step 1** Create a static translation for your inside host by completing the following sub-steps:

1. Select the Translation Rules tab.
2. Select the **Add a New Rule** icon in the toolbar. The Add Address Translation Rule window opens.
3. Verify that the inside interface is selected in the Interface drop-down menu.
4. Click **Browse**. The Select host/network window opens.
5. Verify that the inside interface is selected in the Interface drop-down menu.
6. Click **10.0.P.3**.
7. Click **OK**. The Add Address Translation Rule window opens.
8. Verify that outside is selected in the Translate Address on Less Secured Interface drop-down menu.
9. Choose **Static** from the Translate address to drop-down menu.
10. Enter **192.168.P.10** in the IP Address field.

(where P = pod number)

11. Click **OK**. Your new rule appears on the Translation Rules tab.
12. Click **Apply to PIX**. The Send Command window opens.
13. Observe the command that will be sent to the PIX Firewall.
14. Click **Send**.

**Step 2** Create a static translation for your bastion host by completing the following sub-steps:

1. Select the Translation Rules tab.
2. Click the **Rules** menu.
3. Click **Add**. The Add Address Translation Rule window opens.
4. Verify that the dmz interface is selected in the Interface drop-down menu.
5. Click **Browse**. The Select host/network window opens.
6. Verify that the dmz interface is selected in the Interface drop-down menu.
7. Click **bastionhost 172.16.P.2**.

(where P = pod number)

8. Click **OK**. The Add Address Translation Rule window opens.
9. Verify that outside is selected in the Translate Address on Less Secured Interface drop-down menu.
10. Select **Static** in the Translate address to field.

11. Enter **192.168.P.11** in the IP Address field.

(where P = pod number)

12. Click **OK**. Your new rule appears on the Translation Rules tab.
13. Click **Apply to PIX**. The Send Command window opens.
14. Observe the command that will be sent to the PIX Firewall.
15. Click **Send**.

**Step 3** Ping a peer pod’s inside host from your internal host. The ping should fail because the access policy does not yet allow it.

```
C:\> ping 192.168.Q.10
Pinging 192.168.Q.10 with 32 bytes of data:
Request timed out.
Request timed out.
Request timed out.
```

(where Q = peer’s pod number)

**Step 4** Configure an ACL to allow pinging through your PIX Firewall by completing the following sub-steps:

1. Select the **Access Rules** tab.
2. Choose **Rules** from the main menu.
3. Click **Add**. The Add Rule window opens.
4. Verify that permit is selected in the Select an action drop-down menu.
5. Choose **outside** from the Interface drop-down menu in the Source Host/Network group box.
6. Choose **inside** from the Interface drop-down menu in the Destination Host/Network group box.
7. Select **ICMP** in the Protocol or Service group box.
8. Verify that any is selected in the ICMP type group box.
9. Click **OK**. Your new rule appears on the Access Rules tab.
10. Click **Apply to PIX**. The Send Command window opens.
11. Observe the ACLs to be sent to the PIX Firewall.
12. Click **Send**.

**Step 5** Ping a peer pod’s inside host from your internal host. Be sure to coordinate with your peer pod.

```
C:\> ping 192.168.Q.10

Pinging 192.168.Q.10 with 32 bytes of data:
Reply from 192.168.Q.10:  bytes=32 time<10ms TTL=125>
Reply from 192.168.Q.10:  bytes=32 time<10ms TTL=125>
Reply from 192.168.Q.10:  bytes=32 time<10ms TTL=125>
Reply from 192.168.Q.10:  bytes=32 time<10ms TTL=125>
```

(where Q = peers pod number)

**Step 6** Configure an ACL to allow Web access to the bastion host from the outside by completing the following sub-steps:

1. Select the **Access Rules** tab.
2. Choose **Rules>Add** from the main menu. The Add Rule window opens.
3. Verify that permit is selected in the Select an action drop-down menu.
4. Choose **outside** from the Interface drop-down menu within the Source Host/Network group box.
5. Select **dmz** in the Interface drop-down menu within the Destination Host/Network group box.
6. Click **Browse** in the Destination Host/Network group box. The Select host/network window opens.
7. Verify that dmz is selected in the Interface drop-down menu.
8. Choose **bastionhost 172.16.P.2**.

(where P = pod number)

9. Click **OK**. The Add Rule window becomes active.
10. Select **TCP** in the Protocol or Service group box.
11. Verify that any is selected in the Source Port text box.
12. Verify that = is selected in the drop-down menu under Destination Port.
13. Click the ellipsis button under Destination Port. The Service pop-up window opens.
14. Choose **http** from the drop-down menu under Destination Port.
15. Click **OK**. The Add Rule window becomes active.
16. Click **OK**.
17. Click **Apply to PIX**. The Send Command window re-opens.
18. Observe the ACLs to be sent to the PIX Firewall.
19. Click **Send**.

**Step 7** Configure an ACL to allow FTP access to the bastion host from the outside by completing the following sub-steps:

1. Select the **Access Rules** tab.
2. Choose **Rules>Add**. The Add Rule window opens.
3. Verify that permit is selected in the Select an action drop-down menu.
4. Choose **outside** from the Interface drop-down menu in the Source Host/Network group box.
5. Choose **dmz** from the Interface drop-down menu in the Destination Host/Network group box.
6. Click **Browse** in the Destination Host/Network group box. The Select host/network window opens.
7. Verify that dmz is selected in the Interface drop-down menu.
8. Select **bastionhost 172.16.P.2**.

(where P = pod number)

9. Click **OK**. The Add Rule window becomes active.
10. Select **TCP** in the Protocol or Service group box.
11. Verify that any is selected in the Source Port text box.
12. Verify that = is selected in the drop-down menu under Destination Port.
13. Click the ellipsis button under Destination Port. The Service window opens.
14. Select **FTP**.
15. Click **OK**. The Add Rule window becomes active.
16. Click **OK**.
17. Click **Apply to PIX**. The Send Command window opens.
18. Observe the ACLs to be sent to the PIX Firewall.
19. Click **Send**.

**Step 8** Configure an ACL to allow Web access to the inside host from the outside by completing the following sub-steps:

1. Select the **Access Rules** tab.
2. Choose **Rules>Add**. The Add Rule window opens.
3. Verify that permit is selected in the Select an action drop-down menu.
4. Choose **outside** from the Interface drop-down menu within the Source Host/Network group box.
5. Choose **inside** from the Interface drop-down menu within the Destination Host/Network group box.
6. Click **Browse** in the Destination Host/Network group box. The Select host/network window opens.
7. Verify that inside is selected in the Interface drop-down menu.
8. Select **10.0.P.3**.

(where P = pod number)

9. Click **OK**. The Add Rule window becomes active.
10. Select **TCP** in the Protocol and Service group box.
11. Verify that any is selected in the Source Port text box.
12. Verify that = is selected in the drop-down menu under Destination Port.
13. Click the ellipsis button under Destination Port. The Service window opens.
14. Select **http** from the Service group box under Destination Port.
15. Click **OK**. The Add Rule window re-opens.
16. Click **OK**.
17. Click **Apply to PIX**. The Send Command window appears.
18. Observe the ACLs to be sent to the PIX Firewall.
19. Click **Send**.

**Step 9** Clear current translations by completing the following sub-steps:

1. Choose **Tools>Command Line Interface**. The Command Line Interface window opens.
2. Enter **clear xlate** in the Command field.
3. Click **Send**.
4. Verify that the output in the Response box is similar to the following:

```
Result of PIX command: "clear xlate"
The command has been sent to PIX.
```

**Step 10** View current translations by completing the following sub-steps:

1. Click **Clear Screen** in the Command Line Interface window.
2. Enter **show xlate** in the Command field.
3. Click **Send**.
4. Verify that the output in the Response box is similar to the following:

```
Result of PIX command: "show xlate"
0 in use, 3 most used
```

5. Click **Close** in the Command Line Interface window.

**Step 11** Test Web access to the bastion hosts of opposite pod groups by completing the following sub-steps:

1. Open a web browser on the client PC.
2. Use the web browser to access the bastion host of your peer pod group by entering **http://192.168.Q.11** (where Q = peer pod number). You should be able to access the IP address of the static mapped to the bastion host of the opposite pod group.
3. Have an opposite pod group attempt to access your bastion host in the same way.

**Step 12** Test FTP access to the bastion hosts of other pod groups by completing the following sub-steps:

1. On your FTP client, access the bastion host of another pod group by choosing **Start>Run>ftp 192.168.Q.11** (where Q = peer pod number). You should be able to access your peer's bastion host via FTP.
2. Have an opposite pod group use FTP to access your bastion host.

**Step 13** Test Web access to the inside hosts of opposite pod groups by completing the following sub-steps:

1. Open a web browser on the client PC.
2. Use the web browser to access the inside host of your peer pod group http://192.168.Q.10 (where Q = peer pod number). You should be able to establish a Web connection to your peer's inside host.
3. Have an opposite pod group test your static and ACL configuration in the same way.

**Step 14** Test FTP access to the inside hosts of other pod groups by completing the following sub-steps:

1. On your client PC, use FTP to get into the inside host of another pod group by choosing **Start>Run>ftp 192.168.Q.10** (where Q = peer pod number).. You should be unable to access your peer's inside host via FTP.
2. Have an opposite pod group use FTP to attempt to get into your inside host.

**Step 15** Observe the transactions by completing the following sub-steps:

1. Choose **Tools>Command Line Interface**. The Command Line Interface window opens.
2. Enter **show arp** in the Command field.
3. Click **Send**.
4. Verify that the output in the Response box is similar to the following:

```
result of PIX command: "show arp"

outside 192.168.P.1 0003.6ba4.ca60
inside 10.0.P.3 0050.da31.6130
dmz bastionhost 000d.b782.3431
```

(where P = pod number)

5. Click **Clear Screen**.
6. Enter **show conn** in the Command field.
7. Click **Send**.
8. Verify that the output in the Response box is similar to the following:

```
result of PIX command: "show conn"
0 in use, 6 most used
```

9. Click **Clear Screen**.
10. Enter **show xlate** in the Command field.
11. Click **Send**.
12. Verify that the output in the Response box is similar to the following:

```
result of PIX command: "show xlate"

2 in use, 3 most used
Global 192.168.P.10 Local 10.0.P.3 static
Global 192.168.P.11 Local bastionhost static
```

(where P = pod number)

13. Click **Close**.

## Task 10—Use PDM to Configure the PIX Firewall to Permit ICMP Packets

Complete the following steps to test current access through the PIX Firewall, and then configure the PIX Firewall to allow ICMP packets between the inside and DMZ interfaces:

**Step 1** From your inside host, ping your bastion host:

```
C:\> ping 172.16.P.2
```

```
Pinging 172.16.P.2 with 32 bytes of data:
Request timed out.
Request timed out.
Request timed out.
```

(where P = pod number)

**Step 2** Configure an ACL to permit ICMP packets between the inside and dmz interfaces by completing the following sub-steps:

1. Select the **Access Rules** tab.
2. Choose **Rules>Add**. The Add Rule window opens.
3. Verify that Permit is selected in the Select an action drop-down menu.
4. Choose **dmz** from the Interface drop-down menu under Source Host/Network.
5. Choose **inside** from the Interface drop-down menu under Destination Host/Network.
6. Select **icmp** in the Protocol and Service group box.
7. Click **OK**. You are returned to the Access Rules tab.
8. Click **Apply to PIX**. The Send Command window opens.
9. Click **Send**.

**Step 3** From your inside host, ping your bastion host:

```
C:\> ping 172.16.P.2

Pinging 172.16.P.2 with 32 bytes of data:
Reply from 172.16.P.2:  bytes=32 time<10ms TTL=128
Reply from 172.16.P.2:  bytes=32 time<10ms TTL=128
Reply from 172.16.P.2:  bytes=32 time<10ms TTL=128
Reply from 172.16.P.2:  bytes=32 time<10ms TTL=128
```

(where P = pod number)

**Step 4** From your inside host, ping your bastion host with an ICMP packet size of 10000:

```
C:\> ping -l 10000 172.16.P.2
Pinging 172.16.P.2 with 10000 bytes of data:
Reply from 172.16.P.2:  bytes=10000 time<10ms TTL=128
Reply from 172.16.P.2:  bytes=10000 time<10ms TTL=128
Reply from 172.16.P.2:  bytes=10000 time<10ms TTL=128
Reply from 172.16.P.2:  bytes=10000 time<10ms TTL=128
```

(where P = pod number)

## Task 11—Use PDM to Configure Logging to a Syslog Server

Complete the following steps to configure the PIX Firewall to send messages to a Syslog server:

**Step 1** Select the **System Properties** tab.

**Step 2** Expand Logging from the Categories tree on the left of the panel. Logging Setup appears under Logging.

**Step 3** Select **Logging Setup**.

**Step 4** Select **Enable logging** in the Logging Setup group menu.

**Step 5** Click **Apply to PIX**. The Send Command window opens.

**Step 6** Click **Send**. You are returned to the System Properties tab.

**Step 7** Select **Syslog** under Logging from the Categories tree on the left of the panel. They Syslog group box appears.

**Step 8** Choose **Debugging** from the Level drop-down menu.

**Step 9** Select **Include Timestamp**.

**Step 10** Click **Add**. The Add Syslog Server window opens.

**Step 11** Verify that inside is selected in the Interface drop-down menu.

**Step 12** Enter **10.0.P.3** in the IP Address field.

(where P = pod number)

**Step 13** Click **OK**. You are returned to the System Properties tab.

**Step 14** Click **Apply to PIX**. The Send Command window opens.

**Step 15** Click **Send**.

## Task 12—Test Logging

Complete the following steps to verify that your PIX Firewall is logging to a Syslog server:

**Step 1** From your Windows command line, telnet to your perimeter router to generate traffic to be logged:

```
C:\> telnet 192.168.P.1
```

(where P = pod number)

**Step 2** Close the Telnet window that opens.

**Step 3** Open the file that contains the Syslog messages sent by the PIX Firewall. The contents should appear similar to the following:

```
<166> Jun 20 2001 16:19:14:  %PIX-6-302001: Built outbound TCP connection 38 for
faddr 192.168.1.1/23 gaddr 192.168.1.10/1815 laddr 10.0.1.3/1815
```

## Task 13—Configure Intrusion Detection

Complete the following steps to configure your PIX Firewall to detect ICMP packet attacks, drop the packets, and send an alarm to a Syslog server:

**Step 1** Expand Intrusion Detection from the Categories tree on the left of the panel. IDS Policy appears under Intrusion Detection.

**Step 2** Select **IDS Policy**. The IDS Policy group box opens on the right.

**Step 3** Click **Add**. The Add IDS Policy window opens.

**Step 4** Enter **ATTACKPOLICY** in the Policy Name text field.

**Step 5** Verify that Attack is selected in the Policy Type group box.

**Step 6** Select **Drop** and **Alarm** in the Action group box.

**Step 7** Click **OK**. You are returned to the System Properties tab.

**Step 8** Choose **ATTACKPOLICY** from the drop-down menu for the inside interface under Attack Policy.

**Step 9** Click **Apply to PIX**. The Send Command window opens.

**Step 10** Click **Send**.

## Task 14—Configure PDM to Monitor Intrusion Detection

Complete the following steps to configure monitoring of intrusion detection:

**Step 1** Select the **Monitoring** tab.

**Step 2** Expand Miscellaneous from the Categories tree on the left of the panel. IDS appears under Miscellaneous.

**Step 3** Select **IDS**.

**Step 4** Choose **ICMP Attacks** from the Available Graphs list.

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

**Step 6** Click **Graph It**. The New Graph window opens.

**Step 7** Verify that Real-time, data very 10 sec is selected in the View drop-down menu.

**Step 8** From your Windows command line, ping your bastion host with an ICMP packet size of 10000:

```
C:\> ping –l 10000 172.16.P.2

Pinging 172.16.P.2 with 10000 bytes of data:
Request timed out.
Request timed out.
Request timed out.
Request timed out.
```

(where P = pod number)

**Step 9** From your Windows command line, ping your bastion host with an ICMP packet size of 65000:

```
C:\> ping –l 65000 172.16.P.2

Pinging 172.16.P.2 with 65000 bytes of data:
Request timed out.
Request timed out.
Request timed out.
Request timed out.
```

(where P = pod number)

**Step 10** Observe the graph in the Graph tab.

**Step 11** Select the **Table** tab and observe the statistics in the table view.

**Step 12** Save the PIX Firewall configuration to Flash memory by clicking the **Save to Flash Needed** icon in the PDM toolbar. The Send Command window opens.

**Step 13** Click **Send**. The Save Successful window opens.

**Step 14** Click **OK**.

18

# The Cisco IOS Firewall Context-Based Access Control Configuration

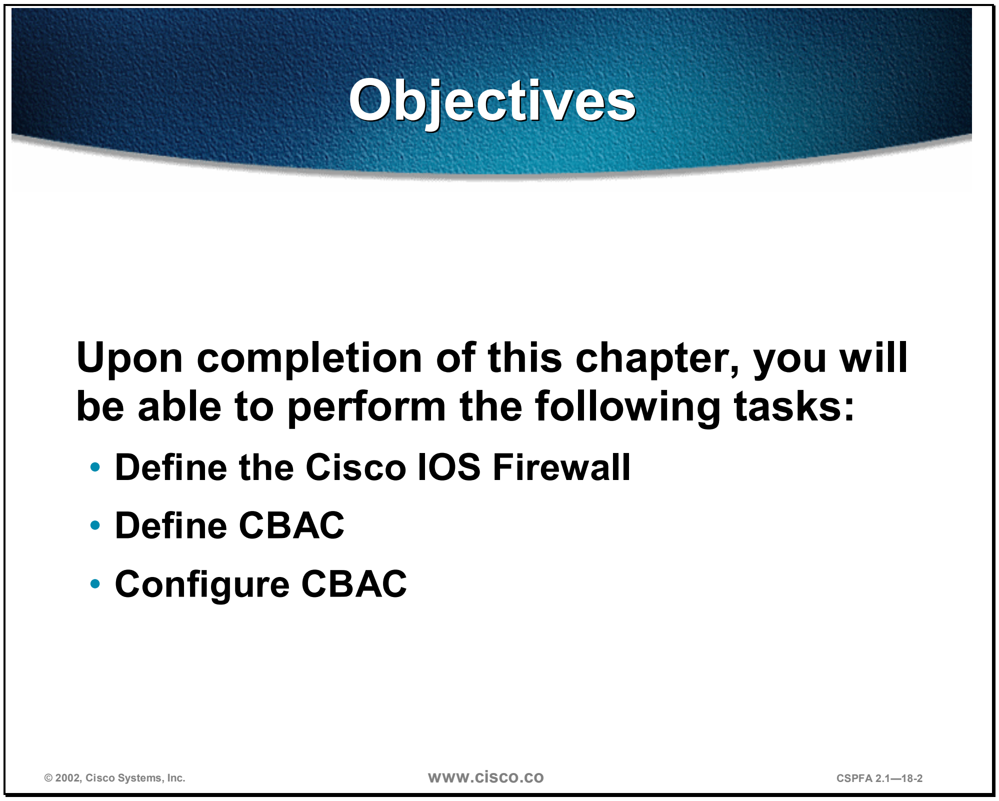
## Overview

This chapter includes the following topics:

- Objectives
- Introduction to the Cisco IOS Firewall
- Context-Based Access Control
- Global timeouts and thresholds
- Port-to-Application Mapping
- Define inspection rules
- Inspection rules and ACLs applied to router interfaces
- Test and verify
- Summary
- Lab exercise

# Objectives

This section lists the chapter’s objectives.



## Objectives

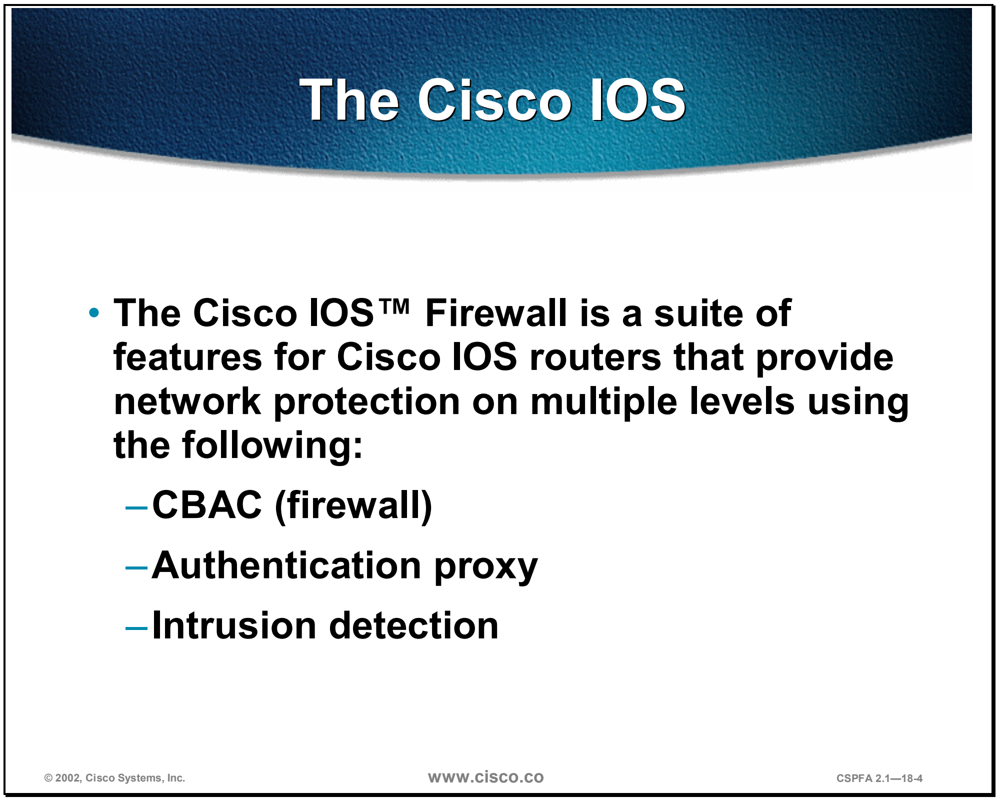
**Upon completion of this chapter, you will be able to perform the following tasks:**

- **Define the Cisco IOS Firewall**
- **Define CBAC**
- **Configure CBAC**

© 2002, Cisco Systems, Inc. www.cisco.co CSPFA 2.1—18-2

# Introduction to the Cisco IOS Firewall

This section introduces the features of The Cisco IOS™ Firewall.



## The Cisco IOS

- **The Cisco IOS™ Firewall is a suite of features for Cisco IOS routers that provide network protection on multiple levels using the following:**
  - **CBAC (firewall)**
  - **Authentication proxy**
  - **Intrusion detection**

© 2002, Cisco Systems, Inc. www.cisco.co CSPFA 2.1—18-4

The Cisco IOS Firewall is a security-specific option for Cisco IOS software. It integrates robust firewall functionality, authentication proxy, and intrusion detection for every network perimeter, and enriches existing Cisco IOS security capabilities. It adds greater depth and flexibility to existing Cisco IOS security solutions, such as authentication, encryption, and failover, by delivering state-of-the-art security features, such as stateful, application-based filtering; dynamic per-user authentication and authorization; defense against network attacks; Java blocking; and real-time alerts. When combined with Cisco IOS Internet Protocol Security (IPSec) software and other Cisco IOS software-based technologies, such as Layer 2 Tunneling Protocol (L2TP) tunneling and quality of service (QoS), the Cisco IOS Firewall provides a complete, integrated virtual private network (VPN) solution.

## Context-Based Access Control

The Cisco IOS Firewall Context-Based Access Control (CBAC) engine provides secure, per-application access control across network perimeters. CBAC enhances security for TCP and UDP applications that use well-known ports, such as FTP and e-mail traffic, by scrutinizing source and destination addresses. CBAC allows network administrators to implement firewall intelligence as part of an integrated, single-box solution.

For example, sessions with an extranet partner involving Internet applications, multimedia applications, or Oracle databases would no longer need to open a network doorway accessible via weaknesses in a partner's network. CBAC enables tightly secured networks to run today's basic application traffic, as well as advanced applications such as multimedia and videoconferencing, securely through a router.

## Authentication Proxy

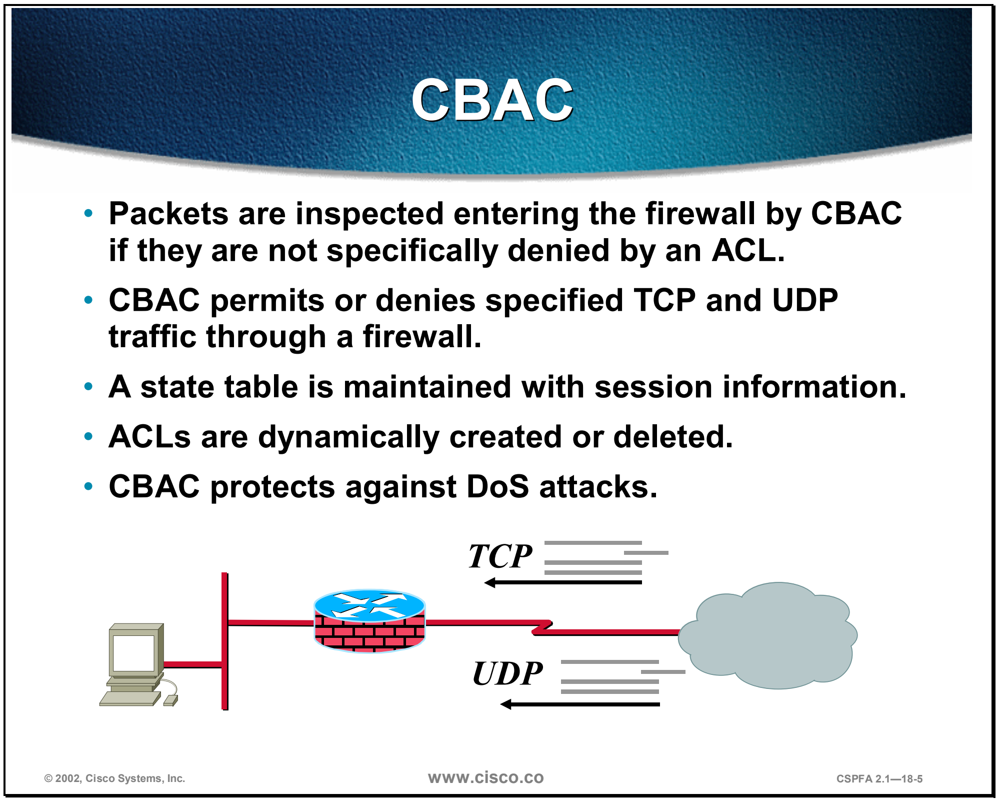
Network administrators can create specific security policies for each user with Cisco IOS Firewall LAN-based, dynamic, per-user authentication and authorization. Previously, user identity and related authorized access were determined by a user's fixed IP address, or a single security policy had to be applied to an entire user group or subnet. Now, per-user policy can be downloaded dynamically to the router from a Terminal Access Controller Access Control System Plus (TACACS+) or Remote Access Dial-In User Service (RADIUS) authentication server using Cisco IOS software authentication, authorization, and accounting (AAA) services.

Users can log into the network or access the Internet via HTTP, and their specific access profiles will automatically be downloaded. Appropriate dynamic individual access privileges are available as required, protecting the network against more general policies applied across multiple users. Authentication and authorization can be applied to the router interface in either direction to secure inbound or outbound extranet, intranet, and Internet usage.

## Intrusion Detection

Intrusion detection systems (IDS) provide a level of protection beyond the firewall by protecting the network from internal and external attacks and threats. Cisco IOS Firewall IDS technology enhances perimeter firewall protection by taking appropriate actions on packets and flows that violate the security policy or represent malicious network activity.

Cisco IOS Firewall intrusion detection capabilities are ideal for providing additional visibility at intranet, extranet, and branch-office Internet perimeters. Network administrators now enjoy more robust protection against attacks on the network, and can automatically respond to threats from internal or external hosts.



# CBAC

- Packets are inspected entering the firewall by CBAC if they are not specifically denied by an ACL.
- CBAC permits or denies specified TCP and UDP traffic through a firewall.
- A state table is maintained with session information.
- ACLs are dynamically created or deleted.
- CBAC protects against DoS attacks.

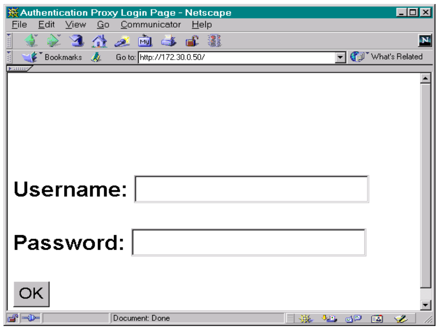
© 2002, Cisco Systems, Inc. www.cisco.co CSPFA 2.1—18-5

CBAC intelligently filters TCP and UDP packets based on application-layer protocol session information. It can inspect traffic for sessions that originate on any interface of the router. CBAC inspects traffic that travels through the firewall to discover and manage state information for TCP and UDP sessions. This state information is used to create temporary openings in the firewall's Access Control Lists (ACLs) to allow return traffic and additional data connections for permissible sessions.

Inspecting packets at the application layer and maintaining TCP and UDP session information provides CBAC with the ability to detect and prevent certain types of network attacks, such as SYN flooding. CBAC also inspects packet sequence numbers in TCP connections to see if they are within expected ranges—CBAC drops any suspicious packets. Additionally, CBAC can detect unusually high rates of new connections and issue alert messages. CBAC inspection can help protect against certain denial of service (DoS) attacks involving fragmented IP packets. Even though the firewall prevents an attacker from making actual connections to a given host, the attacker can disrupt services provided by that host. This is done by sending many non-initial IP fragments or by sending complete fragmented packets through a router with an ACL that filters the first fragment of a fragmented packet. These fragments can tie up resources on the target host as it tries to reassemble the incomplete packets.

# Authentication Proxy

- HTTP-based authentication
- Provides dynamic, per-user authentication and authorization via TACACS+ and RADIUS protocols

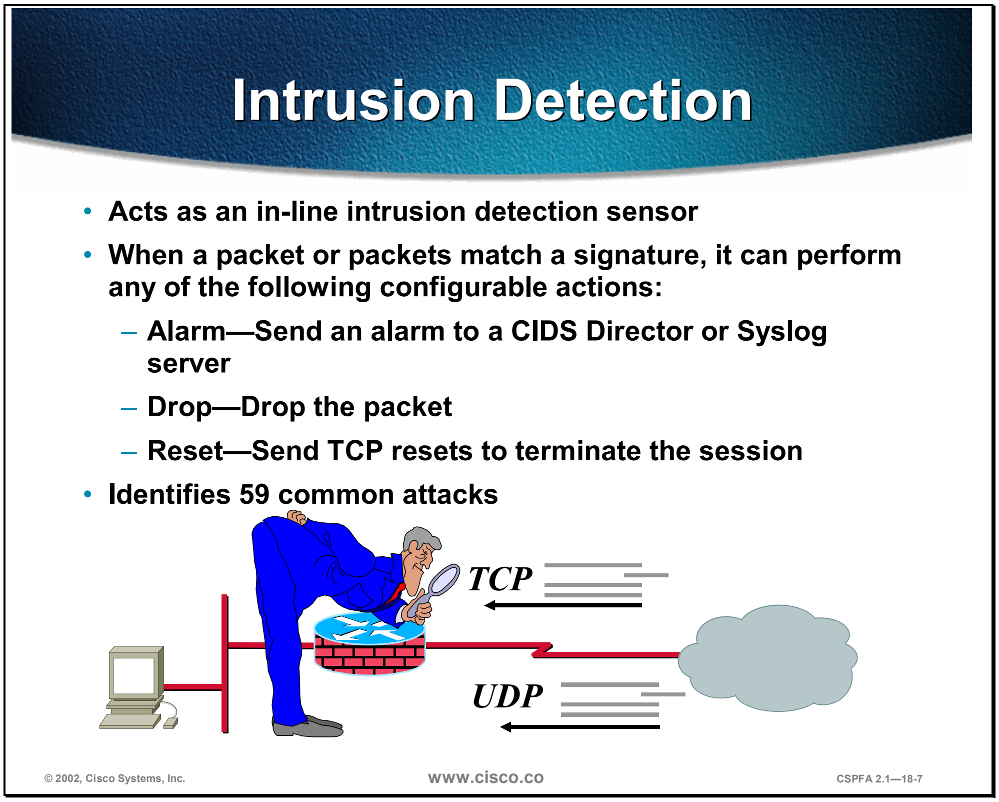

Authentication Proxy Login Page - Netscape
File Edit View Go Communicator Help
Bookmarks
Go to: http://172.30.0.50/
What's Related
Username:
Password:
OK
Document: Done

www.cisco.co CSPFA 2.1—18-6

The Cisco IOS Firewall authentication proxy feature enables network administrators to apply specific security policies on a per-user basis. Previously, user identity and related authorized access was associated with a user's IP address, or a single security policy had to be applied to an entire user group or subnet. Now, users can be identified and authorized on the basis of their per-user policy, and access privileges tailored on an individual basis are possible, as opposed to general policy applied across multiple users.

With the authentication proxy feature, users can log in to the network or access the Internet via HTTP, and their specific access profiles are automatically retrieved and applied from a Cisco Secure Access Control Server (CSACS), or other RADIUS or TACACS+ authentication server. The user profiles are active only when there is active traffic from the authenticated users.

The authentication proxy is compatible with other Cisco IOS security features such as Network Address Translation (NAT), IPSec encryption, and VPN client software.



## Intrusion Detection

- Acts as an in-line intrusion detection sensor
- When a packet or packets match a signature, it can perform any of the following configurable actions:
  - Alarm—Send an alarm to a CIDS Director or Syslog server
  - Drop—Drop the packet
  - Reset—Send TCP resets to terminate the session
- Identifies 59 common attacks

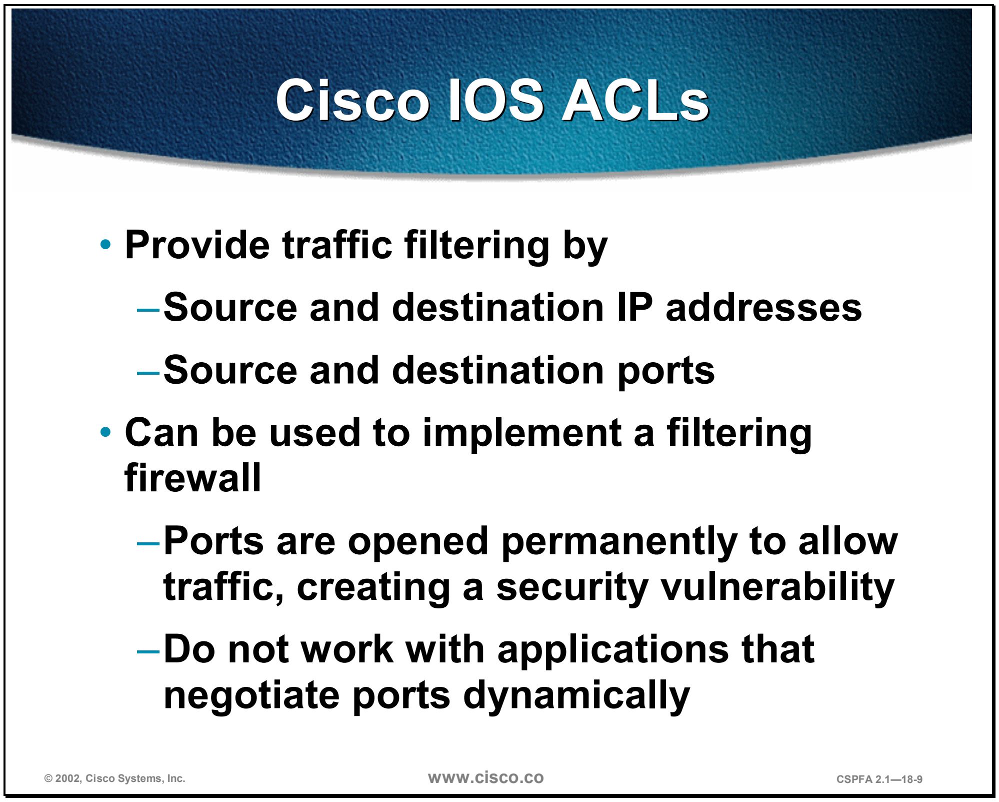
© 2002, Cisco Systems, Inc. www.cisco.co CSPFA 2.1—18-7

The Cisco IOS Firewall now offers intrusion detection technology for mid-range and high-end router platforms with firewall support. It is ideal for any network perimeter, and especially for locations in which a router is being deployed and additional security between network segments is required. It also can protect intranet and extranet connections where additional security is mandated, and branch-office sites connecting to the corporate office or Internet.

The Cisco IOS Firewall's intrusion detection system identifies 59 common attacks using signatures to detect patterns of misuse in network traffic. The intrusion detection signatures available in the new release of the Cisco IOS Firewall were chosen from a broad cross-section of intrusion detection signatures. The signatures represent severe breaches of security and the most common network attacks and information-gathering scans.

# Context-Based Access Control

This section describes the limitations of Cisco IOS ACLs and explains how Context-Based Access Control (CBAC) better protects users from attack. It also lists the protocols supported by CBAC and describes the added alert and audit trail features. Finally, the CBAC configuration tasks are listed.

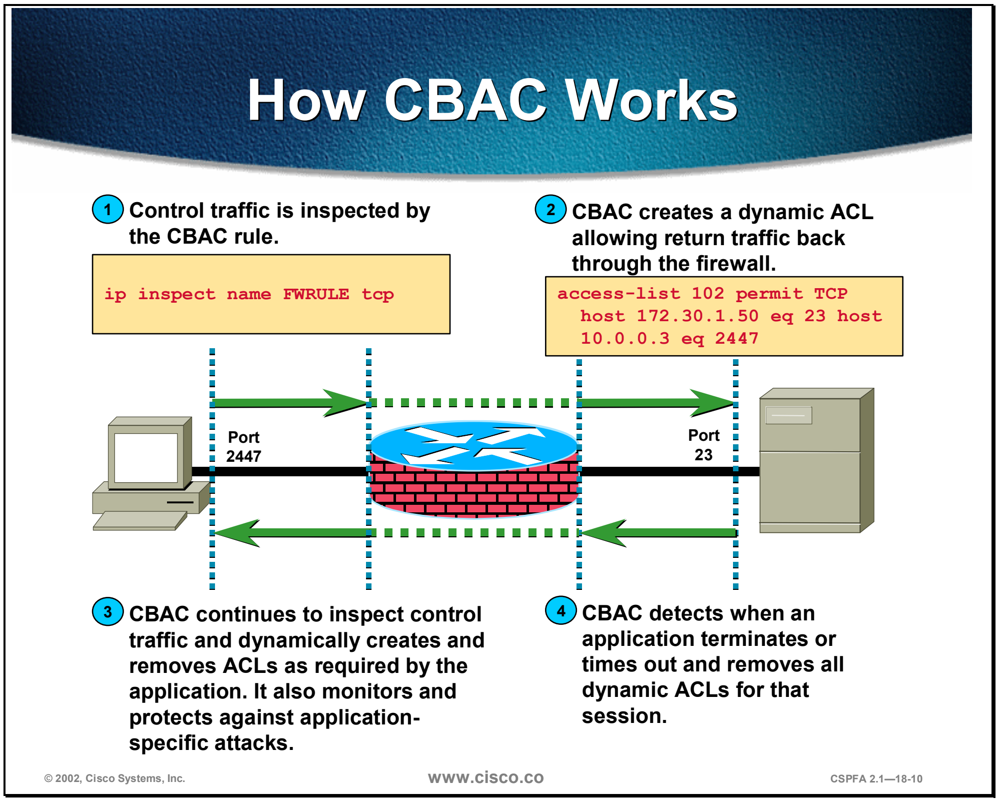


## Cisco IOS ACLs

- Provide traffic filtering by
  - Source and destination IP addresses
  - Source and destination ports
- Can be used to implement a filtering firewall
  - Ports are opened permanently to allow traffic, creating a security vulnerability
  - Do not work with applications that negotiate ports dynamically

© 2002, Cisco Systems, Inc. www.cisco.co CSPFA 2.1—18-9

Before delving into CBAC, some basic ACL concepts need to be covered briefly. An ACL provides packet filtering: it has an implied deny all at the end of the ACL and if the ACL is not configured, it permits all connections. Without CBAC, traffic filtering is limited to ACL implementations that examine packets at the network layer, or at most, the transport layer.



## How CBAC Works

1 Control traffic is inspected by the CBAC rule.

```
ip inspect name FWRULE tcp
```

2 CBAC creates a dynamic ACL allowing return traffic back through the firewall.

```
access-list 102 permit TCP
  host 172.30.1.50 eq 23 host
  10.0.0.3 eq 2447
```

3 CBAC continues to inspect control traffic and dynamically creates and removes ACLs as required by the application. It also monitors and protects against application-specific attacks.

4 CBAC detects when an application terminates or times out and removes all dynamic ACLs for that session.

www.cisco.co CSPFA 2.1—18-10

With CBAC, you specify which protocols you want to be inspected, and you specify an interface and interface direction (in or out) where the inspection originates. Only specified protocols will be inspected by CBAC. For these protocols, packets flowing through the firewall in any direction are inspected, as long as they flow through the interface where inspection is configured. Packets entering the firewall are inspected by CBAC only if they first pass the inbound ACL at the interface. If a packet is denied by the ACL, the packet is simply dropped and not inspected by CBAC.

CBAC inspects and monitors only the control channels of connections; the data channels are not inspected. For example, during FTP sessions both the control and data channels (which are created when a data file is transferred) are monitored for state changes, but only the control channel is inspected (that is, the CBAC software parses the FTP commands and responses).

CBAC inspection recognizes application-specific commands in the control channel, and detects and prevents certain application-level attacks. CBAC inspection tracks sequence numbers in all TCP packets, and drops those packets with sequence numbers that are not within expected ranges. CBAC inspection recognizes application-specific commands (such as illegal Simple Mail Transfer Protocol [SMTP] commands) in the control channel, and detects and prevents certain application-level attacks. When CBAC suspects an attack, the DoS feature can take several actions:

- Generate alert messages
- Protect system resources that could impede performance
- Block packets from suspected attackers

CBAC uses timeout and threshold values to manage session state information, helping to determine when to drop sessions that do not become fully established.

Setting timeout values for network sessions helps prevent DoS attacks by freeing system resources, dropping sessions after a specified amount of time. Setting threshold values for network sessions helps prevent DoS attacks by controlling the number of half-open sessions, which limits the amount of system resources applied to half-open sessions. When a session is dropped, CBAC sends a reset message to the devices at both endpoints (source and destination) of the session. When the system under DoS attack receives a reset command, it releases, or frees, processes and resources related to that incomplete session.

CBAC provides three thresholds against DoS attacks:

- The total number of half-open TCP or UDP sessions
- The number of half-open sessions based on time
- The number of half-open TCP-only sessions per host

If a threshold is exceeded, CBAC has two options:

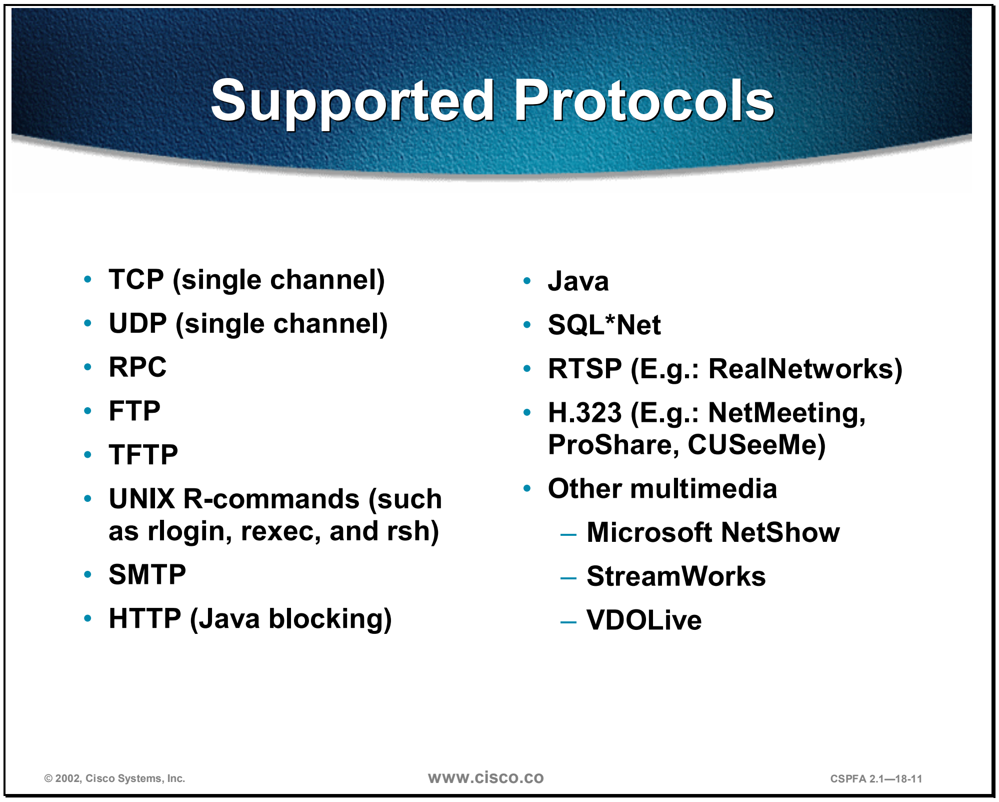
- Send a reset message to the endpoints of the oldest half-open session, making resources available to service newly arriving SYN packets.
- In the case of half-open TCP-only sessions, CBAC blocks all SYN packets temporarily for the duration configured by the threshold value. When the router blocks a SYN packet, the TCP three-way handshake is never initiated, which prevents the router from using memory and processing resources needed for valid connections.

DoS detection and prevention requires that you create a CBAC inspection rule and apply that rule on an interface. The inspection rule must include the protocols that you want to monitor against DoS attacks. For example, if you have TCP inspection enabled on the inspection rule, then CBAC can track all TCP connections to watch for DoS attacks. If the inspection rule includes FTP protocol inspection but not TCP inspection, CBAC tracks only FTP connections for DoS attacks.

A state table maintains session state information. Whenever a packet is inspected, a state table is updated to include information about the state of the packet's connection. Return traffic will only be permitted back through the firewall if the state table contains information indicating that the packet belongs to a permissible session. Inspection controls the traffic that belongs to a valid session and forwards the traffic it does not know. When return traffic is inspected, the state table information is updated as necessary.

UDP sessions are approximated. With UDP there are no actual sessions, so the software approximates sessions by examining the information in the packet and determining if the packet is similar to other UDP packets (for example, similar source or destination addresses and port numbers), and if the packet was detected soon after another, similar UDP packet. Soon means within the configurable UDP idle timeout period.

ACL entries are dynamically created and deleted. CBAC dynamically creates and deletes ACL entries at the firewall interfaces, according to the information maintained in the state tables. These ACL entries are applied to the interfaces to examine traffic flowing back into the internal network. These entries create temporary openings in the firewall to permit only traffic that is part of a permissible session. The temporary ACL entries are never saved to nonvolatile RAM (NVRAM.)



## Supported Protocols

- TCP (single channel)
- UDP (single channel)
- RPC
- FTP
- TFTP
- UNIX R-commands (such as rlogin, rexec, and rsh)
- SMTP
- HTTP (Java blocking)
- Java
- SQL*Net
- RTSP (E.g.: RealNetworks)
- H.323 (E.g.: NetMeeting, ProShare, CUSeeMe)
- Other multimedia
  - Microsoft NetShow
  - StreamWorks
  - VDOLive

© 2002, Cisco Systems, Inc. www.cisco.co CSPFA 2.1—18-11

You can configure CBAC to inspect the following types of sessions:

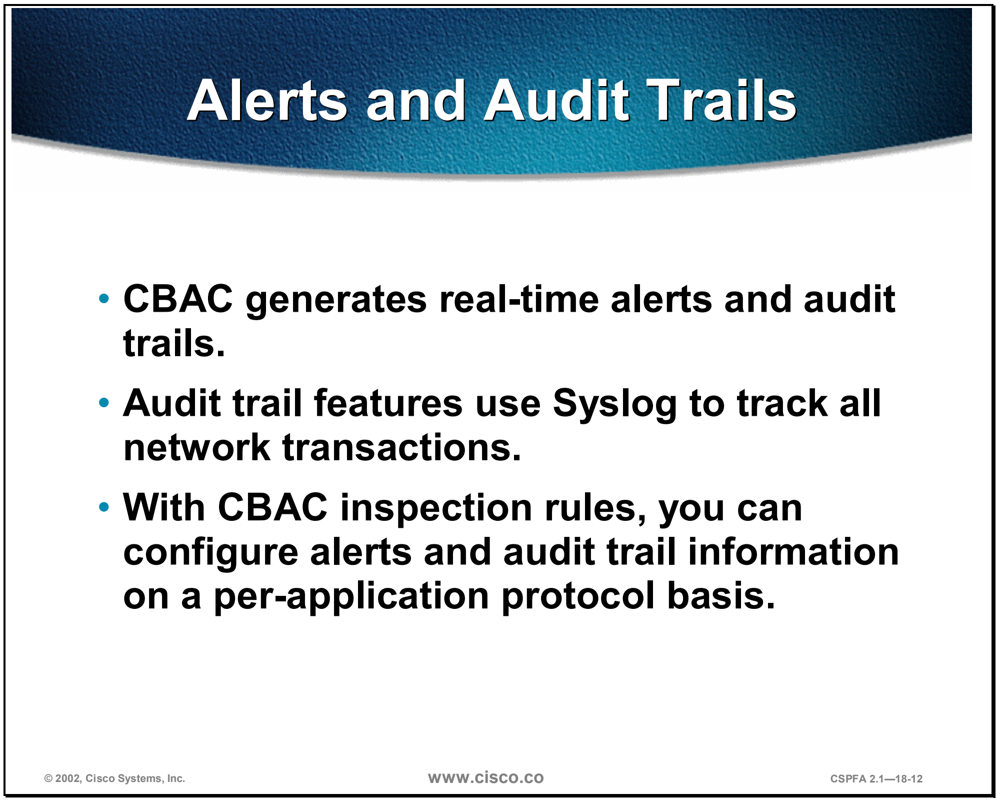
- All TCP sessions, regardless of the application-layer protocol (sometimes called single-channel or generic TCP inspection)
- All UDP sessions, regardless of the application-layer protocol (sometimes called single-channel or generic UDP inspection)

You can also configure CBAC to specifically inspect certain application-layer protocols. The following application-layer protocols can all be configured for CBAC:

- RPC (Sun RPC, not DCE RPC)
- Microsoft RPC
- FTP
- TFTP
- UNIX R-commands (such as rlogin, rexec, and rsh)
- SMTP
- HTTP (Java blocking)
- Java
- SQL*Net
- RTSP (for example: RealNetworks)
- H.323 (for example: NetMeeting, ProShare, CU-SeeMe [only the White Pine version])
- Microsoft NetShow

- StreamWorks
- VDOLive

When a protocol is configured for CBAC, that protocol traffic is inspected, state information is maintained, and, in general, packets are allowed back through the firewall only if they belong to a permissible session.



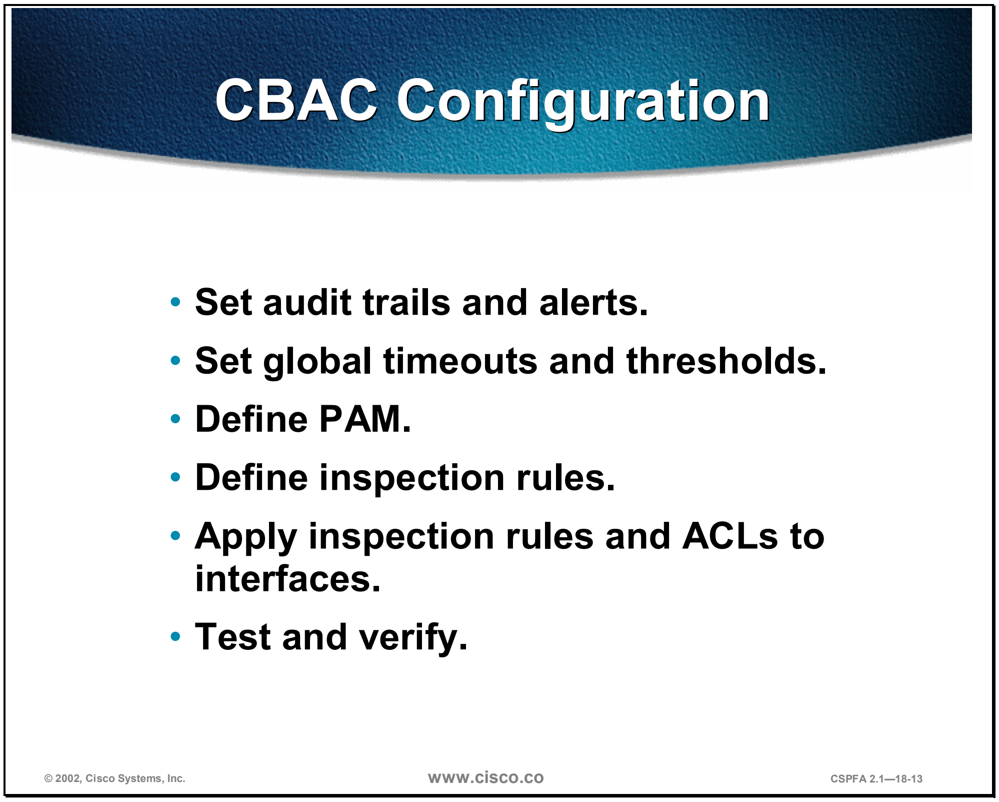
## Alerts and Audit Trails

- CBAC generates real-time alerts and audit trails.
- Audit trail features use Syslog to track all network transactions.
- With CBAC inspection rules, you can configure alerts and audit trail information on a per-application protocol basis.

© 2002, Cisco Systems, Inc. www.cisco.co CSPFA 2.1—18-12

CBAC also generates real-time alerts and audit trails based on events tracked by the firewall. Enhanced audit trail features use Syslog to track all network transactions, recording time stamps, source host, destination host, ports used, and the total number of transmitted bytes, for advanced, session-based reporting.

Real-time alerts send Syslog error messages to central management consoles upon detecting suspicious activity. Using CBAC inspection rules, you can configure alerts and audit trail information on a per-application protocol basis. For example, if you want to generate audit trail information for HTTP traffic, you can specify that in the CBAC rule covering HTTP inspection.



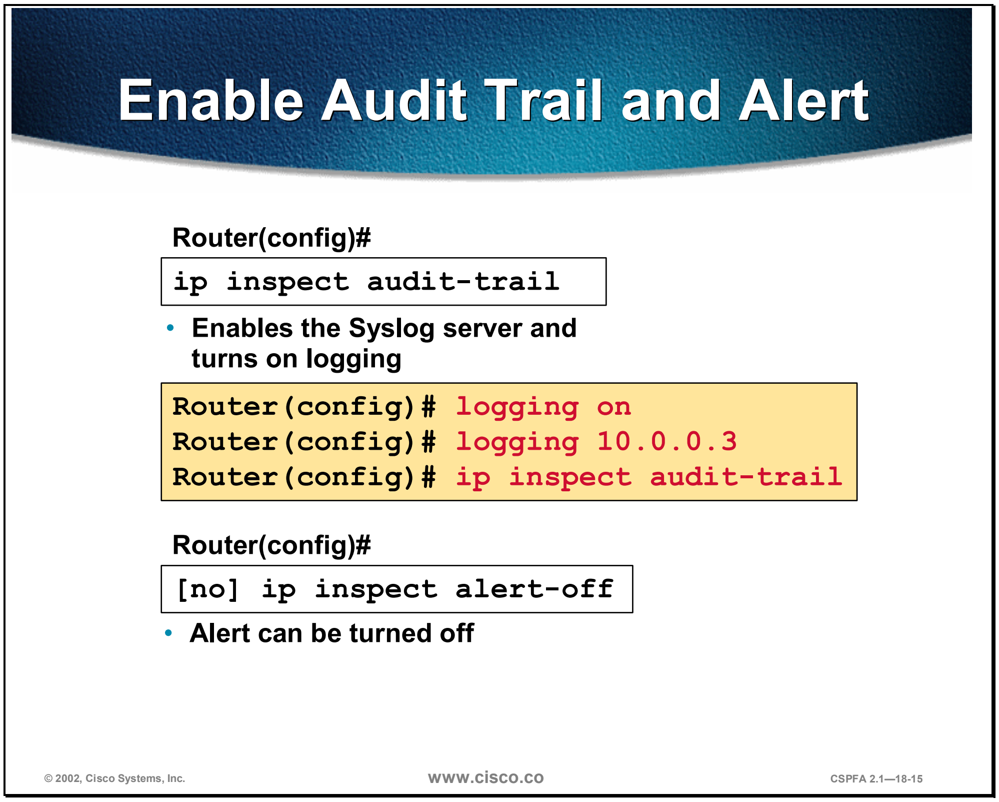
## CBAC Configuration

- Set audit trails and alerts.
- Set global timeouts and thresholds.
- Define PAM.
- Define inspection rules.
- Apply inspection rules and ACLs to interfaces.
- Test and verify.

© 2002, Cisco Systems, Inc. www.cisco.co CSPFA 2.1—18-13

The following are the tasks used to configure CBAC:

- Set audit trails and alerts.
- Set global timeouts and thresholds.
- Define Port-to-Application Mapping (PAM).
- Define inspection rules.
- Apply inspection rules and ACLs to interfaces.
- Test and verify.



## Enable Audit Trail and Alert

Router(config)#

```
ip inspect audit-trail
```

- Enables the Syslog server and turns on logging

```
Router(config)# logging on
Router(config)# logging 10.0.0.3
Router(config)# ip inspect audit-trail
```

Router(config)#

```
[no] ip inspect alert-off
```

- Alert can be turned off

© 2002, Cisco Systems, Inc. www.cisco.co CSPFA 2.1—18-15

Turn on logging and audit trail to provide a record of network access through the firewall, including illegitimate access attempts, and inbound and outbound services.

Use the **ip inspect audit-trail** and **ip inspect alert-off** commands to enable audit trail and alert, respectively.

The syntax for the **ip inspect audit-trail** commands is as follows:

**ip inspect audit-trail**

**no ip inspect audit-trail**

The syntax for the **ip inspect alert-off** commands is as follows:

**ip inspect alert-off**

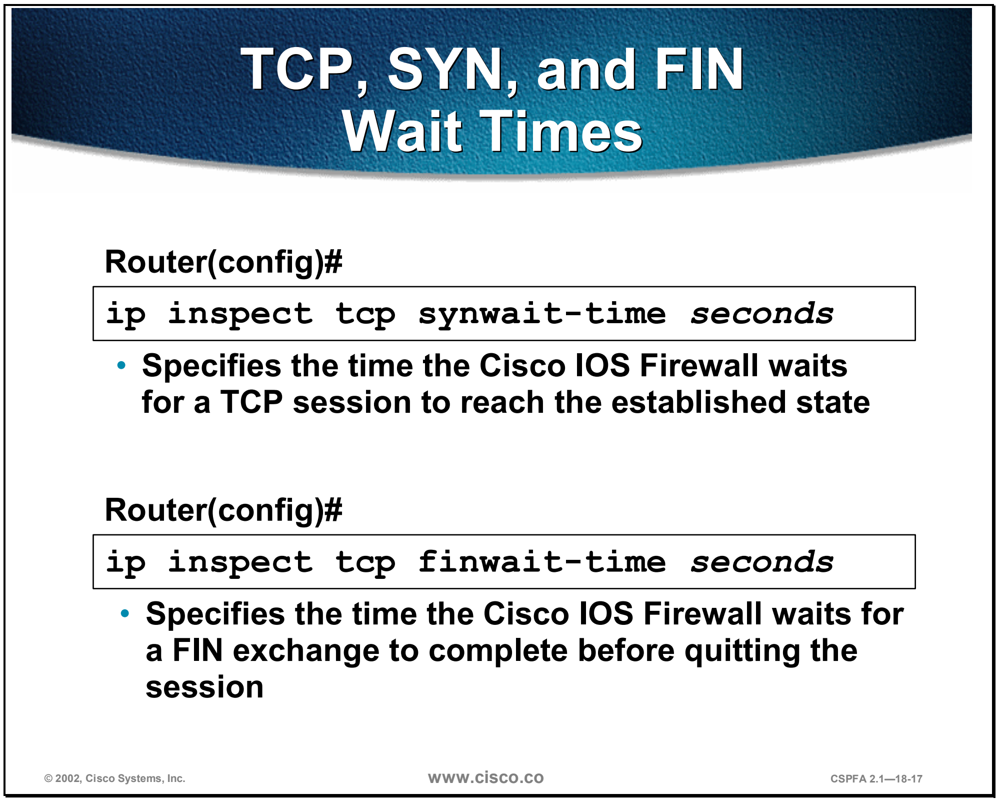
**no ip inspect alert-off**

No other arguments or keywords are used with either command.

# Global Timeouts and Thresholds

This section discusses how to configure the following global timeouts and thresholds:

- TCP, SYN, and FIN wait times
- TCP, UDP, and Domain Name System (DNS) idle times
- TCP flood DoS protection



## TCP, SYN, and FIN Wait Times

Router(config)#

```
ip inspect tcp synwait-time seconds
```

- Specifies the time the Cisco IOS Firewall waits for a TCP session to reach the established state

Router(config)#

```
ip inspect tcp finwait-time seconds
```

- Specifies the time the Cisco IOS Firewall waits for a FIN exchange to complete before quitting the session

www.cisco.co

CSPFA 2.1—18-17

CBAC uses timeouts and thresholds to determine how long to manage state information for a session, and to determine when to drop sessions that do not become fully established. These timeouts and thresholds apply globally to all sessions.

You can use the default timeout and threshold values, or you can change to values more suitable to your security requirements. You should make any changes to the timeout and threshold values before you continue configuring CBAC.

To define how long the software will wait for a TCP session to reach the established state before dropping the session, use the **ip inspect tcp synwait-time** global configuration command. Use the **no** form of this command to reset the timeout to the default.

The syntax of the **ip inspect tcp synwait-time** command is as follows:

**ip inspect tcp synwait-time** *seconds*

**no ip inspect tcp synwait-time**

|  |  |
|---|---|
| ***seconds*** | Specifies how long the software will wait for a TCP session to reach the established state before dropping the session (the default is 30 seconds). |

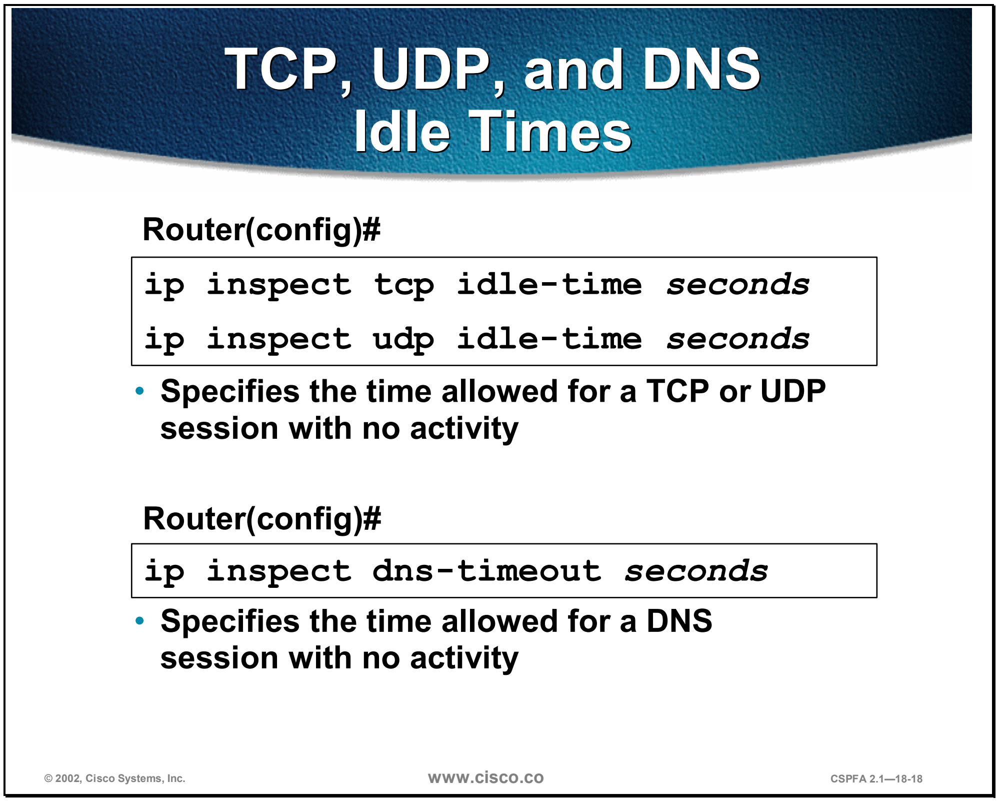
To define how long a TCP session will still be managed after the firewall detects a FIN exchange, use the **ip inspect tcp finwait-time** global configuration command. Use the **no** form of this command to reset the timeout to default.

The syntax of the **ip inspect tcp finwait-time** command is as follows:

**ip inspect tcp finwait-time** ***seconds***

**no ip inspect tcp finwait-time**

|  |  |
|---|---|
| ***seconds*** | Specifies how long a TCP session will be managed after the firewall detects a FIN exchange (the default is 5 seconds). |



## TCP, UDP, and DNS Idle Times

Router(config)#

```
ip inspect tcp idle-time seconds
ip inspect udp idle-time seconds
```

- Specifies the time allowed for a TCP or UDP session with no activity

Router(config)#

```
ip inspect dns-timeout seconds
```

- Specifies the time allowed for a DNS session with no activity

© 2002, Cisco Systems, Inc. www.cisco.co CSPFA 2.1—18-18

To specify the TCP idle timeout (the length of time a TCP session will still be managed after no activity), use the **ip inspect tcp idle-time** global configuration command. Use the **no** form of this command to reset the timeout to default.

To specify the UDP idle timeout (the length of time a UDP session will still be managed after no activity), use the **ip inspect udp idle-time** global configuration command. Use the **no** form of this command to reset the timeout to default.

The syntax for the **ip inspect {tcp | udp} idle-time** commands is as follows:

**ip inspect {tcp | udp} idle-time** ***seconds***

**no ip inspect {tcp | udp} idle-time**

| | |
|---|---|
| ***seconds*** | Specifies the length of time a TDP or a UDP session will still be managed after no activity. For TCP sessions, the default is 3600 seconds (1 hour). For UDP sessions, the default is 30 seconds. |

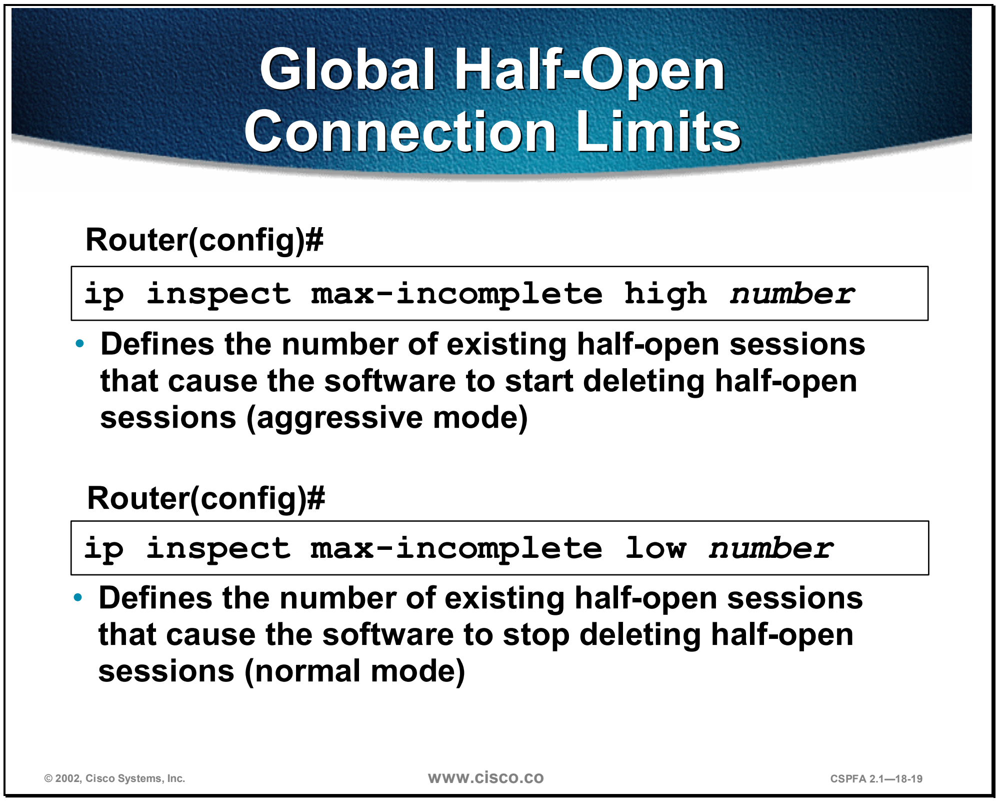
To specify the DNS idle timeout (the length of time a DNS name lookup session will still be managed after no activity), use the **ip inspect dns-timeout** global configuration command. Use the **no** form of this command to reset the timeout to the default.

The syntax for the **ip inspect dns-timeout** command is as follows:

**ip inspect dns-timeout** ***seconds***

**no ip inspect dns-timeout**

| | |
|---|---|
| ***seconds*** | Specifies the length of time a DNS name lookup session will still be managed after no activity (the default is 5 seconds). |



## Global Half-Open Connection Limits

Router(config)#

```
ip inspect max-incomplete high number
```

- **Defines the number of existing half-open sessions that cause the software to start deleting half-open sessions (aggressive mode)**

Router(config)#

```
ip inspect max-incomplete low number
```

- **Defines the number of existing half-open sessions that cause the software to stop deleting half-open sessions (normal mode)**

© 2002, Cisco Systems, Inc. www.cisco.co CSPFA 2.1—18-19

An unusually high number of half-open sessions (either absolute or measured as the arrival rate) could indicate that a DoS attack is occurring. For TCP, half-open means that the session has not reached the established state—the TCP three-way handshake has not yet been completed. For UDP, half-open means that the firewall has detected no return traffic.

CBAC measures both the total number of existing half-open sessions and the rate of session establishment attempts. Both TCP and UDP half-open sessions are counted in the total number and rate measurements. Measurements are made once a minute.

When the number of existing half-open sessions rises above a threshold (the **max-incomplete high *number***), CBAC will go in to "aggressive mode" and delete half-open sessions as required to accommodate new connection requests. The software continues to delete half-open requests as necessary, until the number of existing half-open sessions drops below another threshold (the **max-incomplete low *number***).

To define the number of existing half-open sessions that will cause the software to start deleting half-open sessions, use the **ip inspect max-incomplete high** command in global configuration mode. Use the **no** form of this command to reset the threshold to default.

The syntax for the **ip inspect max-incomplete high** command is as follows:

**ip inspect max-incomplete high *number***

**no ip inspect max-incomplete high *number***

| | |
|---|---|
| **high *number*** | Specifies the number of existing half-open sessions that will cause the software to start deleting half-open sessions (the default is 500 half-open sessions). |

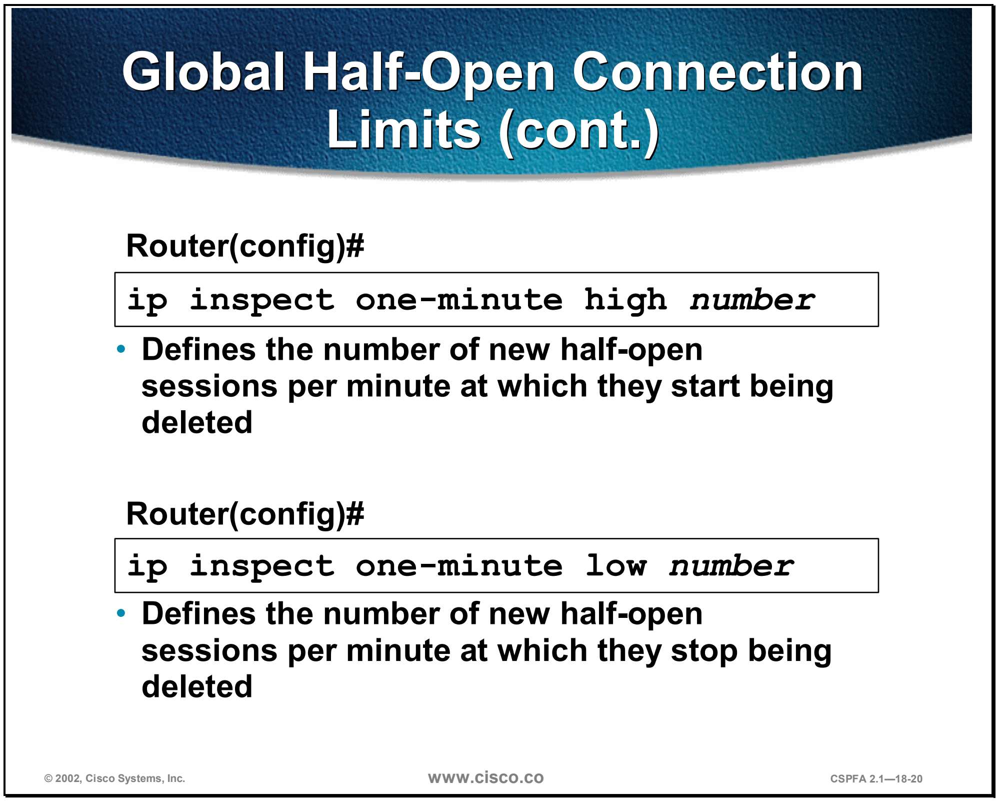
To define the number of existing half-open sessions that will cause the software to stop deleting half-open sessions, use the **ip inspect max-incomplete low** command in global configuration mode. Use the **no** form of this command to reset the threshold to default.

The syntax for the **ip inspect max-incomplete low** command is as follows:

**ip inspect max-incomplete low** ***number***

**no ip inspect max-incomplete low** ***number***

| | |
|---|---|
| **low** ***number*** | Specifies the number of existing half-open sessions that will cause the software to stop deleting half-open sessions (the default is 400 half-open sessions). |



## Global Half-Open Connection Limits (cont.)

Router(config)#

```
ip inspect one-minute high number
```

- Defines the number of new half-open sessions per minute at which they start being deleted

Router(config)#

```
ip inspect one-minute low number
```

- Defines the number of new half-open sessions per minute at which they stop being deleted

© 2002, Cisco Systems, Inc. www.cisco.co CSPFA 2.1—18-20

When the rate of new connection attempts rises above a threshold (the **one-minute high *number***), the software will delete half-open sessions as required to accommodate new connection attempts. The software continues to delete half-open sessions as necessary, until the rate of new connection attempts drops below another threshold (the **one-minute low *number***). The rate thresholds are measured as the number of new session connection attempts detected in the last one-minute sample period. The firewall router reviews the one-minute rate on an ongoing basis, meaning that the router reviews the rate more frequently than one minute and does not keep deleting half-open sessions for one-minute after a DoS attack has stopped—it will be less time.

To define the rate of new unestablished sessions that will cause the software to start deleting half-open sessions, use the **ip inspect one-minute high** command in global configuration mode. Use the **no** form of this command to reset the threshold to default.

The syntax for the **ip inspect one-minute high** command is as follows:

**ip inspect one-minute high *number***

**no ip inspect one-minute high**

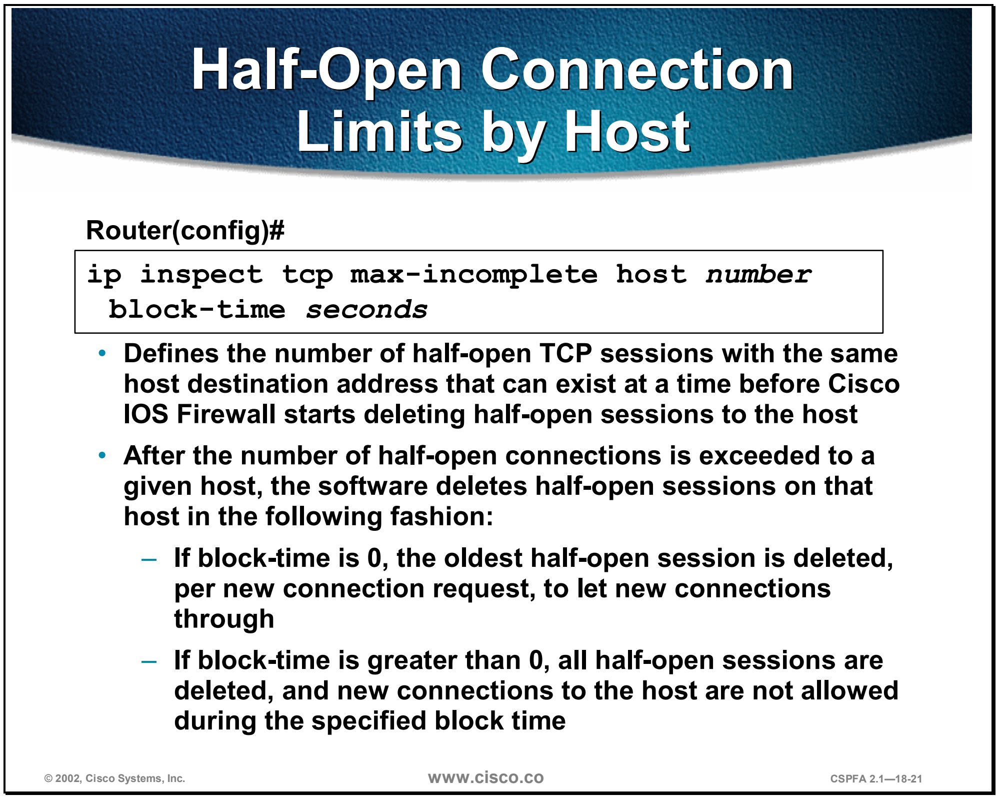
| | |
|---|---|
| **high *number*** | Specifies the rate of new unestablished TCP sessions that will cause the software to start deleting half-open sessions (the default is 500 half-open sessions). |

To define the rate of new unestablished TCP sessions that will cause the software to stop deleting half-open sessions, use the **ip inspect one-minute low** command in global configuration mode. Use the **no** form of this command to reset the threshold to the default.

The syntax for the **ip inspect one-minute low** command is as follows:

```
ip inspect one-minute low number

no ip inspect one-minute low
```

| | |
|---|---|
| **low** ***number*** | Specifies the number of existing half-open sessions that will cause the software to stop deleting half-open sessions (the default is 400 half-open sessions). |



## Half-Open Connection Limits by Host

Router(config)#

```
ip inspect tcp max-incomplete host number
 block-time seconds
```

- Defines the number of half-open TCP sessions with the same host destination address that can exist at a time before Cisco IOS Firewall starts deleting half-open sessions to the host
- After the number of half-open connections is exceeded to a given host, the software deletes half-open sessions on that host in the following fashion:
  - If block-time is 0, the oldest half-open session is deleted, per new connection request, to let new connections through
  - If block-time is greater than 0, all half-open sessions are deleted, and new connections to the host are not allowed during the specified block time

© 2002, Cisco Systems, Inc. www.cisco.co CSPFA 2.1—18-21

An unusually high number of half-open sessions with the same destination host address could indicate that a DoS attack is being launched against the host. Whenever the number of half-open sessions with the same destination host address rises above a threshold (the **max-incomplete host *number***), the software will delete half-open sessions according to one of the following methods:

- If the **block-time *seconds*** timeout is 0 (the default), the software deletes the oldest existing half-open session for the host for every new connection request to the host. This ensures that the number of half-open sessions to a given host will never exceed the threshold.
- If the **block-time *seconds*** timeout is greater than 0, the software deletes all existing half-open sessions for the host, and then blocks all new connection requests to the host. The software will continue to block all new connection requests until the block time expires.

The software also sends Syslog messages whenever the **max-incomplete host *number*** is exceeded, and when blocking of connection initiations to a host starts or ends.

The global values specified for the threshold and blocking time apply to all TCP connections inspected by CBAC.

Use the **ip inspect tcp max-incomplete host** global configuration command to specify threshold and blocking time values for TCP host-specific DoS detection and prevention. Use the **no** form of this command to reset the threshold and blocking time to the default values.

The syntax for the **ip inspect tcp max-incomplete host** command is as follows:

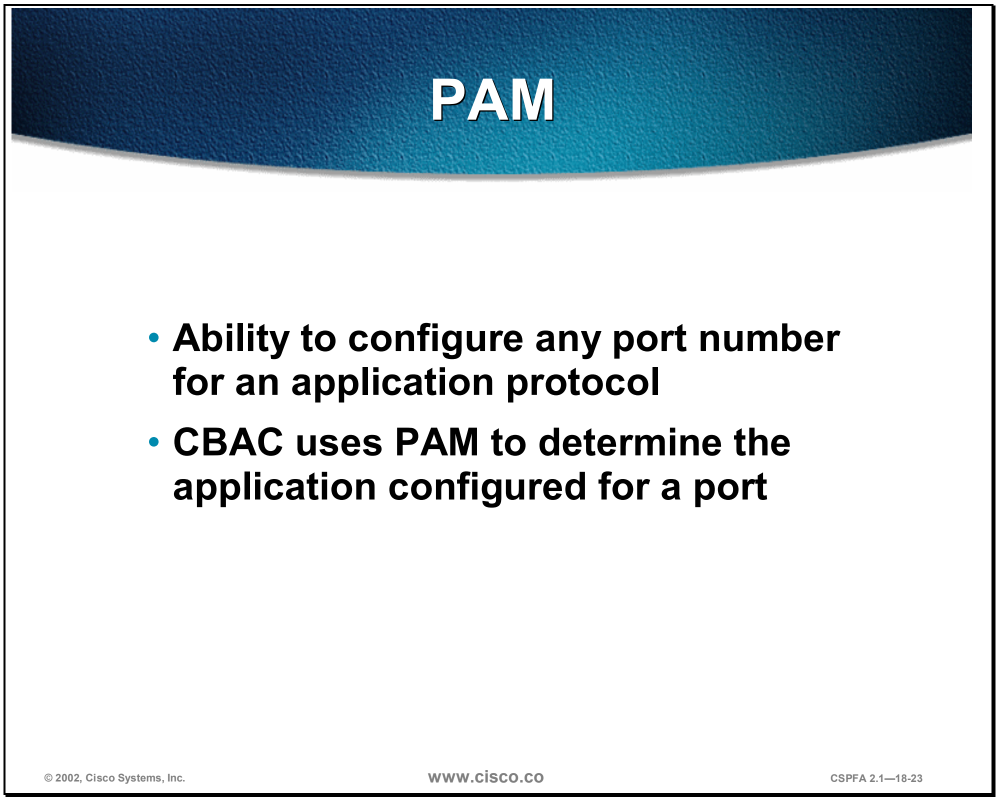
**ip inspect tcp max-incomplete host *number* block-time *seconds***

**no ip inspect tcp max-incomplete host**

| | |
|---|---|
| **host** ***number*** | Specifies how many half-open TCP sessions with the same host destination address can exist at a time before the software starts deleting half-open sessions to the host. Use a number from 1 to 250 (the default is 50 half-open sessions). |
| **block-time** ***seconds*** | Specifies how long the software will continue to delete new connection requests to the host (the default is 0 seconds). |

# Port-to-Application Mapping

This section discusses the configuration of port numbers for application protocols.



## PAM

- Ability to configure any port number for an application protocol
- CBAC uses PAM to determine the application configured for a port

© 2002, Cisco Systems, Inc. www.cisco.co CSPFA 2.1—18-23

Port-to-Application Mapping (PAM) enables you to customize TCP or UDP port numbers for network services or applications. PAM uses this information to support network environments that run services using ports that are different from the registered or well-known ports associated with an application.

Using the port information, PAM establishes a table of default port-to-application mapping information at the firewall. The information in the PAM table enables CBAC supported services to run on nonstandard ports. Previously, CBAC was limited to inspecting traffic using only the well-known or registered ports associated with an application. Now, PAM allows network administrators to customize network access control for specific applications and services.

PAM also supports host or subnet-specific port mapping, which enables you to apply PAM to a single host or subnet using standard ACLs. Host- or subnet-specific port mapping is done using standard ACLs.

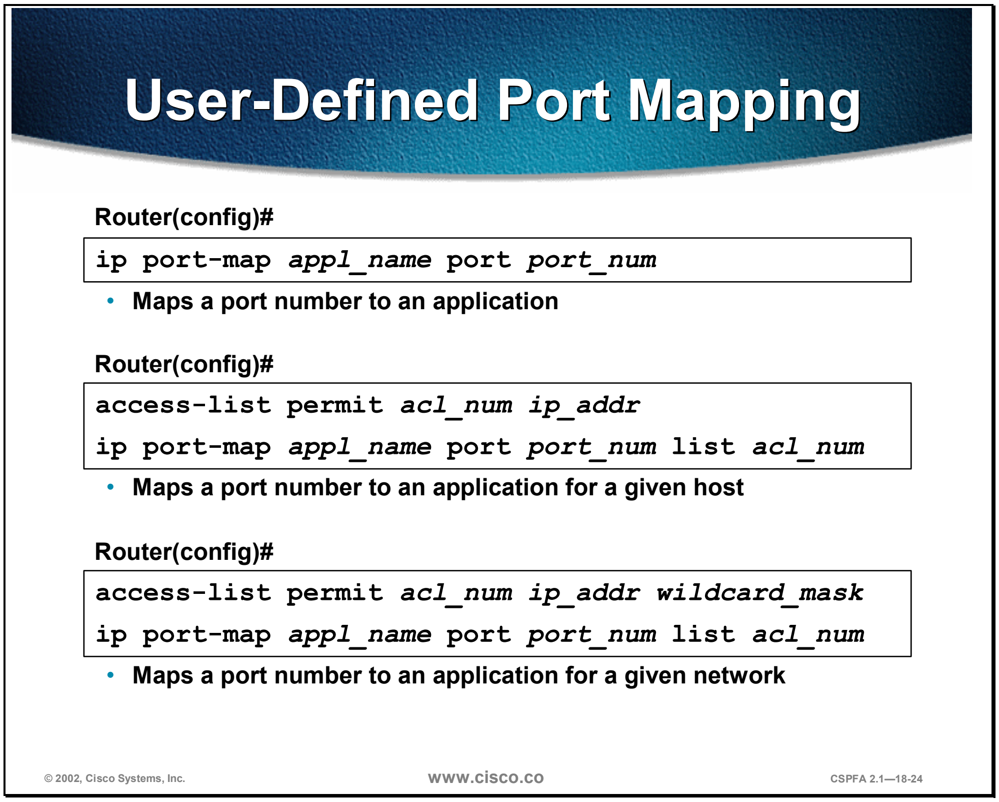
## System-Defined Port Mapping

PAM creates a table, or database, of system-defined mapping entries using the well-known or registered port mapping information set up during the system startup. The system-defined entries comprise all the services supported by CBAC, which requires the system-defined mapping information to function properly.

**Note** The system-defined mapping information cannot be deleted or changed; that is, you cannot map HTTP services to port 21 (FTP) or FTP services to port 80 (HTTP).

The following are the default system-defined services and applications found in the PAM table.

| **Application** | **Port** |
|---|---|
| cuseeme | 7648 |
| exec | 512 |
| ftp | 21 |
| http | 80 |
| h323 | 1720 |
| login | 513 |
| mgcp | 2427 |
| msrpc | 135 |
| netshow | 1755 |
| realmedia | 7070 |
| rtsp | 554 |
| rtsp | 8554 |
| shell | 514 |
| sip | 5060 |
| smtp | 25 |
| sql-net | 1521 |
| streamworks | 1558 |
| sunrpc | 111 |
| telnet | 23 |
| tftp | 69 |
| vdolive | 7000 |



# User-Defined Port Mapping

Router(config)#

```
ip port-map appl_name port port_num
```

- Maps a port number to an application

Router(config)#

```
access-list permit acl_num ip_addr
ip port-map appl_name port port_num list acl_num
```

- Maps a port number to an application for a given host

Router(config)#

```
access-list permit acl_num ip_addr wildcard_mask
ip port-map appl_name port port_num list acl_num
```

- Maps a port number to an application for a given network

© 2002, Cisco Systems, Inc. www.cisco.co CSPFA 2.1—18-24

Network services or applications that use nonstandard ports require user-defined entries in the PAM table. For example, your network might run HTTP services on the nonstandard port 8000 instead of on the system-defined default port 80. In this case, you can use PAM to map port 8000 with HTTP services. If HTTP services run on other ports, use PAM to create additional port mapping entries. After you define a port mapping entry, you can overwrite that entry at a later time by simply mapping that specific port with a different application.

**Note** If you try to map an application to a system-defined port, a message appears warning you of a mapping conflict.

User-defined port mapping information can also specify a range of ports for an application by establishing a separate entry in the PAM table for each port number in the range.

User-defined entries are saved with the default mapping information when you save the router configuration.

To establish PAM, use the **ip port-map** configuration command. Use the **no** form of this command to delete user-defined PAM entries.

The syntax for the **ip port-map** command is as follows:

**ip port-map** ***appl_name*** **port** ***port_num*** **[list** ***acl_num*****]**

| | |
|---|---|
| ***appl_name*** | Specifies the name of the application with which to apply the port mapping. Use one of the following application names: cuseeme, dns, exec, finger, ftp, gopher, http, h323, imap, kerberos, ldap, login, lotusnote, mgcp, msrpc, ms-sql, netshow, nfs, nntp, pop2, pop3, realmedia, rtsp, sap, shell, sip, smtp, snmp, sql-net, streamworks, sunrpc, sybase-sql, tacacs, telnet, tftp, or vdolive |

|  |  |
|---|---|
| **port** ***port_num*** | Identifies a port number in the range 1 to 65535. |
| **list** ***acl_num*** | Identifies the standard ACL number used with PAM for host- or network-specific port mapping. |

User-defined entries in the mapping table can include host- or network-specific mapping information, which establishes port mapping information for specific hosts or subnets. In some environments, it might be necessary to override the default port mapping information for a specific host or subnet.

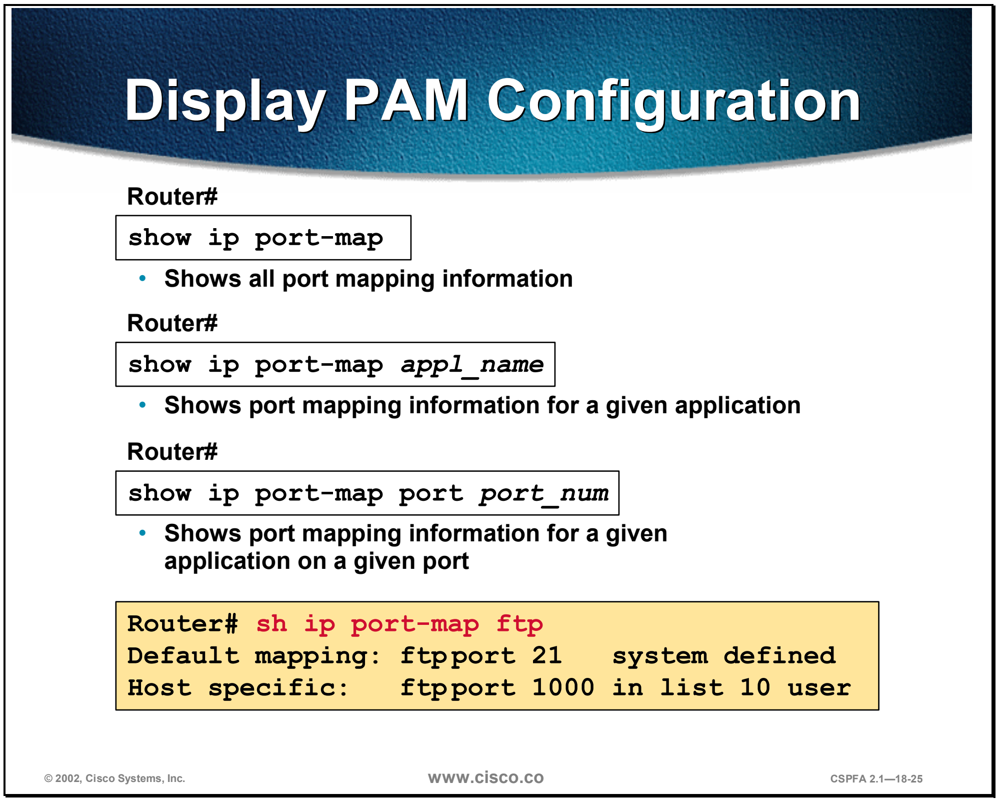
With host-specific port mapping, you can use the same port number for different services on different hosts. This means that you can map port 8000 with HTTP services for one host, while mapping port 8000 with Telnet services for another host.

Host-specific port mapping also enables you to apply PAM to a specific subnet when that subnet runs a service that uses a port number that is different from the port number defined in the default mapping information. For example, hosts on subnet 192.168.0.0 might run HTTP services on nonstandard port 8000, while other traffic through the firewall uses the default port 80 for HTTP services.

Host- or network-specific port mapping enables you to override a system-defined entry in the PAM table. For example, if CBAC finds an entry in the PAM table that maps port 25 (the system-defined port for SMTP) with HTTP for a specific host, CBAC identifies port 25 as HTTP protocol traffic on that host.

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

Use the **list** option for the **ip port-map** command to specify an ACL for a host or subnet that uses PAM.



# Display PAM Configuration

Router#

```
show ip port-map
```

- Shows all port mapping information

Router#

```
show ip port-map appl_name
```

- Shows port mapping information for a given application

Router#

```
show ip port-map port port_num
```

- Shows port mapping information for a given application on a given port

```
Router# sh ip port-map ftp
Default mapping: ftp port 21    system defined
Host specific:   ftp port 1000 in list 10 user
```

© 2002, Cisco Systems, Inc. www.cisco.co CSPFA 2.1—18-25

To display the PAM information, use the **show ip port-map** privileged EXEC command.

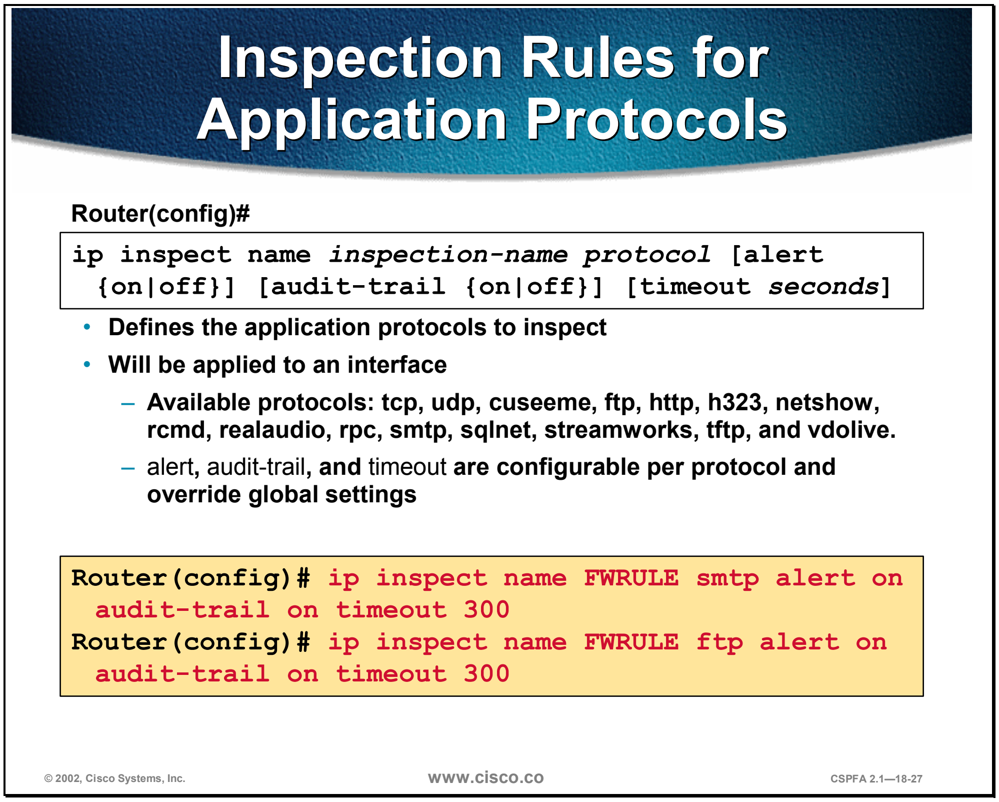
The syntax for the **show ip port-map** command is as follows:

**show ip port-map [*appl_name* | port *port_num*]**

| | |
|---|---|
| ***appl_name*** | Specifies the application for which to display information. |
| **port *port_num*** | Specifies the alternative port number that maps to the application for which to display information. |

# Define Inspection Rules

This section discusses how to configure the rules used to define the application protocols for inspection.



## Inspection Rules for Application Protocols

Router(config)#

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

- Defines the application protocols to inspect
- Will be applied to an interface
  - Available protocols: tcp, udp, cuseeme, ftp, http, h323, netshow, rcmd, realaudio, rpc, smtp, sqlnet, streamworks, tftp, and vdolive.
  - alert, audit-trail, **and** timeout **are configurable per protocol and override global settings**

```
Router(config)# ip inspect name FWRULE smtp alert on
  audit-trail on timeout 300
Router(config)# ip inspect name FWRULE ftp alert on
  audit-trail on timeout 300
```

© 2002, Cisco Systems, Inc. www.cisco.co CSPFA 2.1—18-27

Inspection rules must be defined to specify what IP traffic (which application-layer protocols) will be inspected by CBAC at an interface. Normally, you define only one inspection rule. The only exception might occur if you want to enable CBAC in two directions at a single firewall interface. In this case you must configure two rules, one for each direction.

An inspection rule should specify each desired application-layer protocol, as well as generic TCP or generic UDP, if desired. The inspection rule consists of a series of statements, each listing a protocol and specifying the same inspection rule name.

Inspection rules include options for controlling alert and audit trail messages and for checking IP packet fragmentation.

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

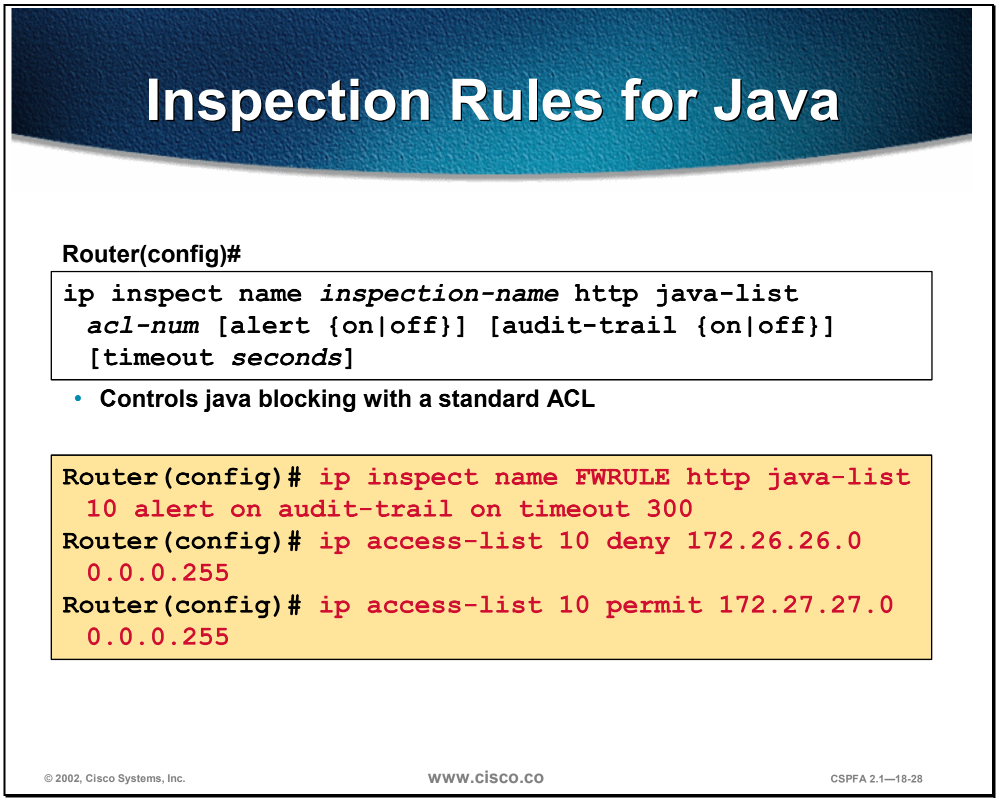
The syntax for the **ip inspect name** command is as follows:

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

**no ip inspect name** *inspection-name protocol*

**no ip inspect name**

| | |
|---|---|
| **name** ***inspection-name*** | Names the set of inspection rules. If you want to add a protocol to an existing set of rules, use the same ***inspection-name***. |
| ***protocol*** | The protocol to inspect. Use of the following keywords: **tcp**, **udp**, **cuseeme**, **ftp**, **http**, **h323**, **netshow**, **rcmd**, **realaudio**, **rpc**, **smtp**, **sqlnet**, **streamworks**, **tftp**, or **vdolive**. |
| **alert {on \| off}** | (Optional.) For each inspected protocol, the generation of alert messages can be set to on or off. If no option is selected, alerts are generated based on the setting of the **ip inspect alert-off** command. |
| **audit-trail {on** \| **off}** | (Optional.) For each inspected protocol, **audit-trail** can be set to on or off. If no option is selected, audit trail messages are generated based on the setting of the **ip inspect audit trail** command. |
| **timeout** ***seconds*** | (Optional.) To override the global TCP or UDP idle timeouts for the specified protocol, specify the number of seconds for a different idle timeout. This timeout overrides the global TCP and UPD timeouts, but will not override the global DNS timeout. |



## Inspection Rules for Java

Router(config)#

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

- Controls java blocking with a standard ACL

```
Router(config)# ip inspect name FWRULE http java-list
  10 alert on audit-trail on timeout 300
Router(config)# ip access-list 10 deny 172.26.26.0
  0.0.0.255
Router(config)# ip access-list 10 permit 172.27.27.0
  0.0.0.255
```

© 2002, Cisco Systems, Inc. www.cisco.co CSPFA 2.1—18-28

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

**Note** If you do not configure an ACL, but use a "placeholder" ACL in the **ip inspect name *inspection-name* http** command, all Java applets will be blocked.

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

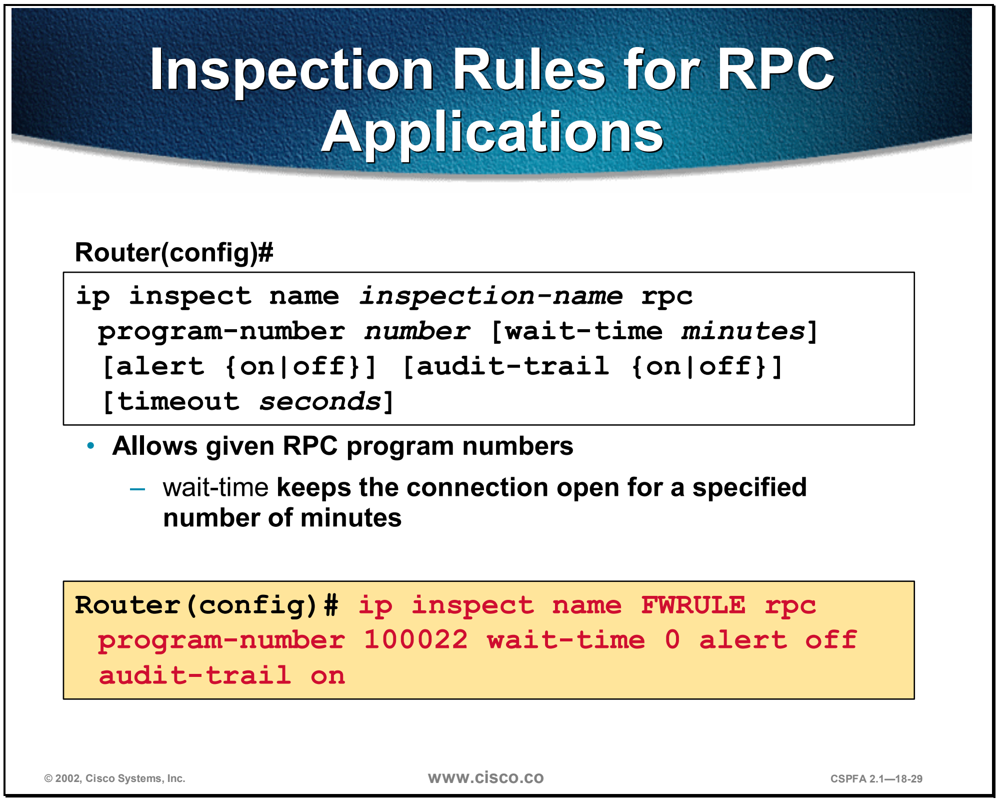
The syntax for the **ip inspect name** command for Java applet filtering inspection is as follows:

**ip inspect name *inspection-name* http java-list *acl-num* [alert {on | off}] [audit-trail {on | off}] [timeout *seconds*]**

**no ip inspect name *inspection-name* http**

| | |
|---|---|
| **name *inspection-name*** | Names the set of inspection rules. If you want to add a protocol to an existing set of rules, use the same ***inspection-name*** as the existing set of rules. |
| **http** | Specifies the HTTP protocol used. |

| | |
|---|---|
| **java-list *acl-num*** | Specifies the ACL (name or number) to use to determine "friendly" sites. This keyword is available only for the HTTP protocol for Java applet blocking. Java blocking only works with standard ACLs. |
| **alert {on \| off}** | (Optional.) For each inspected protocol, the generation of alert messages can be set to on or off. If no option is selected, alerts are generated based on the setting of the **ip inspect alert-off** command. |
| **audit-trail {on \| off}** | (Optional.) For each inspected protocol, **audit-trail** can be set to on or off. If no option is selected, audit trail messages are generated based on the setting of the **ip inspect audit-trail** command. |
| **timeout *seconds*** | (Optional.) To override the global TCP or UDP idle timeouts for the specified protocol, specify the number of seconds for a different idle timeout. This timeout overrides the global TCP and UPD timeouts, but will not override the global DNS timeout. |



## Inspection Rules for RPC Applications

Router(config)#

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

- **Allows given RPC program numbers**
  - wait-time **keeps the connection open for a specified number of minutes**

```
Router(config)# ip inspect name FWRULE rpc
 program-number 100022 wait-time 0 alert off
 audit-trail on
```

© 2002, Cisco Systems, Inc. www.cisco.co CSPFA 2.1—18-29

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

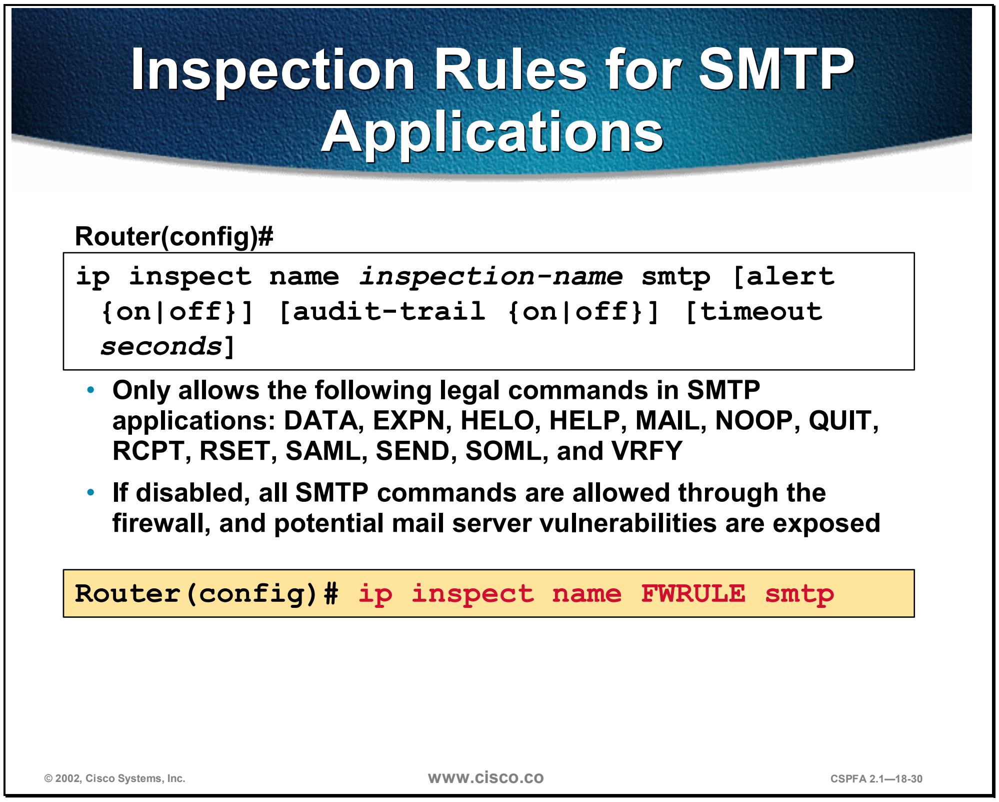
The syntax of the **ip inspect name** command for RPC applications is as follows:

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

**no ip inspect name** ***inspection-name protocol***

| | |
|---|---|
| ***inspection-name*** | Names the set of inspection rules. If you want to add a protocol to an existing set of rules, use the same ***inspection-name*** as the existing set of rules. |
| **rpc program_number** ***number*** | Specifies the program number to permit. |
| **wait-time** *minutes* | (Optional.) Specifies the number of minutes to keep the connection opened in the firewall, even after the application terminates, allowing subsequent connections from the same source address and to the same destination address and port. The default **wait-time** is zero minutes. |
| **alert {on \| off}** | (Optional.) For each inspected protocol, the generation of alert messages can be set to on or off. If no option is selected, alerts are generated based on the setting of the **ip inspect alert-off** command. |
| **audit-trail {on** \| **off}** | (Optional.) For each inspected protocol, **audit-trail** can be set to on or off. If no option is selected, audit trail messages are generated based on the setting of the **ip inspect audit-trail** command. |

| | |
|---|---|
| **timeout *seconds*** | (Optional.) To override the global TCP or UDP idle timeouts for the specified protocol, specify the number of seconds for a different idle timeout. This timeout overrides the global TCP and UPD timeouts, but will not override the global DNS timeout. |



## Inspection Rules for SMTP Applications

Router(config)#

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

- **Only allows the following legal commands in SMTP applications: DATA, EXPN, HELO, HELP, MAIL, NOOP, QUIT, RCPT, RSET, SAML, SEND, SOML, and VRFY**
- **If disabled, all SMTP commands are allowed through the firewall, and potential mail server vulnerabilities are exposed**

```
Router(config)# ip inspect name FWRULE smtp
```

© 2002, Cisco Systems, Inc. www.cisco.co CSPFA 2.1—18-30

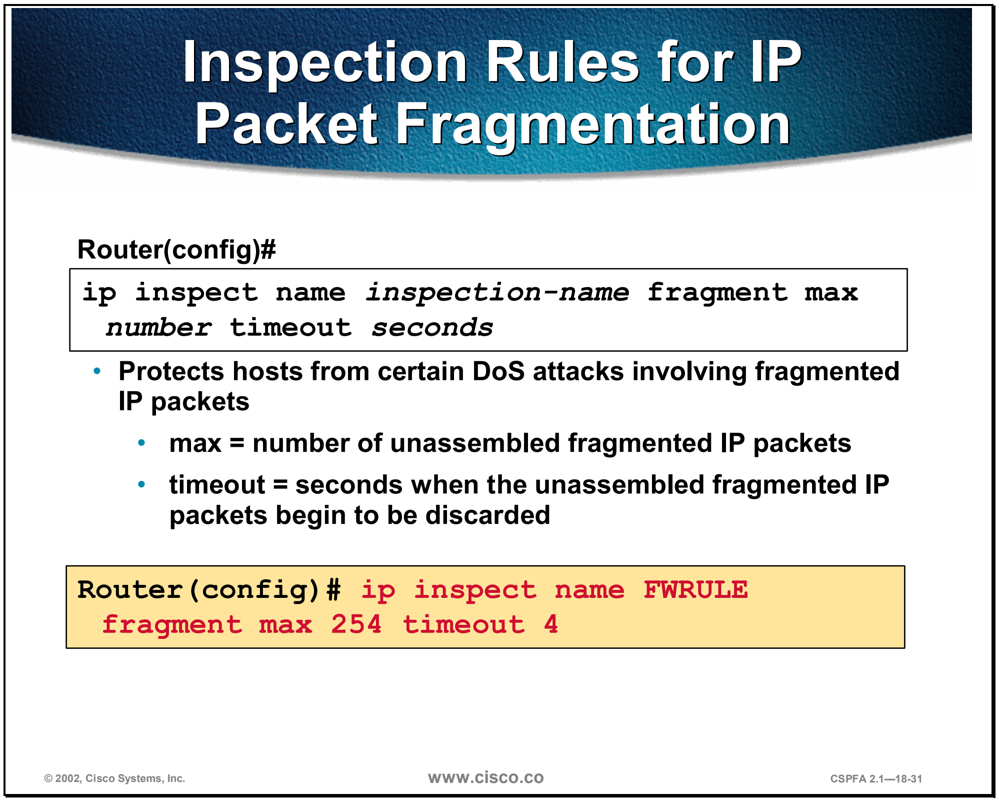
SMTP inspection causes SMTP commands to be inspected for illegal commands. Any packets with illegal commands are dropped, and the SMTP session hangs and eventually times out. An illegal command is any command except for the following legal commands: DATA, EXPN, HELO, HELP, MAIL, NOOP, QUIT, RCPT, RSET, SAML, SEND, SOML, and VRFY.

The syntax for the **ip inspect name** command for SMTP application inspection is as follows:

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

**no ip inspect name** ***inspection-name*** **smtp**

| | |
|---|---|
| **name *inspection-name*** | Names the set of inspection rules. If you want to add a protocol to an existing set of rules, use the same ***inspection-name*** as the existing set of rules. |
| **smtp** | Specifies the SMTP protocol for inspection. |
| **alert {on \| off}** | (Optional.) For each inspected protocol, the generation of alert messages can be set to on or off. If no option is selected, alerts are generated based on the setting of the **ip inspect alert-off** command. |
| **audit-trail {on \| off}** | (Optional.) For each inspected protocol, **audit-trail** can be set to on or off. If no option is selected, audit trail messages are generated based on the setting of the **ip inspect audit-trail** command. |
| **timeout *seconds*** | (Optional.) To override the global TCP or UDP idle timeouts for the specified protocol, specify the number of seconds for a different idle timeout. This timeout overrides the global TCP and UDP timeouts, but will not override the global DNS timeout. |



## Inspection Rules for IP Packet Fragmentation

Router(config)#

```
ip inspect name inspection-name fragment max
  number timeout seconds
```

- **Protects hosts from certain DoS attacks involving fragmented IP packets**
  - **max = number of unassembled fragmented IP packets**
  - **timeout = seconds when the unassembled fragmented IP packets begin to be discarded**

```
Router(config)# ip inspect name FWRULE
  fragment max 254 timeout 4
```

© 2002, Cisco Systems, Inc. www.cisco.co CSPFA 2.1—18-31

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

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

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

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

The syntax of the **ip inspect name** command for IP packet fragmentation is as follows:

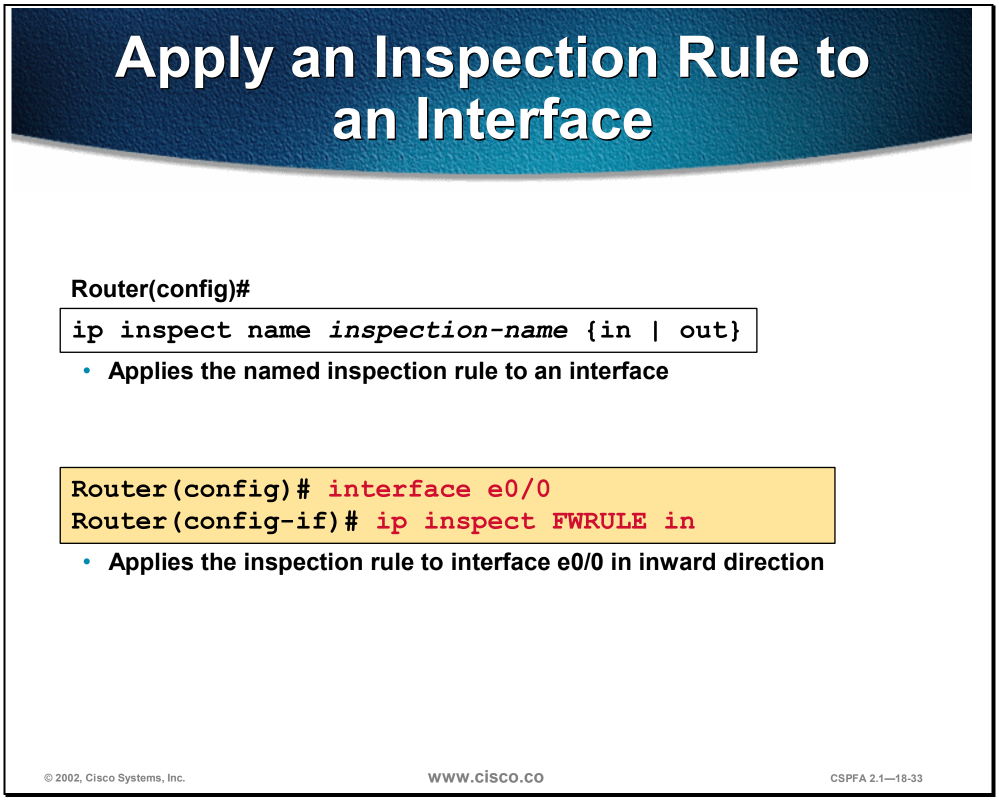
**ip inspect name** ***inspection-name*** **fragment max** ***number*** **timeout** ***seconds***

**no ip inspect name** ***inspection-name*** **fragment**

| | |
|---|---|
| ***inspection-name*** | Names the set of inspection rules. If you want to add a protocol to an existing set of rules, use the same ***inspection-name*** as the existing set of rules. |
| **fragment** | Specifies fragment inspection for the named rule. |
| **max** ***number*** | Specifies the maximum number of unassembled packets for which state information (structures) is allocated by the software. Unassembled packets are packets that arrive at the router interface before the initial packet for a session. The acceptable range is 50 through 10000. The default is 256 state entries.<br>Memory is allocated for the state structures, and setting this value to a larger number may cause memory resources to be exhausted. |
| **timeout** ***seconds*** | Configures the number of seconds that a packet state structure remains active. When the timeout value expires, the router drops the unassembled packet, freeing that structure for use by another packet. The default timeout value is one second.<br>If this number is set to a value greater than one second, it will be automatically adjusted by the software when the number of free state structures goes below certain thresholds: when the number of free states is less than 32, the timeout will be divided by 2; when the number of free states is less than 16, the timeout will be set to 1 second. |

# Inspection Rules and ACLs Applied to Router Interfaces

This section discusses the application of inspection rules and ACLs to router interfaces.



## Apply an Inspection Rule to an Interface

Router(config)#

```
ip inspect name inspection-name {in | out}
```

- Applies the named inspection rule to an interface

```
Router(config)# interface e0/0
Router(config-if)# ip inspect FWRULE in
```

- Applies the inspection rule to interface e0/0 in inward direction

© 2002, Cisco Systems, Inc.  www.cisco.co  CSPFA 2.1—18-33

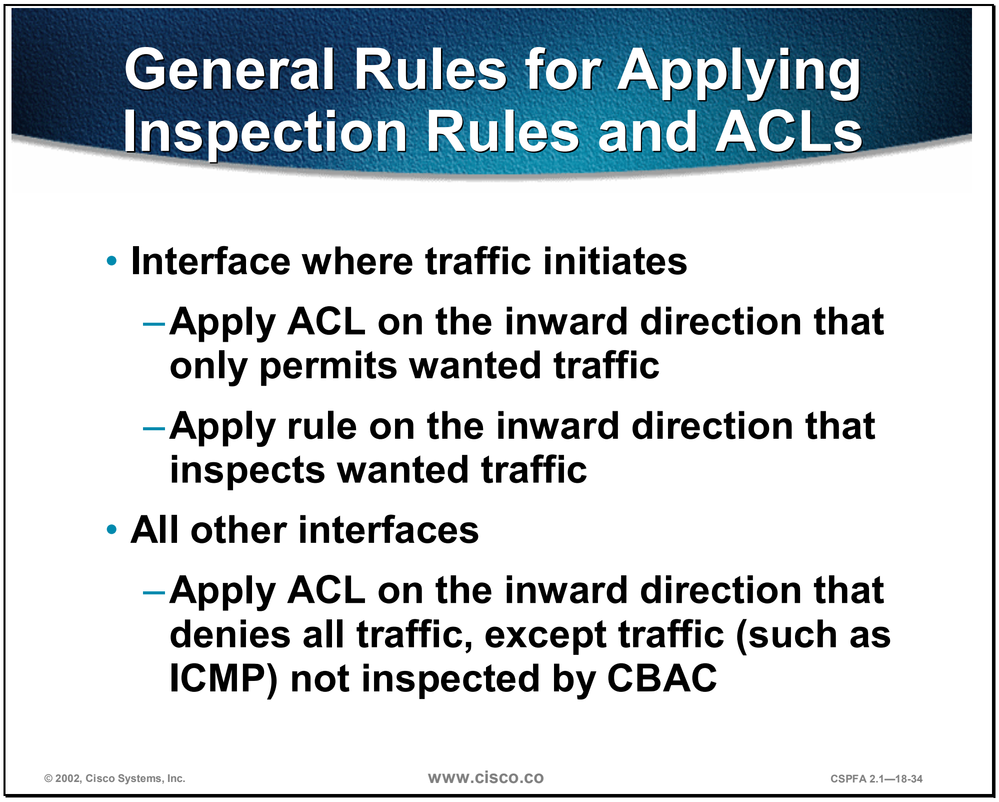
To apply a set of inspection rules to an interface, use the **ip inspect** interface configuration command. Use the **no** form of this command to remove the set of rules from the interface.

The syntax for the **ip inspect** command is as follows:

**ip inspect name *inspection-name* {in | out }**

**no ip inspect inspection-name {in | out}**

| | |
|---|---|
| ***inspection-name*** | Names the set of inspection rules. |
| **in** | Applies the inspection rules to inbound traffic. |
| **out** | Applies the inspection rules to outbound traffic. |



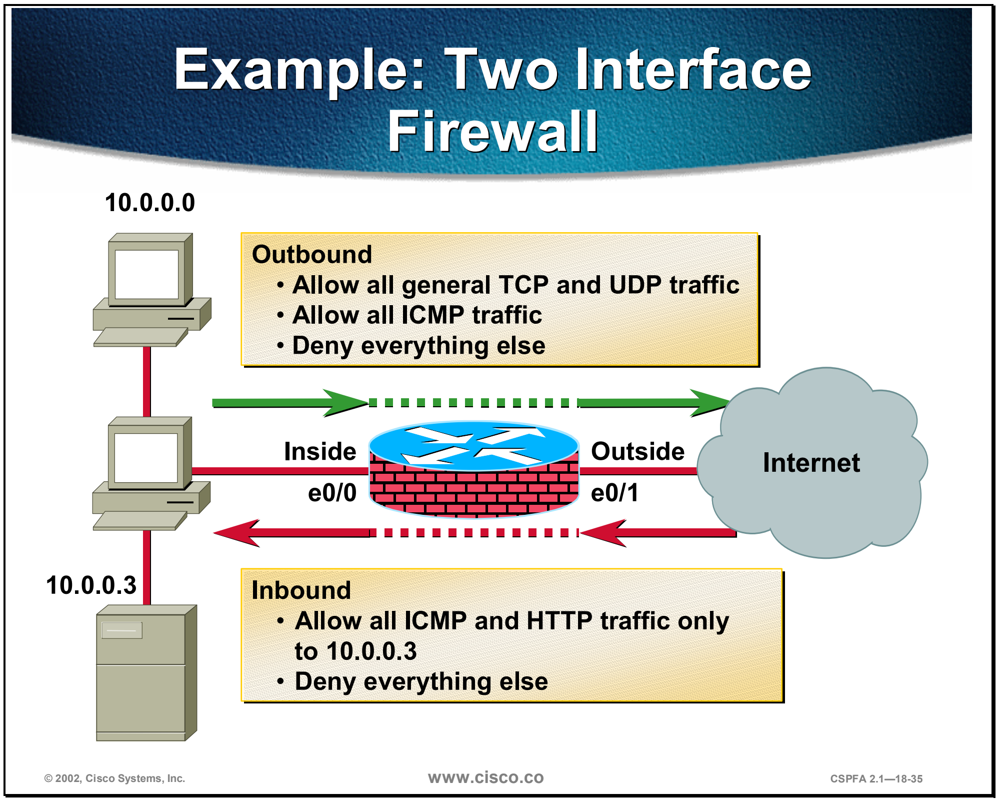
## General Rules for Applying Inspection Rules and ACLs

- Interface where traffic initiates
  - Apply ACL on the inward direction that only permits wanted traffic
  - Apply rule on the inward direction that inspects wanted traffic
- All other interfaces
  - Apply ACL on the inward direction that denies all traffic, except traffic (such as ICMP) not inspected by CBAC

© 2002, Cisco Systems, Inc. www.cisco.co CSPFA 2.1—18-34

For the Cisco IOS Firewall to be effective, both inspection rules and ACLs must be strategically applied to all the router's interfaces. The following is the general rule of thumb for applying inspection rules and ACLs on the router:

- On the interface where traffic initiates
  - Apply the ACL on the inward direction that only permits wanted traffic.
  - Apply the rule on the inward direction that inspects wanted traffic.
- On all other interfaces apply the ACL on the inward direction that denies all traffic, except traffic (such as ICMP) not inspected by CBAC.



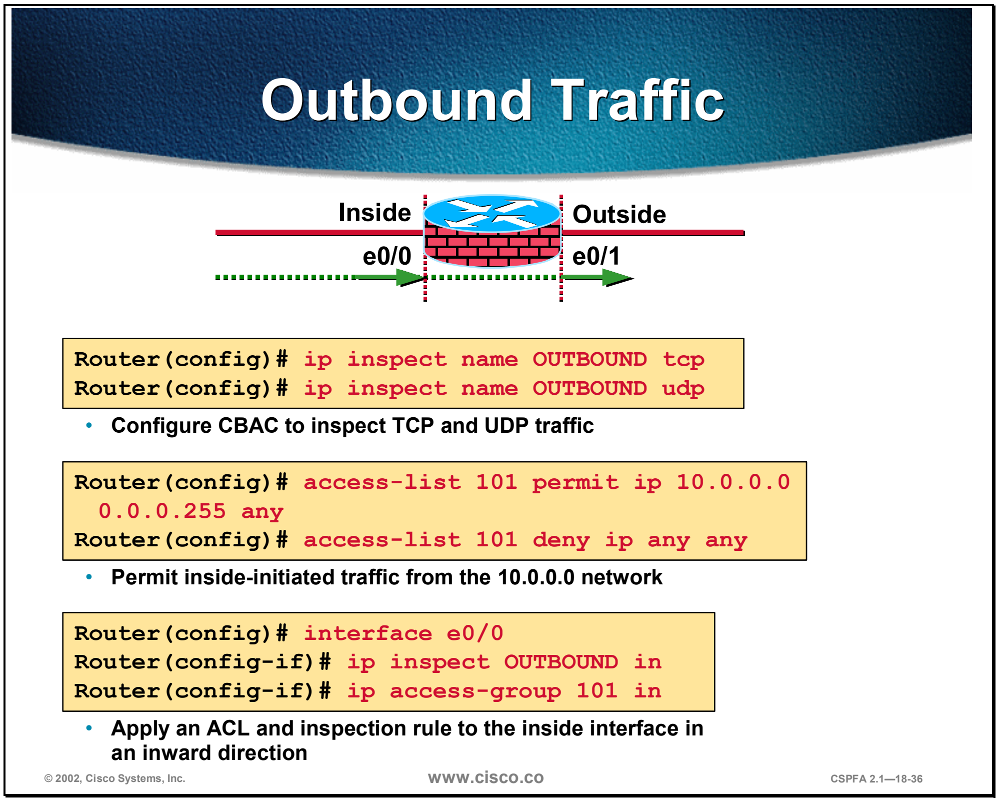
## Example: Two Interface Firewall

Outbound
- Allow all general TCP and UDP traffic
- Allow all ICMP traffic
- Deny everything else

Inbound
- Allow all ICMP and HTTP traffic only to 10.0.0.3
- Deny everything else

www.cisco.co CSPFA 2.1—18-35

As an example, configure the router to be a firewall between two networks: inside and outside. The following is the security policy to implement: allow all general TCP and UDP traffic initiated on the inside (outbound) from network 10.0.0.0 to access the Internet. ICMP traffic will also be allowed from the same network. Other networks on the inside, which are not defined, must be denied. For traffic initiated on the outside (inbound), allow everyone to access only ICMP and HTTP to host 10.0.0.3. Any other traffic must be denied.



## Outbound Traffic

```
Router(config)# ip inspect name OUTBOUND tcp
Router(config)# ip inspect name OUTBOUND udp
```

- Configure CBAC to inspect TCP and UDP traffic

```
Router(config)# access-list 101 permit ip 10.0.0.0
  0.0.0.255 any
Router(config)# access-list 101 deny ip any any
```

- Permit inside-initiated traffic from the 10.0.0.0 network

```
Router(config)# interface e0/0
Router(config-if)# ip inspect OUTBOUND in
Router(config-if)# ip access-group 101 in
```

- Apply an ACL and inspection rule to the inside interface in an inward direction

© 2002, Cisco Systems, Inc. www.cisco.co CSPFA 2.1—18-36

To implement the security policy of the previous example, do the following for outbound traffic:

**Step 1** Write a rule to inspect TCP and UDP traffic:

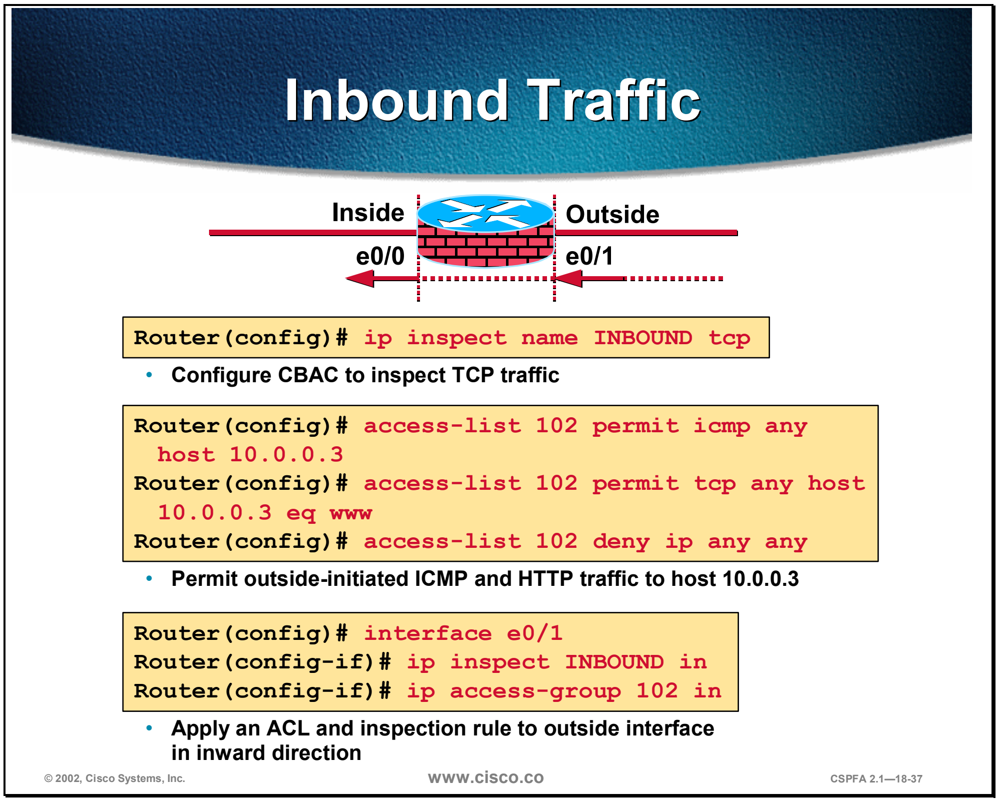
```
Router(config)# ip inspect name OUTBOUND tcp
Router(config)# ip inspect name OUTBOUND udp
```

**Step 2** Write an ACL that permits IP traffic from the 10.0.0.0 network to any destination:

```
Router(config)# access-list 101 permit ip 10.0.0.0 0.0.0.255 any
Router(config)# access-list 101 deny ip any any
```

**Step 3** Apply the inspection rule and ACL to the inside interface on the inward direction:

```
Router(config)# interface e0/0
Router(config-if)# ip inspect OUTBOUND in
Router(config-if)# ip access-group 101 in
```



## Inbound Traffic

```
Router(config)# ip inspect name INBOUND tcp
```

- Configure CBAC to inspect TCP traffic

```
Router(config)# access-list 102 permit icmp any
  host 10.0.0.3
Router(config)# access-list 102 permit tcp any host
  10.0.0.3 eq www
Router(config)# access-list 102 deny ip any any
```

- Permit outside-initiated ICMP and HTTP traffic to host 10.0.0.3

```
Router(config)# interface e0/1
Router(config-if)# ip inspect INBOUND in
Router(config-if)# ip access-group 102 in
```

- Apply an ACL and inspection rule to outside interface in inward direction

© 2002, Cisco Systems, Inc. www.cisco.co CSPFA 2.1—18-37

To implement the security policy of the previous example, do the following for inbound traffic:

**Step 1** Write a rule to inspect TCP traffic:

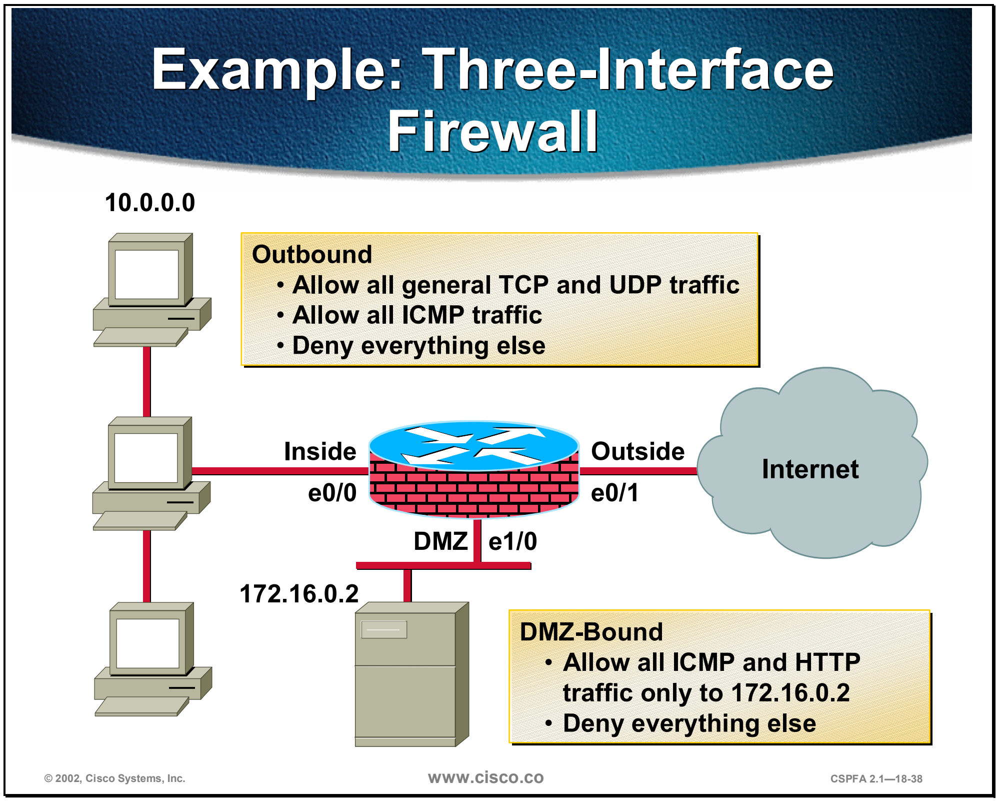
```
Router(config)# ip inspect name INBOUND tcp
```

**Step 2** Write an ACL that permits ICMP- and HTTP-only traffic from the Internet to the 10.0.0.3 host:

```
Router(config)# access-list 102 permit icmp any host 10.0.0.3
Router(config)# access-list 102 permit tcp any host 10.0.0.3 eq www
Router(config)# access-list 102 deny ip any any
```

**Step 3** Apply the inspection rule and ACL to the outside interface in the inward direction:

```
Router(config)# interface e0/1
Router(config-if)# ip inspect INBOUND in
Router(config-if)# ip access-group 102 in
```



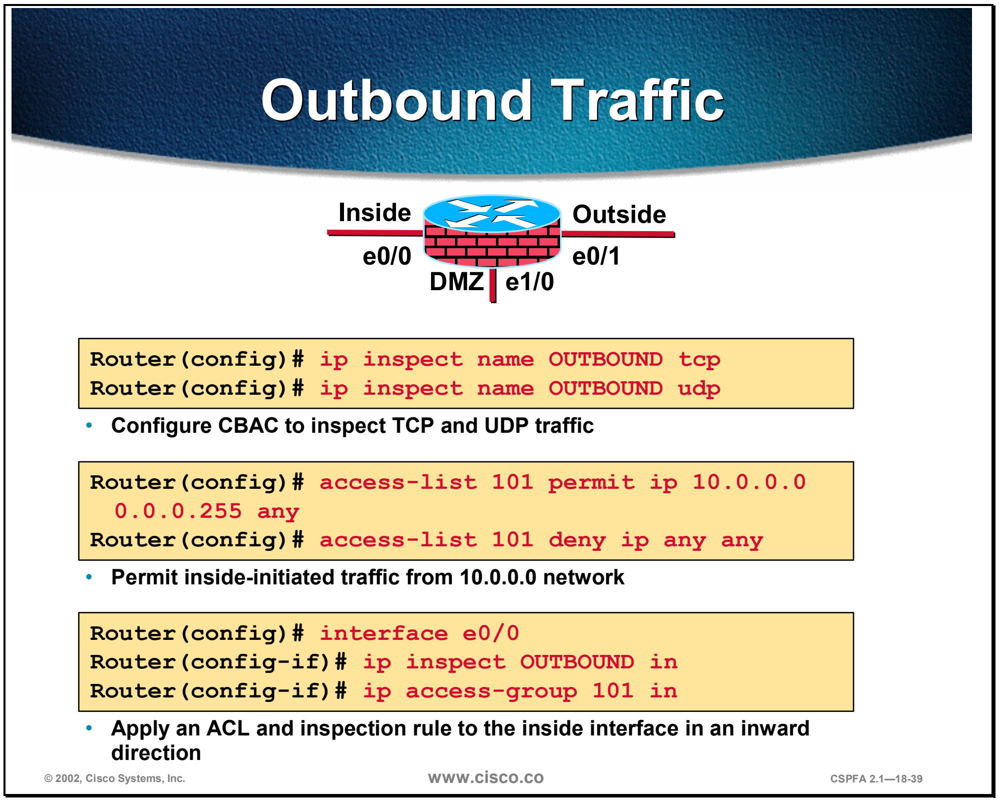
## Example: Three-Interface Firewall

Outbound
- Allow all general TCP and UDP traffic
- Allow all ICMP traffic
- Deny everything else

DMZ-Bound
- Allow all ICMP and HTTP traffic only to 172.16.0.2
- Deny everything else

© 2002, Cisco Systems, Inc. www.cisco.co CSPFA 2.1—18-38

As an example, configure the router to be a firewall between three networks: inside, outside, and DMZ. The following is the security policy to implement: allow all general TCP and UDP traffic initiated on the inside (outbound) from network 10.0.0.0 to access the Internet and the DMZ host 172.16.0.2. ICMP traffic will also be allowed from the same network to the Internet and the DMZ host. Other networks on the inside, which are not defined, must be denied. For traffic initiated on the outside (inbound) allow everyone to only access ICMP and HTTP to DMZ host 172.16.0.2. Any other traffic must be denied.



# Outbound Traffic

```
Router(config)# ip inspect name OUTBOUND tcp
Router(config)# ip inspect name OUTBOUND udp
```

- Configure CBAC to inspect TCP and UDP traffic

```
Router(config)# access-list 101 permit ip 10.0.0.0
  0.0.0.255 any
Router(config)# access-list 101 deny ip any any
```

- Permit inside-initiated traffic from 10.0.0.0 network

```
Router(config)# interface e0/0
Router(config-if)# ip inspect OUTBOUND in
Router(config-if)# ip access-group 101 in
```

- Apply an ACL and inspection rule to the inside interface in an inward direction

© 2002, Cisco Systems, Inc. www.cisco.co CSPFA 2.1—18-39

To implement the security policy of the previous example, do the following for outbound traffic:

**Step 1** Write a rule to inspect TCP and UDP traffic:

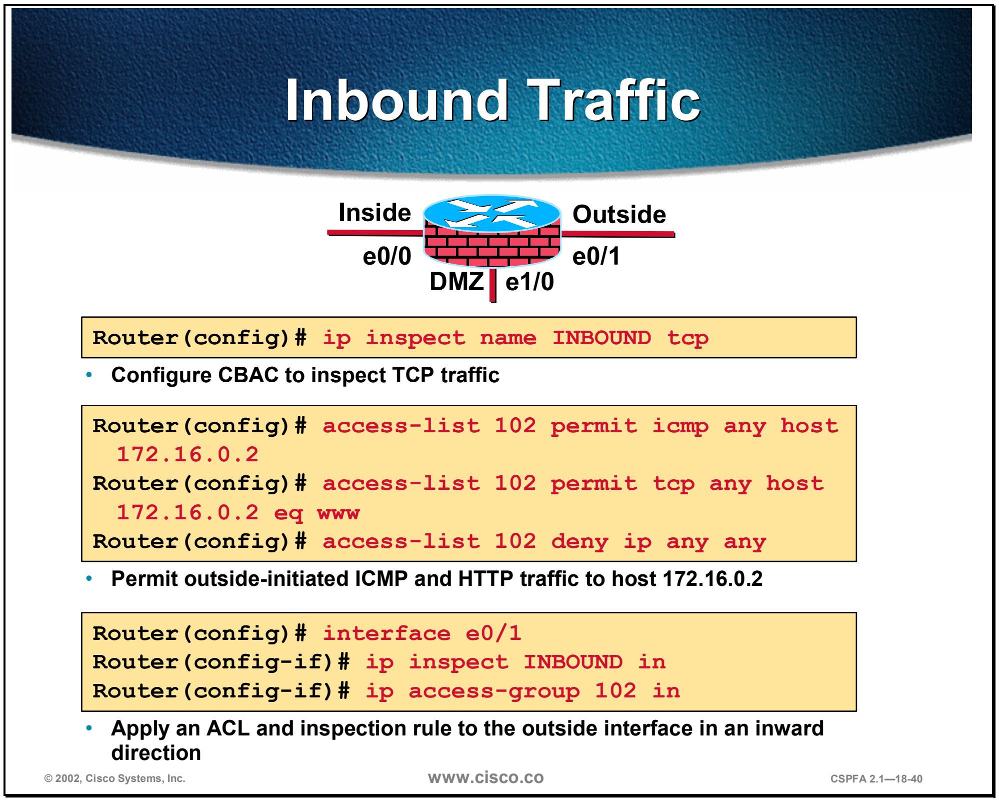
```
Router(config)# ip inspect name OUTBOUND tcp
Router(config)# ip inspect name OUTBOUND udp
```

**Step 2** Write an ACL that permits IP traffic from the 10.0.0.0 network to any destination:

```
Router(config)# access-list 101 permit ip 10.0.0.0 0.0.0.255 any
Router(config)# access-list 101 deny ip any any
```

**Step 3** Apply the inspection rule and ACL to the inside interface in the inward direction:

```
Router(config)# interface e0/0
Router(config-if)# ip inspect OUTBOUND in
Router(config-if)# ip access-group 101 in
```


Inside
Outside
e0/0
e0/1
DMZ
e1/0

## Inbound Traffic

```
Router(config)# ip inspect name INBOUND tcp
```

- Configure CBAC to inspect TCP traffic

```
Router(config)# access-list 102 permit icmp any host
  172.16.0.2
Router(config)# access-list 102 permit tcp any host
  172.16.0.2 eq www
Router(config)# access-list 102 deny ip any any
```

- Permit outside-initiated ICMP and HTTP traffic to host 172.16.0.2

```
Router(config)# interface e0/1
Router(config-if)# ip inspect INBOUND in
Router(config-if)# ip access-group 102 in
```

- Apply an ACL and inspection rule to the outside interface in an inward direction

© 2002, Cisco Systems, Inc. www.cisco.co CSPFA 2.1—18-40

To implement the security policy of the previous example, do the following for inbound traffic:

**Step 1** Write a rule to inspect TCP traffic:

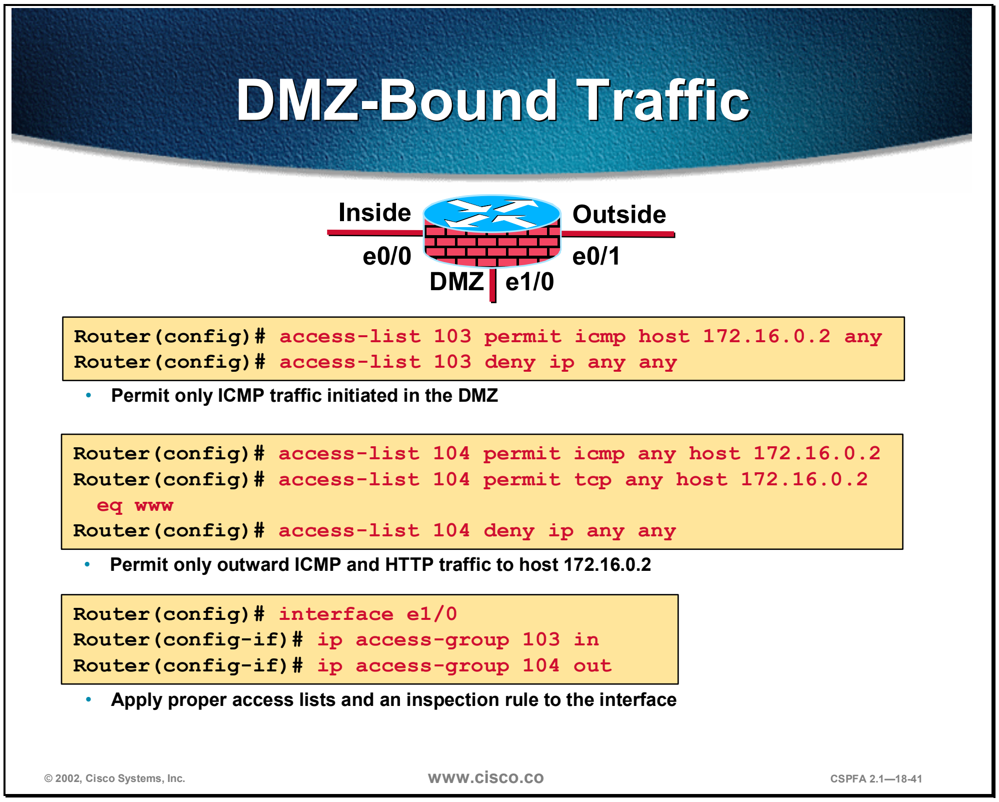
```
Router(config)# ip inspect name INBOUND tcp
```

**Step 2** Write an ACL that permits ICMP- and HTTP-only traffic from the Internet to the 172.16.0.2 host:

```
Router(config)# access-list 102 permit icmp any host 172.16.0.2
Router(config)# access-list 102 permit tcp any host 172.16.0.2 eq www
Router(config)# access-list 102 deny ip any any
```

**Step 3** Apply the inspection rule and ACL to the outside interface in the inward direction:

```
Router(config)# interface e0/1
Router(config-if)# ip inspect INBOUND in
Router(config-if)# ip access-group 102 in
```



## DMZ-Bound Traffic

```
Router(config)# access-list 103 permit icmp host 172.16.0.2 any
Router(config)# access-list 103 deny ip any any
```

- Permit only ICMP traffic initiated in the DMZ

```
Router(config)# access-list 104 permit icmp any host 172.16.0.2
Router(config)# access-list 104 permit tcp any host 172.16.0.2
  eq www
Router(config)# access-list 104 deny ip any any
```

- Permit only outward ICMP and HTTP traffic to host 172.16.0.2

```
Router(config)# interface e1/0
Router(config-if)# ip access-group 103 in
Router(config-if)# ip access-group 104 out
```

- Apply proper access lists and an inspection rule to the interface

© 2002, Cisco Systems, Inc. www.cisco.co CSPFA 2.1—18-41

To implement the security policy of the previous example, do the following for inbound traffic:

**Step 1** Write an ACL to permit only ICMP traffic to initiate from the DMZ host:

```
Router(config)# access-list 103 permit icmp host 172.16.0.2 any
Router(config)# access-list 103 deny ip any any
```

**Step 2** Write an ACL that permits ICMP- and HTTP-only traffic from any network to the 172.16.0.2 host:

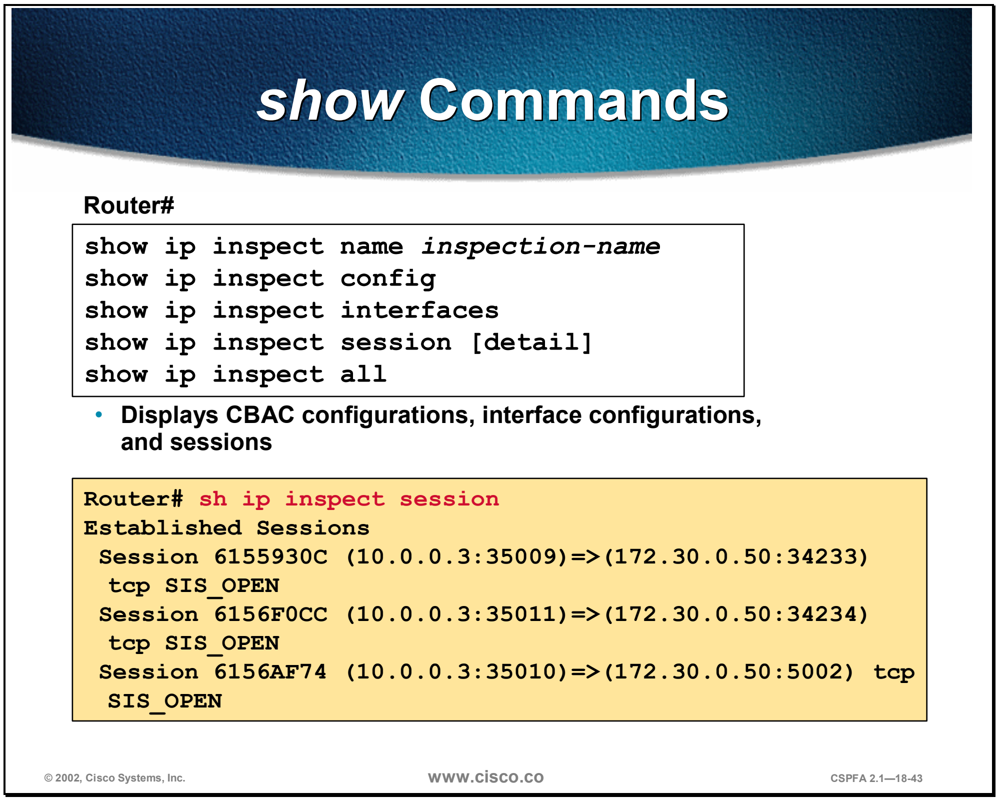
```
Router(config)# access-list 104 permit icmp any host 172.16.0.2
Router(config)# access-list 104 permit tcp any host 172.16.0.2 eq www
Router(config)# access-list 104 deny ip any any
```

**Step 3** Apply the ACLs to the DMZ interface:

```
Router(config)# interface e1/0
Router(config-if)# ip access-group 103 in
Router(config-if)# ip access-group 104 out
```

# Test and Verify

This section discusses the commands available to help test and verify CBAC.



## *show* Commands

Router#

```
show ip inspect name inspection-name
show ip inspect config
show ip inspect interfaces
show ip inspect session [detail]
show ip inspect all
```

- Displays CBAC configurations, interface configurations, and sessions

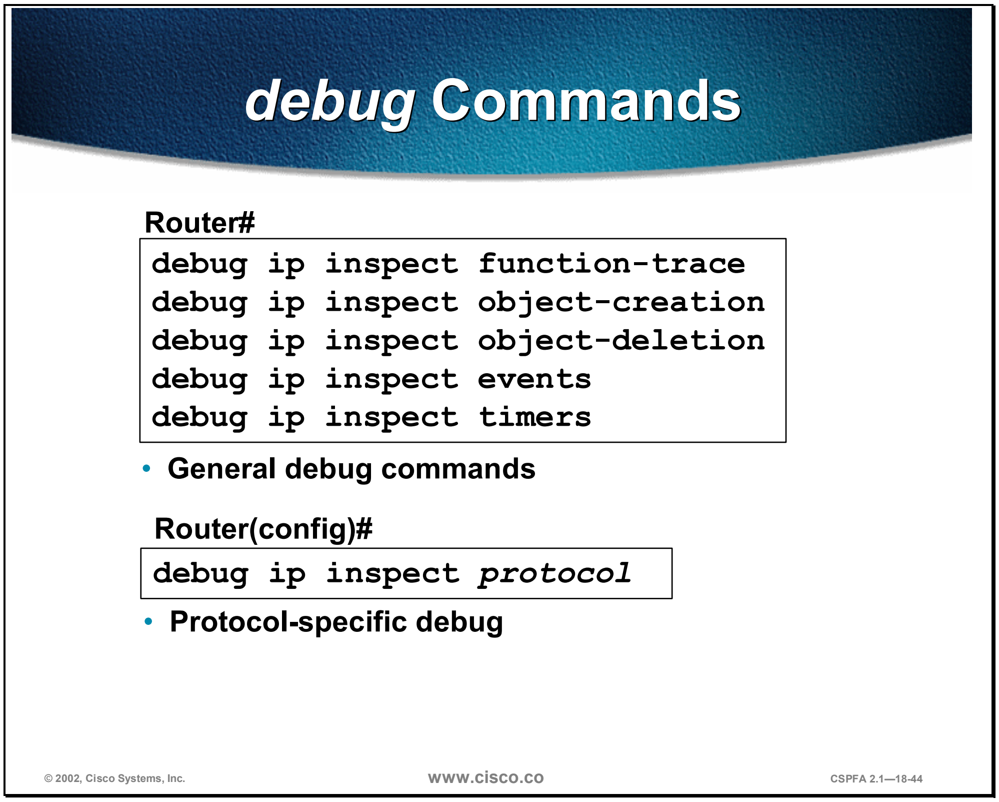
```
Router# sh ip inspect session
Established Sessions
 Session 6155930C (10.0.0.3:35009)=>(172.30.0.50:34233)
  tcp SIS_OPEN
 Session 6156F0CC (10.0.0.3:35011)=>(172.30.0.50:34234)
  tcp SIS_OPEN
 Session 6156AF74 (10.0.0.3:35010)=>(172.30.0.50:5002) tcp
  SIS_OPEN
```

© 2002, Cisco Systems, Inc. www.cisco.co CSPFA 2.1—18-43

The syntax for the **show ip inspect** command is as follows:

**show ip inspect name** ***inspection-name*** | **config** | **interfaces** | **session [detail]** | **all**

| | |
|---|---|
| ***inspection-name*** | Shows the configured inspection rule for ***inspection-name***. |
| **config** | Shows the complete CBAC inspection configuration. |
| **interfaces** | Shows interface configuration with respect to applied inspection rules and ACLs. |
| **session [detail]** | Shows existing sessions that are currently being tracked and inspected by CBAC. The optional **detail** keyword shows additional details about these sessions. |
| **all** | Shows the complete CBAC configuration and all existing sessions that are currently being tracked and inspected by CBAC. |



## *debug* Commands

Router#

```
debug ip inspect function-trace
debug ip inspect object-creation
debug ip inspect object-deletion
debug ip inspect events
debug ip inspect timers
```

- General debug commands

Router(config)#

```
debug ip inspect protocol
```

- Protocol-specific debug

© 2002, Cisco Systems, Inc. www.cisco.co CSPFA 2.1—18-44

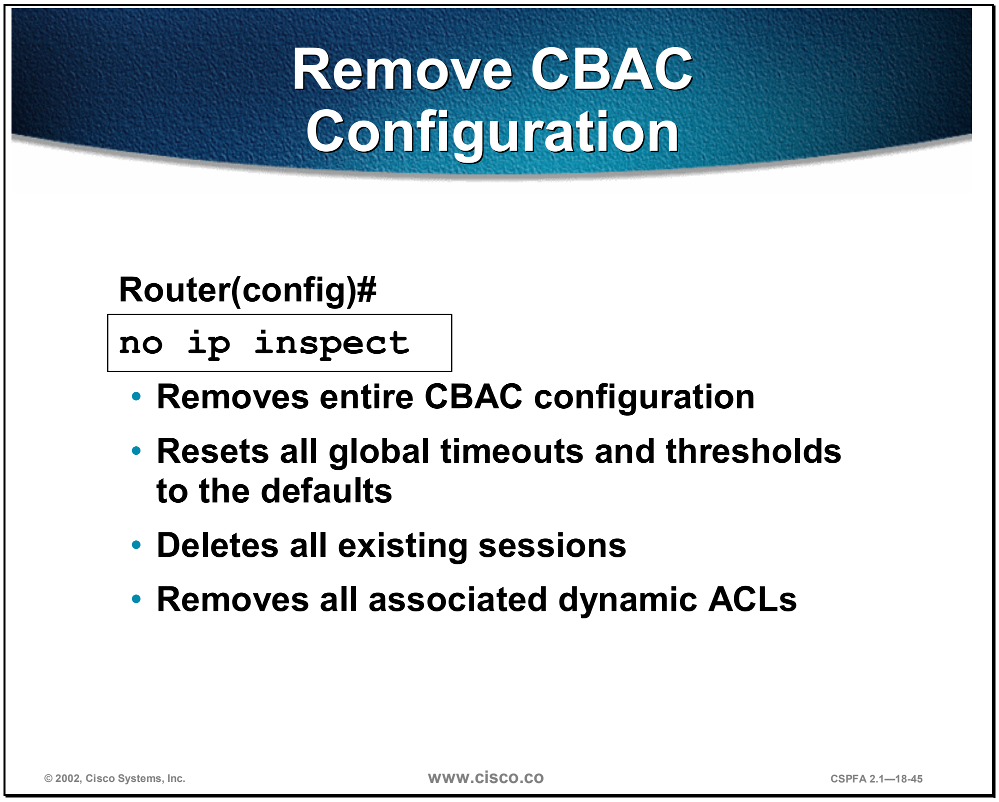
To display messages about CBAC events, use the **debug ip inspect** EXEC command. The **no** form of this command disables debugging output.

The syntax for the **debug ip inspect** command is as follows:

**debug ip inspect {function-trace | object-creation | object-deletion | events | timers |** ***protocol*** **| detailed}**

**no debug ip inspect**

| | |
|---|---|
| **function-trace** | Displays messages about software functions called by CBAC. |
| **object-creation** | Displays messages about software objects being created by CBAC. Object creation corresponds to the beginning of CBAC-inspected sessions. |
| **object-deletion** | Displays messages about software objects being deleted by CBAC. Object deletion corresponds to the closing of CBAC-inspected sessions. |
| **events** | Displays messages about CBAC software events, including information about CBAC packet processing. |
| **timers** | Displays messages about CBAC timer events, such as when a CBAC idle timeout is reached. |
| ***protocol*** | Displays messages about CBAC-inspected protocol events, including details about the protocol's packets. |
| **detailed** | Use this form of the command in conjunction with other CBAC debugging commands. This displays detailed information for all other enabled CBAC debugging. |



# Remove CBAC Configuration

Router(config)#

```
no ip inspect
```

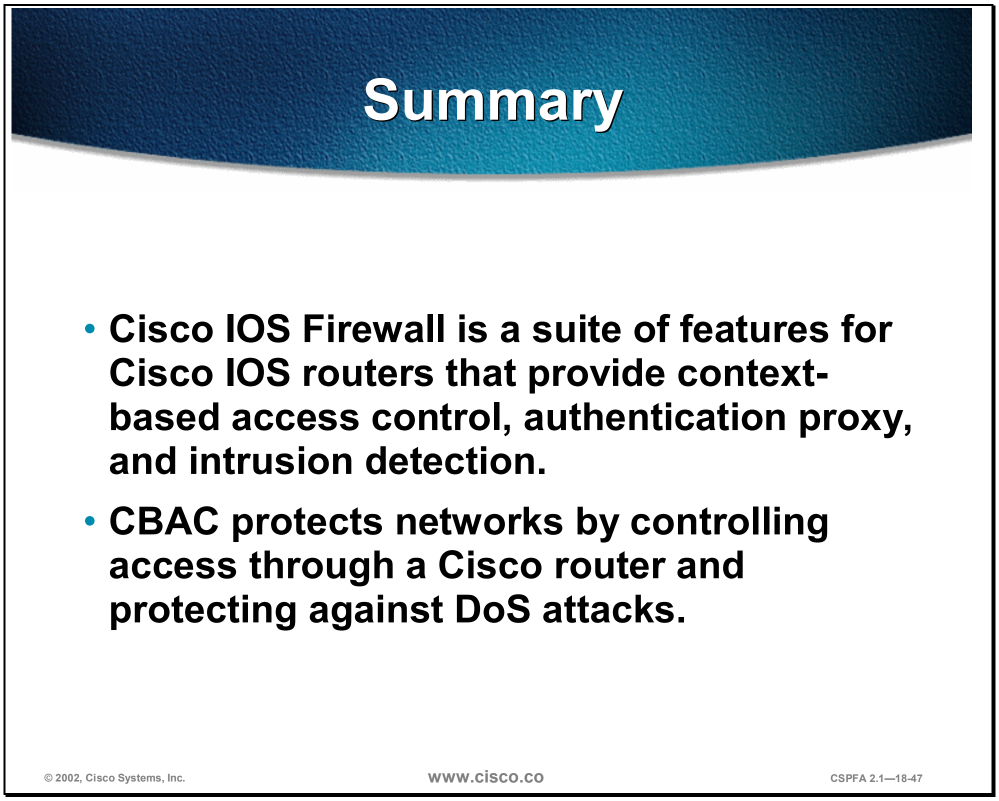
- Removes entire CBAC configuration
- Resets all global timeouts and thresholds to the defaults
- Deletes all existing sessions
- Removes all associated dynamic ACLs

© 2002, Cisco Systems, Inc. www.cisco.co CSPFA 2.1—18-45

Use the **no ip inspect** command to remove the entire CBAC configuration, reset all global timeouts and thresholds to their defaults, delete all existing sessions, and remove all associated dynamic ACLs. This command has no other arguments, keywords, default behavior, or values.

# Summary

This section summarizes what you learned in this chapter.



## Summary

- Cisco IOS Firewall is a suite of features for Cisco IOS routers that provide context-based access control, authentication proxy, and intrusion detection.
- CBAC protects networks by controlling access through a Cisco router and protecting against DoS attacks.

© 2002, Cisco Systems, Inc. www.cisco.co CSPFA 2.1—18-47

# Lab Exercise—Configure CBAC on a Cisco Router

Complete the following lab exercise to practice what you learned in this chapter.

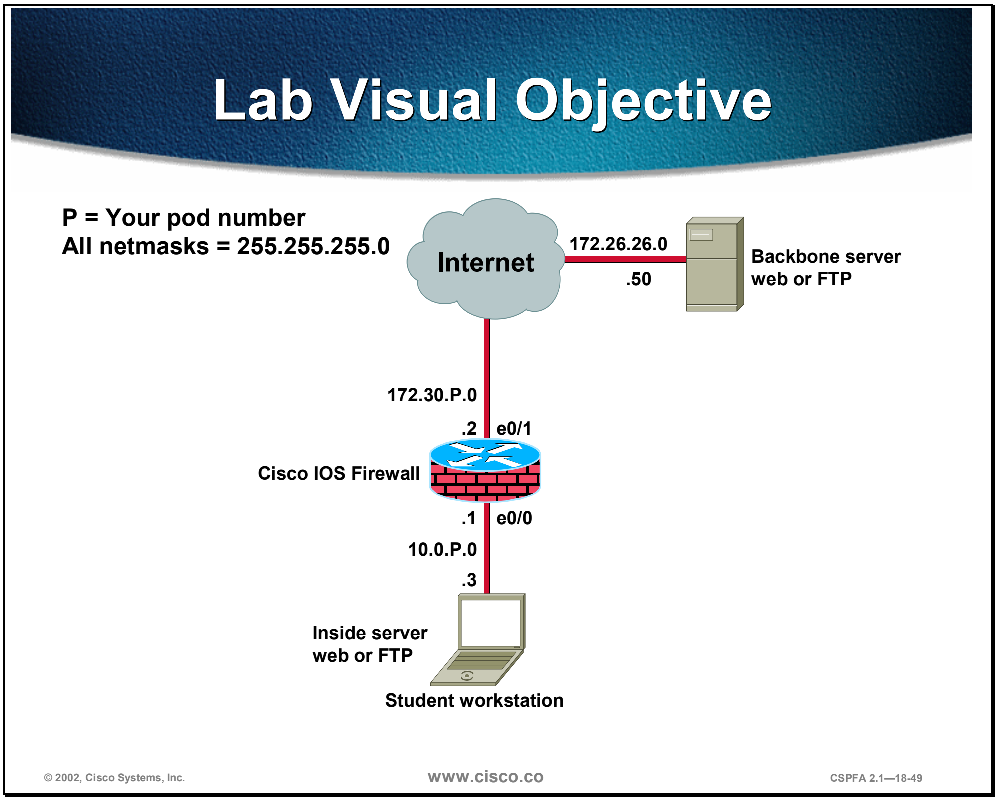
## Objectives

In this lab exercise you will complete the following tasks:

- Configure logging and audit trails.
- Define and apply inspection rules ACLs.
- Test and verify CBAC.

## Lab Visual Objective

The following figure displays the configuration you will complete in this lab exercise.



Lab Visual Objective

P = Your pod number
All netmasks = 255.255.255.0

© 2002, Cisco Systems, Inc. www.cisco.co CSPFA 2.1—18-49

### Directions

Your task in this exercise is to configure CBAC on a Cisco router. Work with your lab partner to perform the following steps in this lab exercise:

- Task 1—Configure Logging and Audit Trails

- Task 2—Define and Apply Inspection Rules ACLs
- Task 3—Test and Verify CBAC

## Task 1—Configure Logging and Audit Trails

To configure logging and audit trails, complete the following steps:

**Step 1** On your router, enable logging to the console and the Syslog server.

```
Router(config)# logging on
Router(config)# logging 10.0.P.3
```

(where P = pod number)

**Step 2** Enable the audit trail.

```
Router(config)# ip inspect audit-trail
```

**Step 3** Save your configuration and return to global configuration mode.

```
Router(config)# end
Router# write memory
```

## Task 2—Define and Apply Inspection Rules ACLs

To define and apply inspection rules and ACLs, complete the following steps:

**Step 1** On your router, define a CBAC rule to inspect all TCP and FTP traffic.

```
Router(config)# ip inspect name FWRULE tcp timeout 300
Router(config)# ip inspect name FWRULE ftp timeout 300
```

**Step 2** Define the ACLs to allow outbound ICMP traffic and CBAC traffic (FTP and WWW). Block all other inside-initiated traffic.

```
Router(config)# access-list 101 permit icmp any any
Router(config)# access-list 101 permit tcp 10.0.P.0 0.0.0.255 any eq ftp
Router(config)# access-list 101 permit tcp 10.0.P.0 0.0.0.255 any eq www
Router(config)# access-list 101 deny ip any any
```

(where P = pod number)

**Step 3** Define ACLs to allow inbound ICMP traffic and CBAC traffic (FTP and WWW) to the inside web or FTP server. Block all other outside-initiated traffic.

```
Router(config)# access-list 102 permit eigrp any any
Router(config)# access-list 102 permit icmp any any
Router(config)# access-list 102 permit tcp any host 10.0.P.3 eq ftp
Router(config)# access-list 102 permit tcp any host 10.0.P.3 eq www
Router(config)# access-list 102 deny ip any any
```

(where P = pod number)

**Step 4** Apply the inspection rule and ACL to the inside interface.

```
Router(config)# interface ethernet 0/0
Router(config-if)# ip inspect FWRULE in
Router(config-if)# ip access-group 101 in
```

**Step 5** Apply the ACL to the outside interface.

```
Router(config-if)# interface ethernet 0/1
Router(config-if)# ip inspect FWRULE in
Router(config-if)# ip access-group 102 in
```

**Step 6** Save your configuration and return to global configuration mode.

```
Router(config-if)# end
Router# write memory
```

## Task 3—Test and Verify CBAC

To test and verify CBAC, complete the following steps:

**Step 1** Check your ACLs.

```
Router# show access-lists
```

**Step 2** From your workstation command prompt, ping the backbone server.

```
C:\> ping 172.26.25.50
Pinging 172.26.26.50 with 32 bytes of data:

Reply from 172.26.26.50: bytes=32 time=34ms TTL=125
Reply from 172.26.26.50: bytes=32 time=34ms TTL=125
Reply from 172.26.26.50: bytes=32 time=34ms TTL=125
Reply from 172.26.26.50: bytes=32 time=36ms TTL=125
```

**Step 3** Use your web browser to connect to the backbone web server. Enter **http://172.26.26..50** in the URL field.

**Step 4** Connect to the backbone FTP server using anonymous FTP.

```
C:\> ftp 172.26.26.50
...
User (10.0.0.3:(none)): anonymous
...
Password: user@
```

**Step 5** Do a directory listing to verify data channel connectivity.

```
ftp> ls
```

**Step 6** On your router, use the following **show** commands to verify CBAC operation.

```
Router# show ip inspect name FWRULE
Router# show ip inspect config
Router# show ip inspect interfaces
Router# show ip inspect sessions
Router# show ip inspect sessions detail
Router# show ip inspect all
```

**Step 7** From your workstation command prompt, ping your peer's inside server.

```
C:\> ping 10.0.Q.3
Pinging 10.0.1.3 with 32 bytes of data:

Reply from 10.0.1.3: bytes=32 time=34ms TTL=125
Reply from 10.0.1.3: bytes=32 time=34ms TTL=125
Reply from 10.0.1.3: bytes=32 time=34ms TTL=125
Reply from 10.0.1.3: bytes=32 time=36ms TTL=125
```

(where Q = peer pod number)

**Step 8** Use your Web browser to connect to your peer's inside server. Enter **http://10.0.Q.3** in the URL field.

(where Q = peer pod number)

**Step 9** Connect to your peer's FTP server using anonymous FTP.

```
C:\> ftp 10.0.Q.3
...
User (10.0.1.3:(none)): anonymous
...
Password: user@
```

(where Q = peer pod number)

**Step 10** On your router, use the following **show** commands to verify CBAC operation.

```
Router# show ip inspect name FWRULE
Router# show ip inspect config
Router# show ip inspect interfaces
Router# show ip inspect sessions
Router# show ip inspect sessions detail
Router# show ip inspect all
```

19

# Cisco IOS Firewall Authentication Proxy Configuration

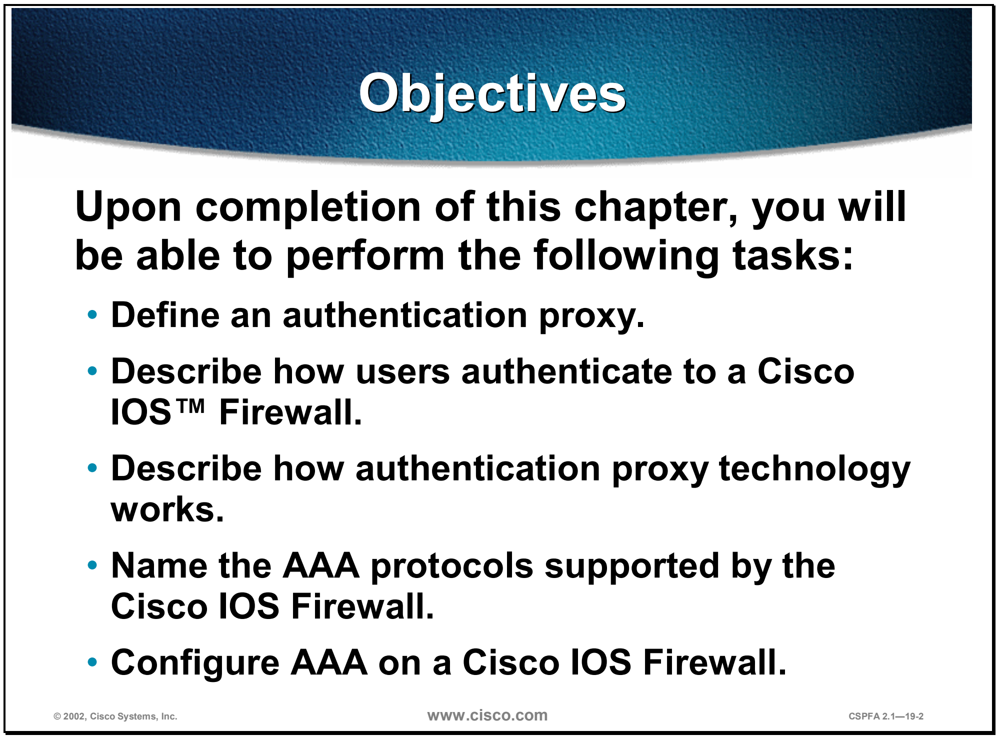
## Overview

This chapter includes the following topics:

- Introduction to the Cisco IOS Firewall authentication proxy
- AAA server configuration
- AAA configuration
- Authentication proxy configuration
- Test and verify the configuration
- Summary
- Lab exercise

# Objectives

This section lists the chapter's objectives.



## Objectives

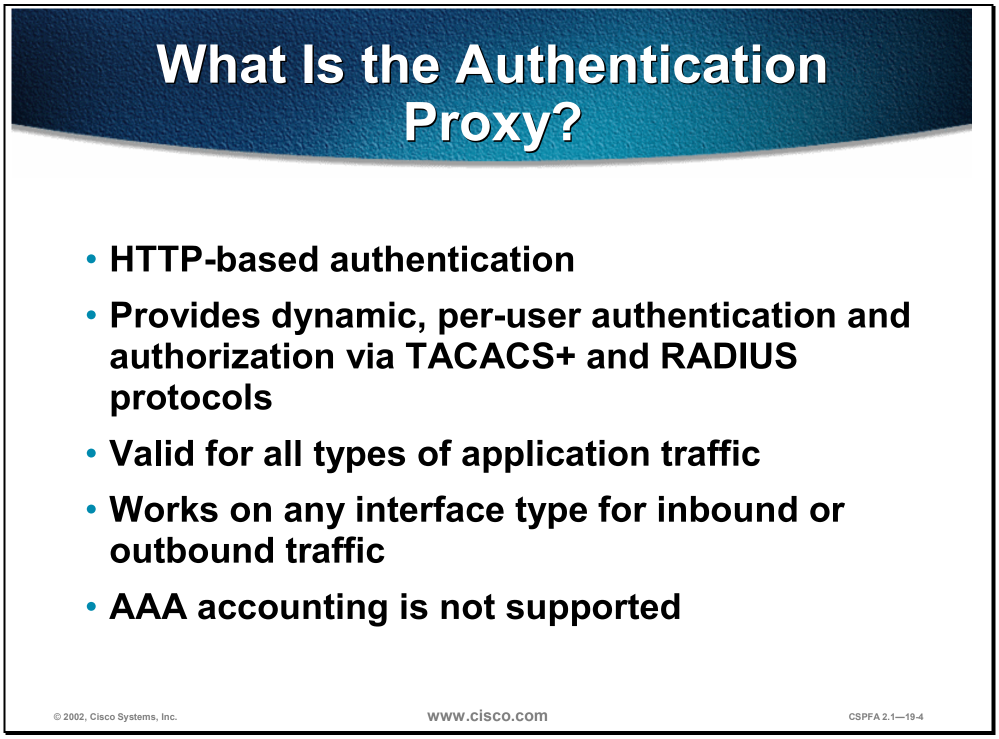
**Upon completion of this chapter, you will be able to perform the following tasks:**

- Define an authentication proxy.
- Describe how users authenticate to a Cisco IOS™ Firewall.
- Describe how authentication proxy technology works.
- Name the AAA protocols supported by the Cisco IOS Firewall.
- Configure AAA on a Cisco IOS Firewall.

www.cisco.com CSPFA 2.1—19-2

# Introduction to the Cisco IOS Firewall Authentication Proxy

This section introduces the features of the IOS™ Firewall authentication proxy.



## What Is the Authentication Proxy?

- HTTP-based authentication
- Provides dynamic, per-user authentication and authorization via TACACS+ and RADIUS protocols
- Valid for all types of application traffic
- Works on any interface type for inbound or outbound traffic
- AAA accounting is not supported

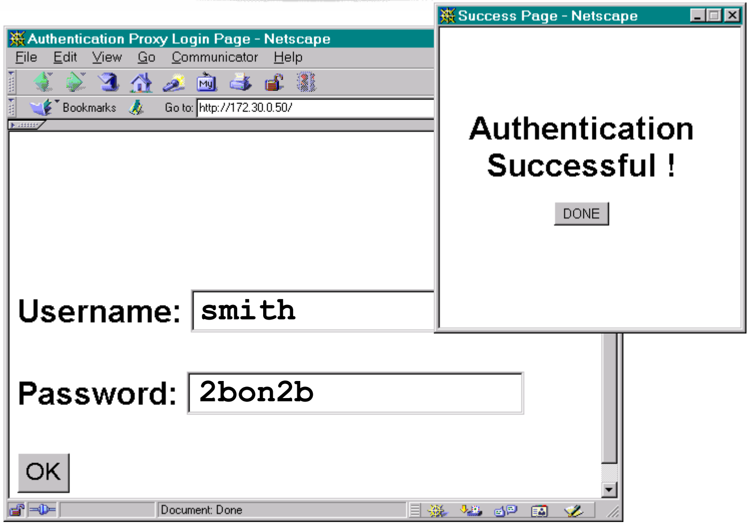
www.cisco.com CSPFA 2.1—19-4

The Cisco IOS Firewall authentication proxy feature enables network administrators to apply specific security policies on a per-user basis. Previously, user identity and related authorized access were associated with a user's IP address, or a single security policy had to be applied to an entire user group or subnet. Now, users can be identified and authorized on the basis of their per-user policy, and access privileges can be tailored on an individual basis, as opposed to a general policy applied across multiple users.

With the authentication proxy feature, users can log into the network or access the Internet via HTTP, and their specific access profiles are automatically retrieved and applied from a Cisco Secure Access Control Server (CSACS), or other Remote Access Dial-In User Service (RADIUS) or Terminal Access Controller Access Control System Plus (TACACS+) authentication server. The user profiles are active only when there is active traffic from the authenticated users.

The authentication proxy is compatible with other Cisco IOS security features such as Network Address Translation (NAT), Context-Based Access Control (CBAC), IP Security (IPSec) encryption, and virtual private network (VPN) client software.

## What the User Sees


Authentication Proxy Login Page - Netscape
File Edit View Go Communicator Help
Bookmarks
Go to: http://172.30.0.50/
OK
Document: Done
Success Page - Netscape
DONE

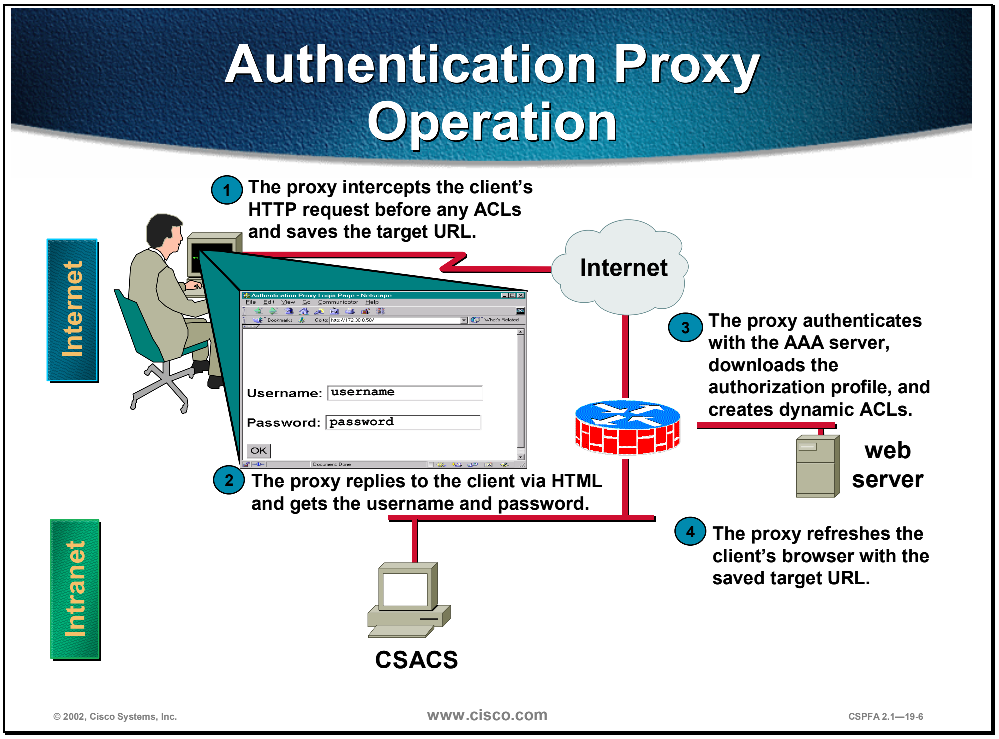
Username: smith

Password: 2bon2b

Authentication Successful !

www.cisco.com CSPFA 2.1—19-5

When a user initiates an HTTP session through the firewall, it triggers the authentication proxy. If a valid authentication entry exists for the user, the session is allowed and no further intervention is required by the authentication proxy. If no entry exists, the authentication proxy responds to the HTTP connection request by prompting the user for a username and password, as shown above.



# Authentication Proxy Operation

1 The proxy intercepts the client's HTTP request before any ACLs and saves the target URL.

2 The proxy replies to the client via HTML and gets the username and password.

3 The proxy authenticates with the AAA server, downloads the authorization profile, and creates dynamic ACLs.

4 The proxy refreshes the client's browser with the saved target URL.

© 2002, Cisco Systems, Inc. www.cisco.com CSPFA 2.1—19-6

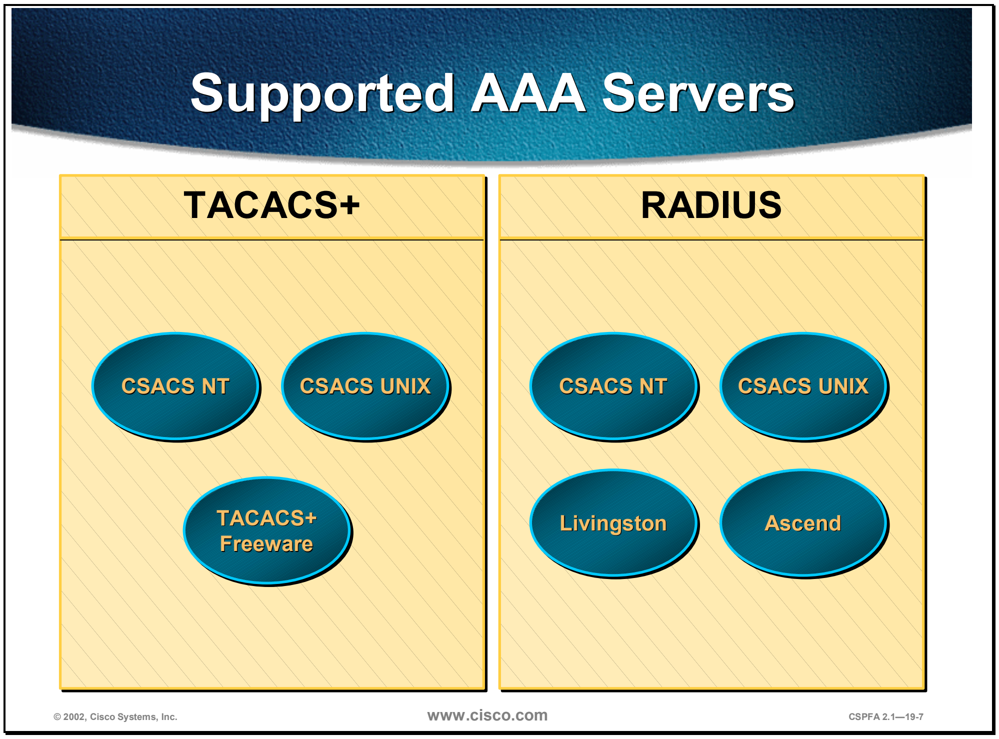
When a user initiates an HTTP session through the firewall, it triggers the authentication proxy. The authentication proxy first checks to see if the user has been authenticated. If a valid authentication entry exists for the user, the session is allowed and no further intervention is required by the authentication proxy. If no entry exists, the authentication proxy responds to the HTTP connection request by prompting the user for a username and password.

Users must successfully authenticate with the authentication server by entering a valid username and password. If the authentication succeeds, the user's authorization profile is retrieved from the authentication, authorization, and accounting (AAA) server. The authentication proxy uses the information in this profile to create dynamic access control entries (ACEs) and add them to the inbound (input) access control list (ACL) of an input interface, and to the outbound (output) ACL of an output interface if an output ACL exists at the interface. By doing this, the firewall allows authenticated users access to the network as permitted by the authorization profile. For example, a user can initiate a Telnet connection through the firewall if Telnet is permitted in the user's profile.

If the authentication fails, the authentication proxy reports the failure to the user and prompts the user with multiple retries. If the user fails to authenticate after five attempts, the user must wait two minutes and initiate another HTTP session to trigger the authentication proxy.

The authentication proxy sets up an inactivity (idle) timer for each user profile. As long as there is activity through the firewall, new traffic initiated from the user's host does not trigger the authentication proxy, and all authorized user traffic is permitted access through the firewall.

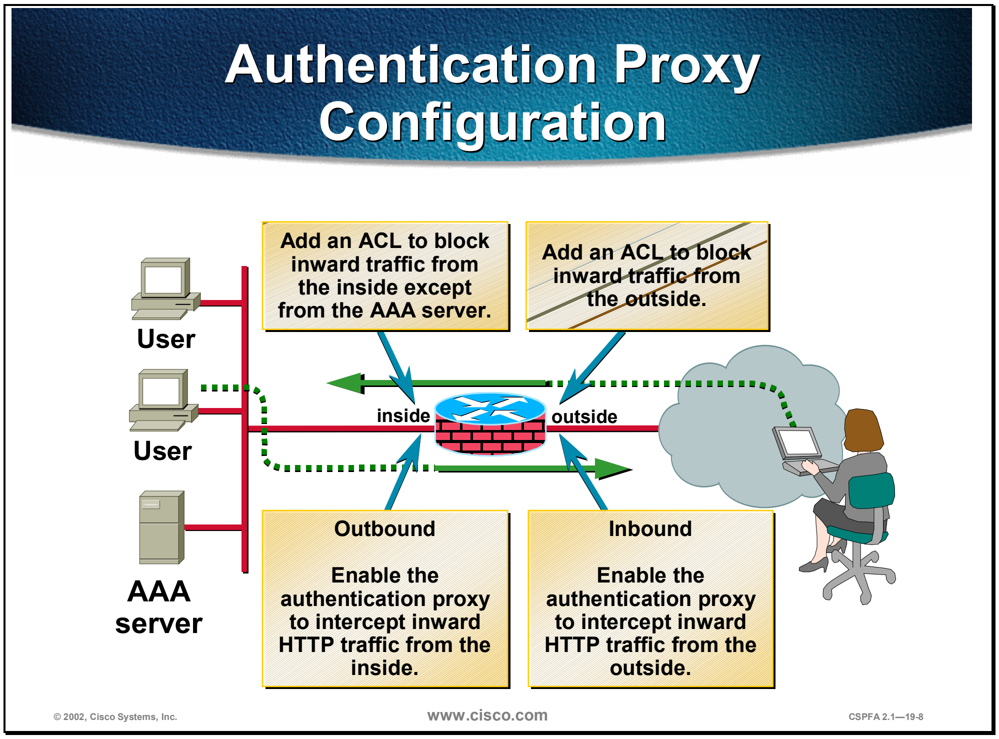
If the idle timer expires, the authentication proxy removes the user's profile information and dynamic ACL entries. When this happens, traffic from the client host is blocked. The user must initiate another HTTP connection to trigger the authentication proxy.



## Supported AAA Servers

The Cisco IOS Firewall authentication proxy supports the following AAA protocols and servers:

- Terminal Access Controller Access Control System Plus (TACACS+)
  - Cisco Secure Access Control Server (CSACS) for Windows NT (CSACS-NT)
  - CSACS for UNIX (CSACS-UNIX)
  - TACACS+ Freeware
- Remote Authentication Dial-In User Service (RADIUS)
  - CSACS for Windows NT (CSACS-NT)
  - CSACS for UNIX (CSACS-UNIX)
  - Livingston
  - Ascend



## Authentication Proxy Configuration

Add an ACL to block inward traffic from the inside except from the AAA server.

Add an ACL to block inward traffic from the outside.

**Outbound**

Enable the authentication proxy to intercept inward HTTP traffic from the inside.

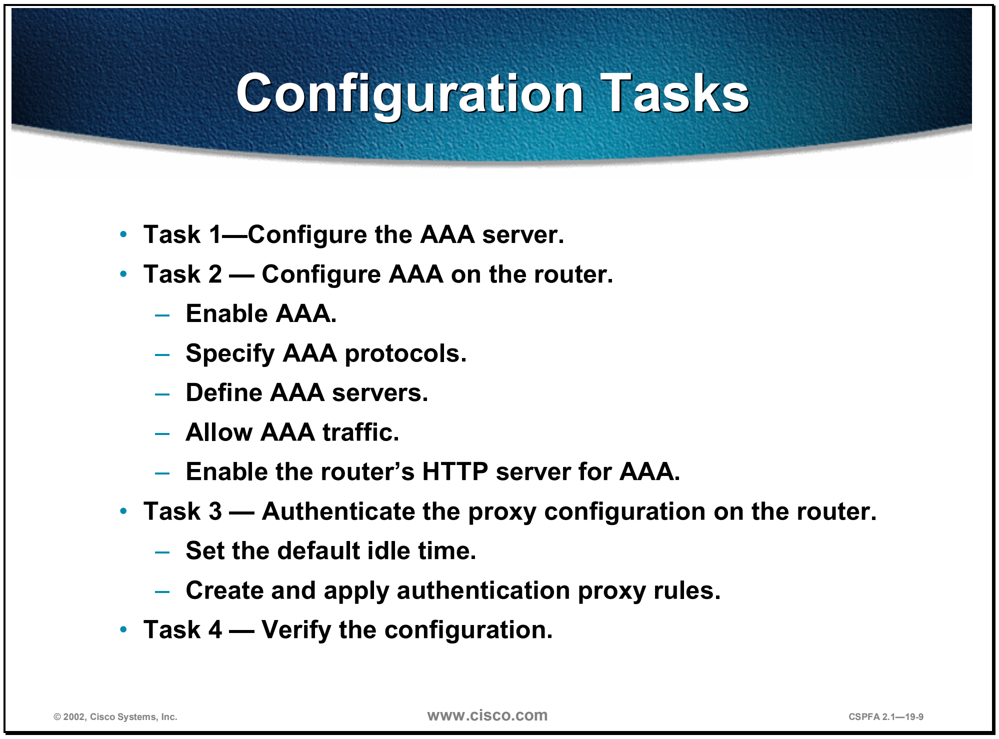
**Inbound**

Enable the authentication proxy to intercept inward HTTP traffic from the outside.

© 2002, Cisco Systems, Inc. www.cisco.com CSPFA 2.1—19-8

Apply the authentication proxy in the inward direction at any interface on the router where you want per-user authentication and authorization. Applying the authentication proxy inward at an interface causes it to intercept a user's initial connection request before that request is subjected to any other processing by the firewall. If the user fails to authenticate with the AAA server, the connection request is dropped.

How you apply the authentication proxy depends on your security policy. For example, you can block all traffic through an interface, and enable the authentication proxy feature to require authentication and authorization for all user-initiated HTTP connections. Users are authorized for services only after successful authentication with the AAA server. The authentication proxy feature also enables you to use standard ACLs to specify a host or group of hosts whose initial HTTP traffic triggers the proxy.



## Configuration Tasks

- Task 1—Configure the AAA server.
- Task 2 — Configure AAA on the router.
  - Enable AAA.
  - Specify AAA protocols.
  - Define AAA servers.
  - Allow AAA traffic.
  - Enable the router's HTTP server for AAA.
- Task 3 — Authenticate the proxy configuration on the router.
  - Set the default idle time.
  - Create and apply authentication proxy rules.
- Task 4 — Verify the configuration.

© 2002, Cisco Systems, Inc. www.cisco.com CSPFA 2.1—19-9

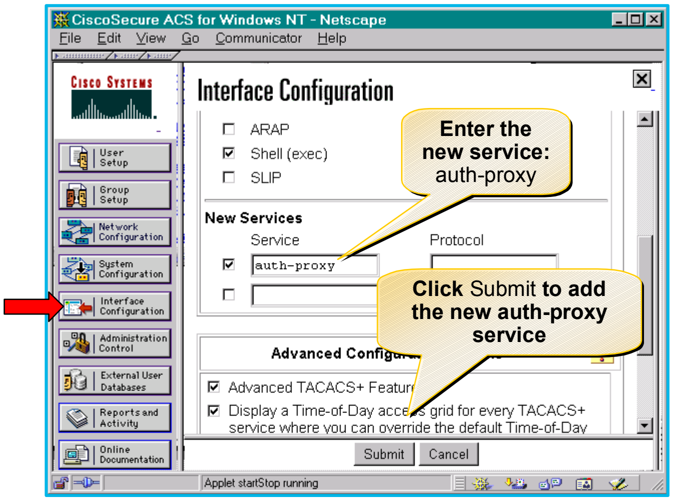
The following are the tasks to configure the authentication proxy:

- Task 1—Configure the AAA server.
- Task 2—Configure AAA on the router.
  - Enable AAA.
  - Specify AAA protocols.
  - Define AAA servers.
  - Allow AAA traffic.
  - Enable the router's HTTP server for AAA.
- Task 3—Authenticate the proxy configuration on the router.
  - Set the default idle time.
  - Create and apply authentication proxy rules.
- Task 4—Verify the configuration.

# AAA Server Configuration

This section discusses how to configure the AAA server to provide authentication and authorization for the Cisco IOS Firewall authorization proxy.

## Create auth-proxy Service in CSACS-NT



Interface Configuration

☐ ARAP
☑ Shell (exec)
☐ SLIP

New Services

| | Service | Protocol |
|---|---|---|
| ☑ | auth-proxy | |
| ☐ | | |

Advanced Configura...

☑ Advanced TACACS+ Featur...
☑ Display a Time-of-Day acce...s grid for every TACACS+ service where you can override the default Time-of-Day

Submit  Cancel

Enter the new service: auth-proxy

Click Submit to add the new auth-proxy service

© 2002, Cisco Systems, Inc.  www.cisco.com  CSPFA 2.1—19-11

To support the authentication proxy, configure the AAA authorization service auth-proxy on the AAA server. This creates a new section in the Group Setup frame in which user profiles can be created. This does not interfere with other type of services that the AAA server may have.

Complete the following steps to add authorization rules for specific services in CSACS:

**Step 1** In the navigation bar, click **Interface Configuration**. The Interface Configuration frame opens.

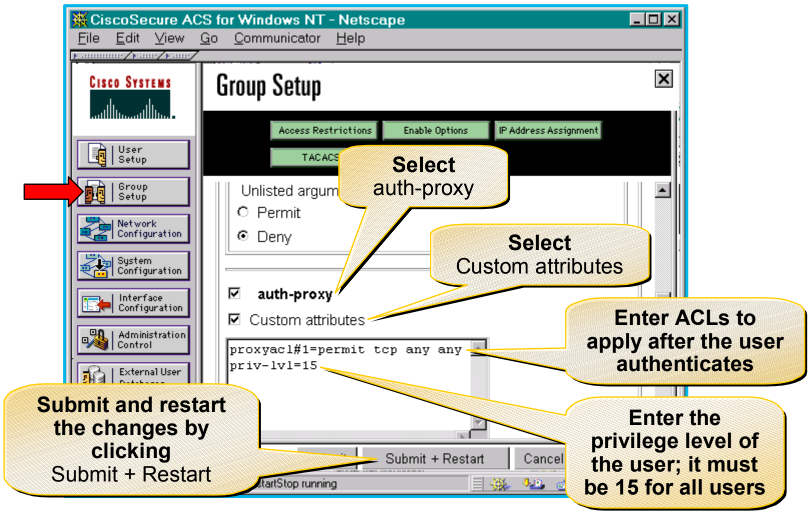
**Step 2** Scroll down in the Interface Configuration frame until you find the New Services group box.

**Step 3** Select the first check box in the Service column.

**Step 4** Enter **auth-proxy** in the first empty Service field next to the check box you just selected.

**Step 5** Click **Submit** when finished.

# Create a User Authorization Profile in CSACS-NT



Unlisted argum

○ Permit

● Deny

☑ auth-proxy

☑ Custom attributes

```
proxyacl#1=permit tcp any any
priv-lvl=15
```

**Select** auth-proxy

**Select** Custom attributes

**Enter ACLs to apply after the user authenticates**

**Enter the privilege level of the user; it must be 15 for all users**

**Submit and restart the changes by clicking** Submit + Restart

© 2002, Cisco Systems, Inc. www.cisco.com CSPFA 2.1—19-12

**Step 6** In the navigation bar, click **Group Setup**. The Group Setup frame opens.

**Step 7** Scroll down in the Group Setup frame until you find the newly created auth-proxy service.

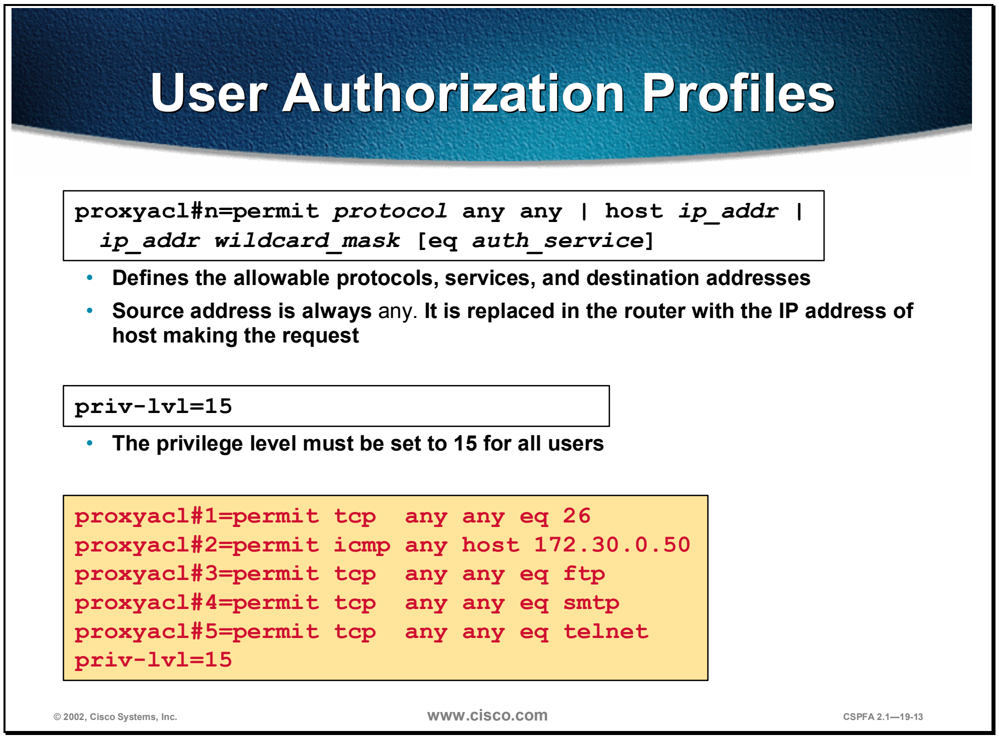
**Step 8** Select the **auth-proxy** check box.

**Step 9** Select the **Custom attributes** check box.

**Step 10** Using the proxyacl#n format described on the following page, enter ACLs in the field below the Custom attributes check box. These ACLs will be applied after the user authenticates.

**Step 11** Enter the privilege level of the user (must be 15 for all users) using the format from the following page.

**Step 12** Click **Submit** + **Restart** when finished.



## User Authorization Profiles

```
proxyacl#n=permit protocol any any | host ip_addr |
  ip_addr wildcard_mask [eq auth_service]
```

- **Defines the allowable protocols, services, and destination addresses**
- **Source address is always** any. **It is replaced in the router with the IP address of host making the request**

```
priv-lvl=15
```

- **The privilege level must be set to 15 for all users**

```
proxyacl#1=permit tcp  any any eq 26
proxyacl#2=permit icmp any host 172.30.0.50
proxyacl#3=permit tcp  any any eq ftp
proxyacl#4=permit tcp  any any eq smtp
proxyacl#5=permit tcp  any any eq telnet
priv-lvl=15
```

www.cisco.com CSPFA 2.1—19-13

Use the **proxyacl#n** attribute when configuring the ACLs in the profile. The **proxyacl#n** attribute is for both RADIUS and TACACS+ attribute-value pairs. The ACLs in the user profile on the AAA server must have **permit** access commands only. Set the source address to **any** in each of the user profile ACL entries. The source address in the ACLs is replaced with the source IP address of the host making the authentication proxy request when the user profile is downloaded to the firewall.

The following is the format for the ACLs used to enter in the Custom attributes box:

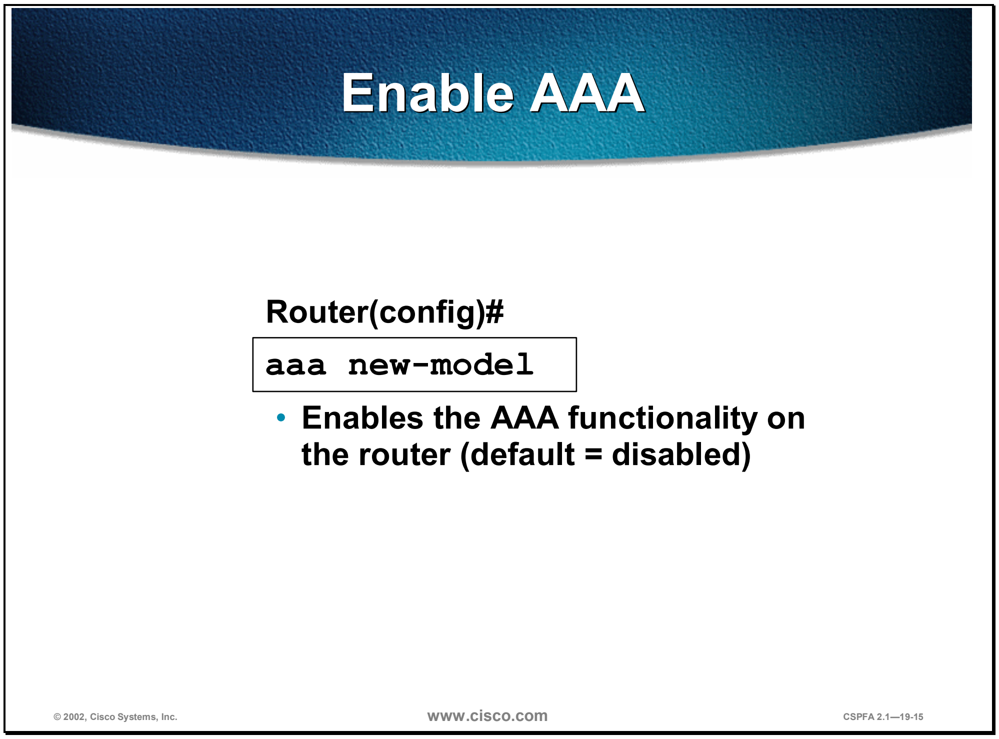
```
proxyacl#n=permit protocol any any | host ip_addr | ip_addr wildcard_mask [eq
auth_service]
```

| | |
|---|---|
| ***protocol*** | Keyword indicating the protocol to allow users to access: **tcp**, **udp**, or **icmp**. |
| **any** | Indicates any hosts. The first **any** after ***protocol*** is mandatory. This indicates any source IP address, which is actually replaced with the IP address of the user that requests authorization in the ACL applied in the router. |
| **host *ip_addr*** | IP address of a specific host users can access. |
| ***ip_addr wildcard mask*** | IP address and wildcard mask for a network that users can access. |
| **eq *auth_service*** | Specific service that users are allowed to access. |

Use **priv-lvl=15** to configure the privilege level of the authenticated user. The privilege level must be set to 15 for all users.

# AAA Configuration

This section discusses how to configure the Cisco IOS Firewall to work with an authentication, authorization, and accounting (AAA) server and enable the authentication proxy feature.



## Enable AAA

Router(config)#

```
aaa new-model
```

- **Enables the AAA functionality on the router (default = disabled)**

www.cisco.com

CSPFA 2.1—19-15

Use the **aaa new-model** global configuration command to enable the AAA access control system. Use the **no** form of this command to disable the AAA access control model.

---

**Note** After you have enabled AAA, TACACS and extended TACACS commands are no longer available. If you initialize AAA functionality and later decide to use TACACS or extended TACACS, issue the **no** version of this command and then enable the version of TACACS that you want to use.

---

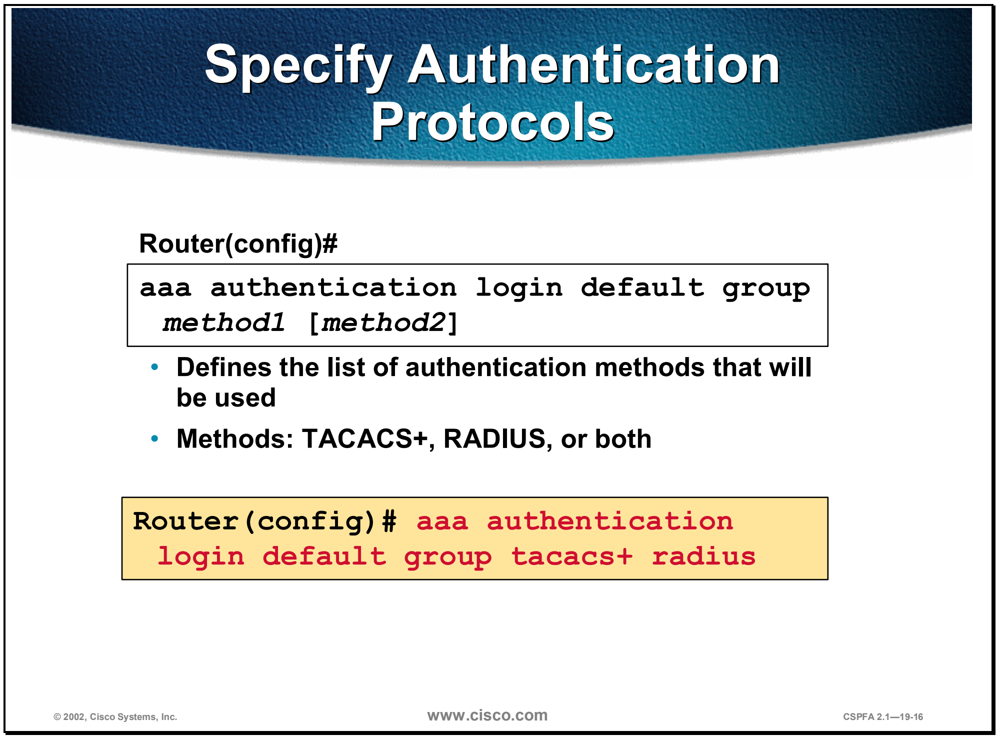
The syntax of the **aaa new-model** command is as follows:

**aaa new-model**

**no aaa new-model**

This command has no arguments.

By default, **aaa new-model** is not enabled.



# Specify Authentication Protocols

Router(config)#

```
aaa authentication login default group
  method1 [method2]
```

- Defines the list of authentication methods that will be used
- Methods: TACACS+, RADIUS, or both

```
Router(config)# aaa authentication
  login default group tacacs+ radius
```

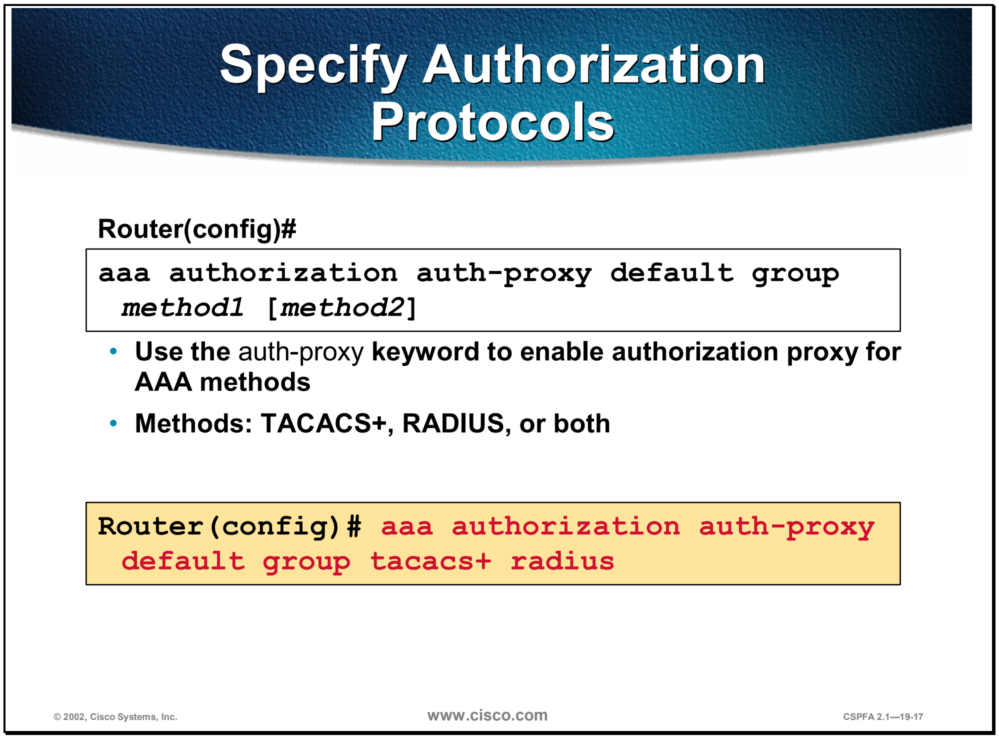
To set AAA authentication, use the **aaa authentication login** global configuration command. Use the **no** form of this command to disable AAA authentication.

The syntax of the **aaa authentication login** command is as follows:

**aaa authentication login default group** ***method1*** **[*****method2*****]**

**no aaa authentication login default group** ***method1*** **[*****method2*****]**

| | |
|---|---|
| ***method1***, ***method2*** | The following are the authentication protocols to use: **tacacs+**, **radius**, or both. |



## Specify Authorization Protocols

Router(config)#

```
aaa authorization auth-proxy default group
  method1 [method2]
```

- **Use the** auth-proxy **keyword to enable authorization proxy for AAA methods**
- **Methods: TACACS+, RADIUS, or both**

```
Router(config)# aaa authorization auth-proxy
  default group tacacs+ radius
```

© 2002, Cisco Systems, Inc. www.cisco.com CSPFA 2.1—19-17

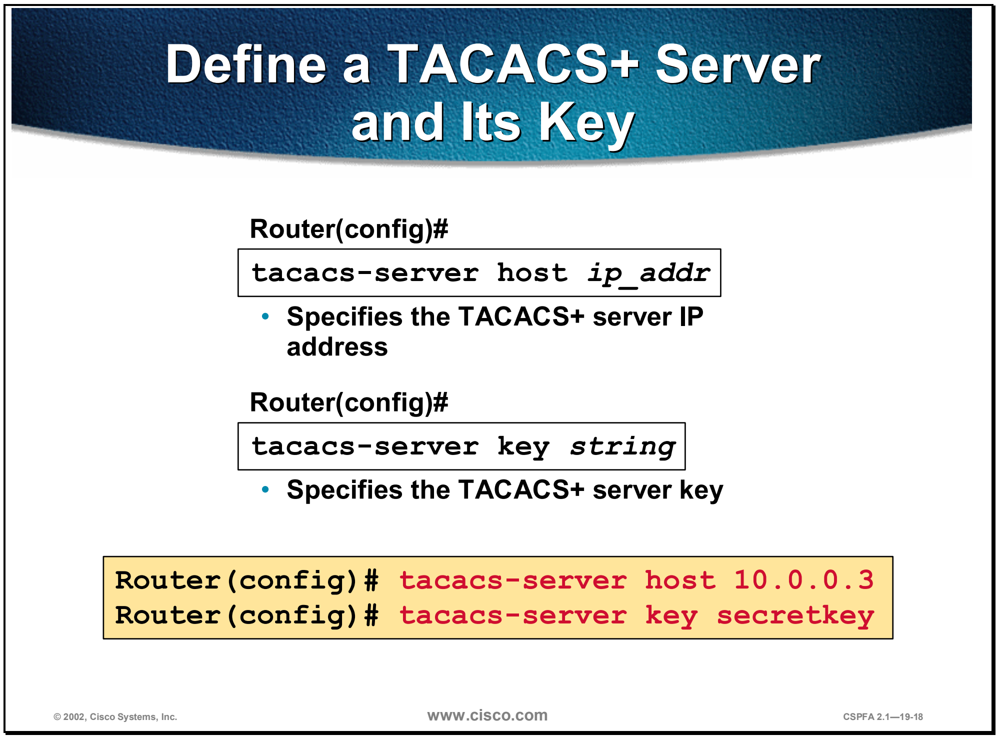
To set AAA authorization, use the **aaa authorization auth-proxy** global configuration command. Use the **no** form of this command to disable AAA authorization.

The syntax of the **aaa authorization auth-proxy** command is as follows:

**aaa authorization auth-proxy default group** *method1* **[***method2***]**

**no aaa authorization auth-proxy default group** *method1* **[***method2***]**

| | |
|---|---|
| ***method1***, ***method2*** | The following are the authorization protocols to use: **tacacs+**, **radius**, or both. |



## Define a TACACS+ Server and Its Key

Router(config)#

```
tacacs-server host ip_addr
```

- Specifies the TACACS+ server IP address

Router(config)#

```
tacacs-server key string
```

- Specifies the TACACS+ server key

```
Router(config)# tacacs-server host 10.0.0.3
Router(config)# tacacs-server key secretkey
```

© 2002, Cisco Systems, Inc. www.cisco.com CSPFA 2.1—19-18

To specify the IP address of a TACACS+ server, use the **tacacs-server host** global configuration command. Use the **no** form of this command to delete the specified IP address. You can use multiple **tacacs-server host** commands to specify additional servers. The Cisco IOS Firewall software searches for servers in the order in which you specify them.

The syntax of the **tacacs-server host** command is as follows:

**tacacs-server host** ***ip_addr***

**no tacacs-server host** ***ip_addr***

| | |
|---|---|
| ***ip_addr*** | IP address of the TACACS+ server. |

To set the authentication encryption key used for all TACACS+ communications between the Cisco IOS Firewall router and the AAA server, use the **tacacs-server key** global configuration command. Use the **no** form of this command to disable the key.

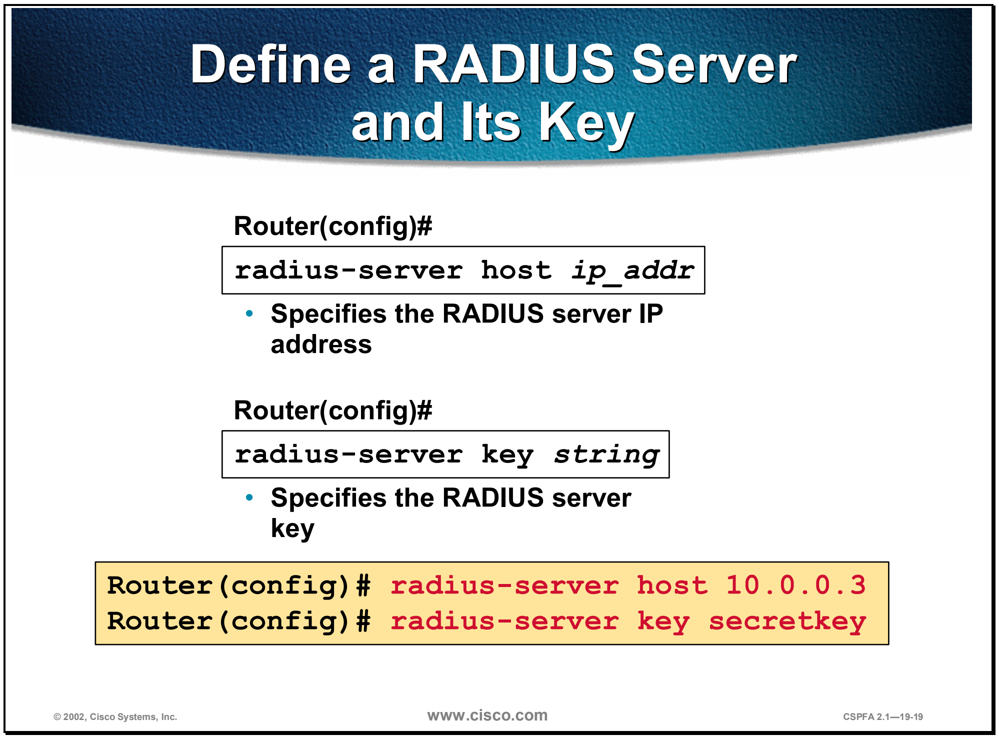
**Note** The key entered must match the key used on the AAA server. All leading spaces are ignored; spaces within and at the end of the key are not. If you use spaces in your key, do not enclose the key in quotation marks unless the quotation marks themselves are part of the key.

The syntax of the **tacacs-server key** command is as follows:

**tacacs-server key** ***string***

**no tacacs-server key** ***string***

| | |
|---|---|
| ***string*** | Key used for authentication and encryption. |



## Define a RADIUS Server and Its Key

Router(config)#

```
radius-server host ip_addr
```

- Specifies the RADIUS server IP address

Router(config)#

```
radius-server key string
```

- Specifies the RADIUS server key

```
Router(config)# radius-server host 10.0.0.3
Router(config)# radius-server key secretkey
```

© 2002, Cisco Systems, Inc. www.cisco.com CSPFA 2.1—19-19

To specify the IP address of a RADIUS server, use the **radius-server host** global configuration command. Use the **no** form of this command to delete the specified IP address. You can use multiple **radius-server host** commands to specify additional servers. The Cisco IOS Firewall software searches for servers in the order in which you specify them.

The syntax of the **radius-server host** command is as follows:

**radius-server host** ***ip_addr***

**no radius-server host** ***ip_addr***

| | |
|---|---|
| ***ip_addr*** | IP address of the RADIUS server. |

To set the authentication encryption key used for all RADIUS communications between the Cisco IOS Firewall router and the AAA server, use the **radius-server key** global configuration command. Use the **no** form of this command to disable the key.

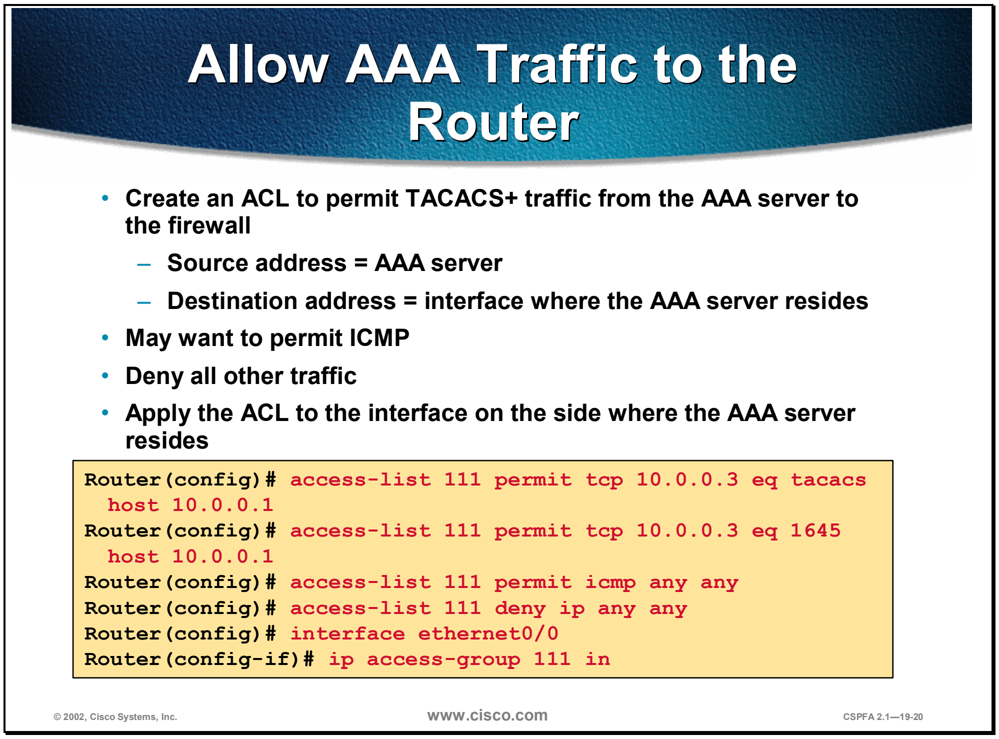
**Note** The key entered must match the key used on the AAA server. All leading spaces are ignored; spaces within and at the end of the key are not. If you use spaces in your key, do not enclose the key in quotation marks unless the quotation marks themselves are part of the key.

The syntax of the **radius-server key** command is as follows:

**radius-server key *string***

**no radius-server key *string***

| | |
|---|---|
| ***string*** | Key used for authentication and encryption. |



## Allow AAA Traffic to the Router

- Create an ACL to permit TACACS+ traffic from the AAA server to the firewall
  - Source address = AAA server
  - Destination address = interface where the AAA server resides
- May want to permit ICMP
- Deny all other traffic
- Apply the ACL to the interface on the side where the AAA server resides

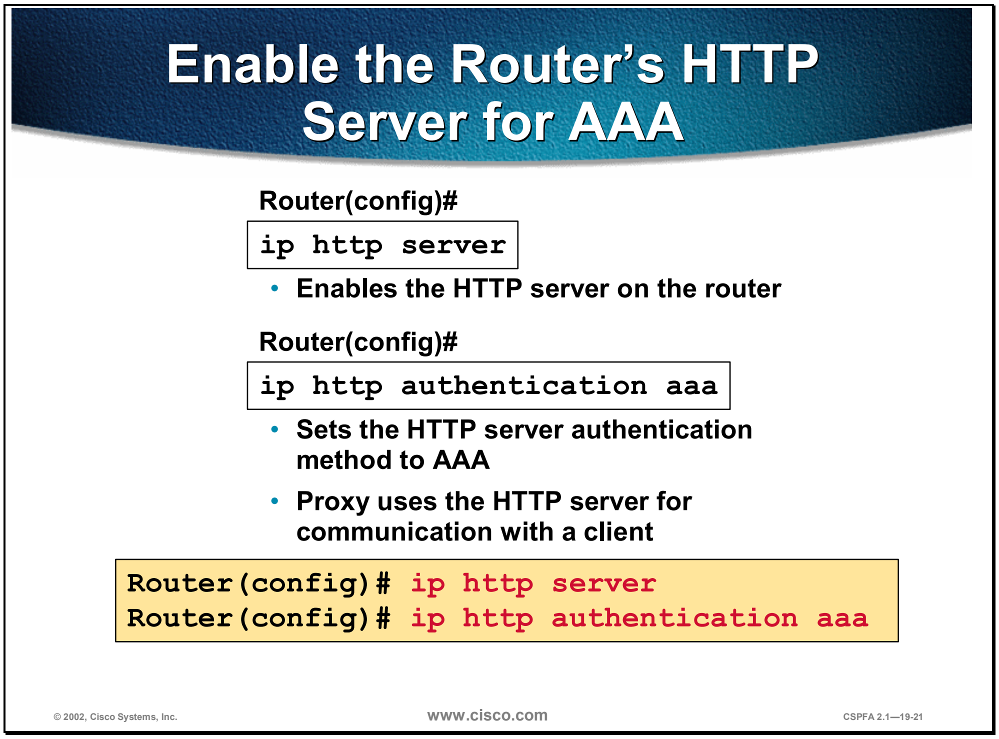
```
Router(config)# access-list 111 permit tcp 10.0.0.3 eq tacacs
  host 10.0.0.1
Router(config)# access-list 111 permit tcp 10.0.0.3 eq 1645
  host 10.0.0.1
Router(config)# access-list 111 permit icmp any any
Router(config)# access-list 111 deny ip any any
Router(config)# interface ethernet0/0
Router(config-if)# ip access-group 111 in
```

© 2002, Cisco Systems, Inc. www.cisco.com CSPFA 2.1—19-20

At this point you need to configure and apply an ACL to permit TACACS+ and RADIUS traffic from the AAA server to the firewall.

Use the following guidelines when writing the ACL:

- The source address is AAA server
- The destination address is interface where the AAA server resides
- You may want to permit ICMP
- Deny all other traffic
- Apply the ACL to the interface on the side where the AAA server resides



## Enable the Router's HTTP Server for AAA

Router(config)#

```
ip http server
```

- Enables the HTTP server on the router

Router(config)#

```
ip http authentication aaa
```

- Sets the HTTP server authentication method to AAA
- Proxy uses the HTTP server for communication with a client

```
Router(config)# ip http server
Router(config)# ip http authentication aaa
```

www.cisco.com

CSPFA 2.1—19-21

To use the authentication proxy, use the **ip http server** command to enable the HTTP server on the router and the **ip http authentication aaa** command to make the HTTP server use AAA for authentication.

The syntax of the **ip http server** command is as follows:

**ip http server**

This command has no arguments.

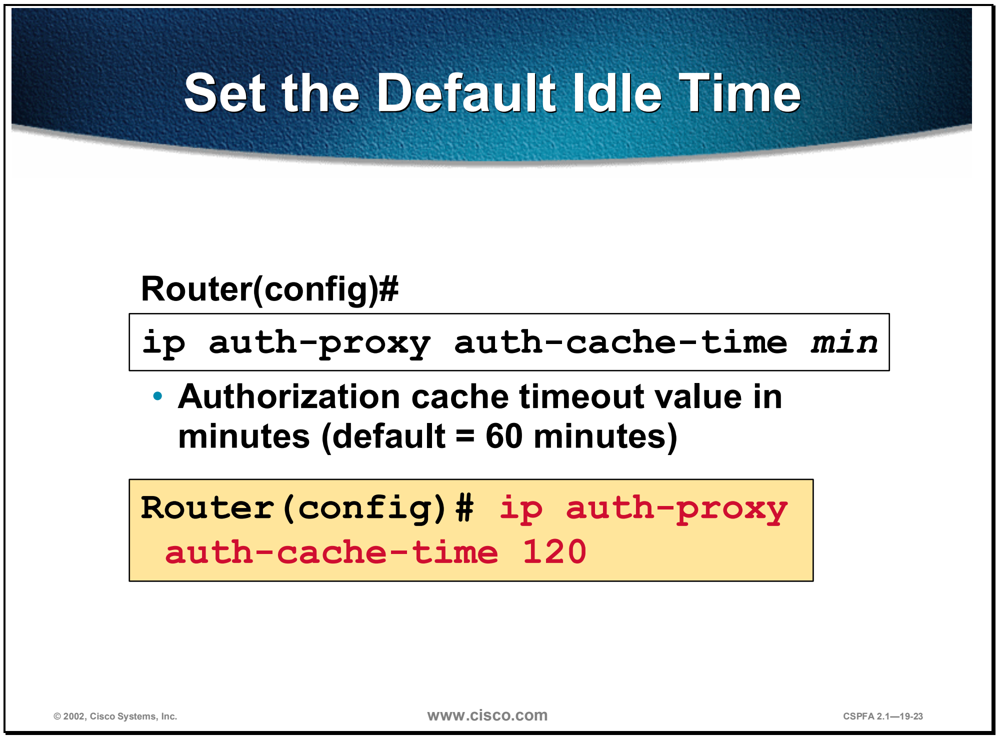
The syntax of the **ip http authentication aaa** command is as follows:

**ip http authentication aaa**

This command has no arguments.

# Authentication Proxy Configuration

This section discusses how to configure the authentication proxy settings on a Cisco router.



## Set the Default Idle Time

Router(config)#

```
ip auth-proxy auth-cache-time min
```

- Authorization cache timeout value in minutes (default = 60 minutes)

```
Router(config)# ip auth-proxy
 auth-cache-time 120
```

www.cisco.com

CSPFA 2.1—19-23

To set the authentication proxy idle timeout value (the length of time an authentication cache entry, along with its associated dynamic user ACL, is managed after a period of inactivity), use the **ip auth-proxy auth-cache-time** global configuration command. To set the default value, use the **no** form of this command.

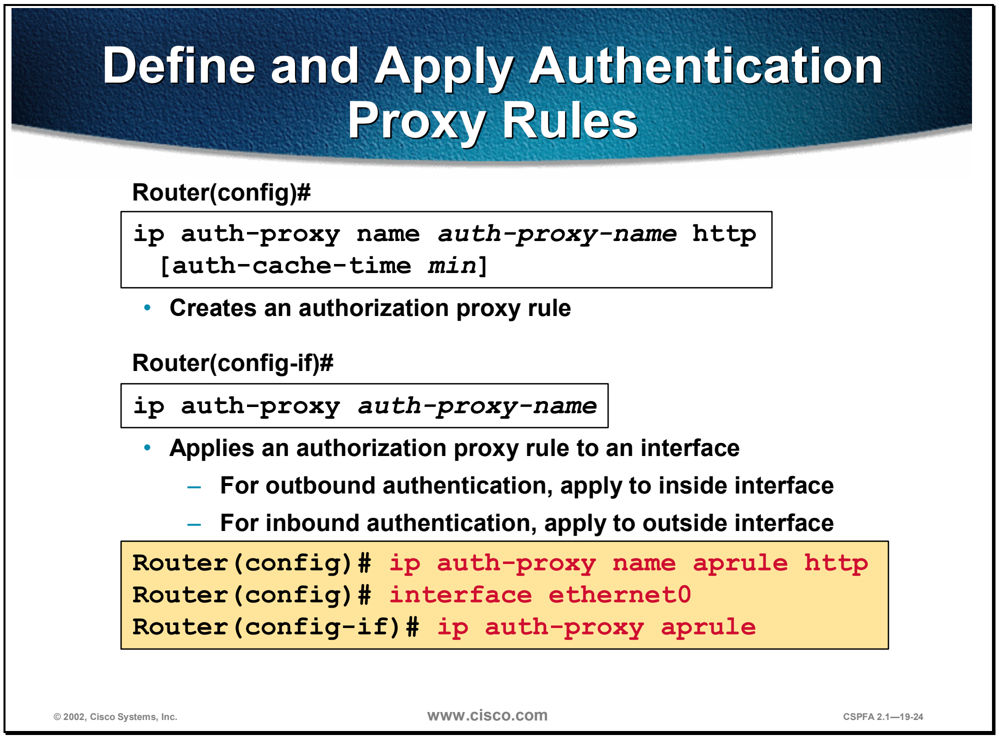
**Note** Set the **auth-cache-time** option for any authentication proxy rule to a higher value than the idle timeout value for any CBAC inspection rule. When the authentication proxy removes an authentication cache along with its associated dynamic user ACL, there might be some idle connections monitored by CBAC, and removal of user-specific ACLs could cause those idle connections to hang. If CBAC has a shorter idle timeout, CBAC resets these connections when the idle timeout expires; that is, before the authentication proxy removes the user profile.

The syntax of the **ip auth-proxy auth-cache-time** command is as follows:

**ip auth-proxy auth-cache-time** ***min***

**no ip auth-proxy auth-cache-time**

| | |
|---|---|
| ***min*** | Specifies the length of time, in minutes that an authentication cache entry, along with its associated dynamic user ACL, is managed after a period of inactivity. Enter a value in the range of 1 to 2,147,483,647. The default value is 60 minutes. |



## Define and Apply Authentication Proxy Rules

Router(config)#

```
ip auth-proxy name auth-proxy-name http
  [auth-cache-time min]
```

- Creates an authorization proxy rule

Router(config-if)#

```
ip auth-proxy auth-proxy-name
```

- Applies an authorization proxy rule to an interface
  - For outbound authentication, apply to inside interface
  - For inbound authentication, apply to outside interface

```
Router(config)# ip auth-proxy name aprule http
Router(config)# interface ethernet0
Router(config-if)# ip auth-proxy aprule
```

© 2002, Cisco Systems, Inc. www.cisco.com CSPFA 2.1—19-24

To create an authentication proxy rule, use the **ip auth-proxy name** global configuration command. To remove the authentication proxy rules, use the **no** form of this command.

The syntax of the **ip auth-proxy name** command is as follows:

**ip auth-proxy name *auth-proxy-name* http [auth-cache-time *min*]**

**no ip auth-proxy name *auth-proxy-name***

| | |
|---|---|
| ***auth-proxy-name*** | Associates a name with an authentication proxy rule. Enter a name of up to 16 alphanumeric characters. |
| **auth-cache-time *min*** | (Optional) Overrides the global authentication proxy cache timer for a specific authentication proxy name, offering more control over timeout values. Enter a value in the range of 1 to 2,147,483,647. The default value is equal to the value set with the **ip auth-proxy auth-cache-time** command. |

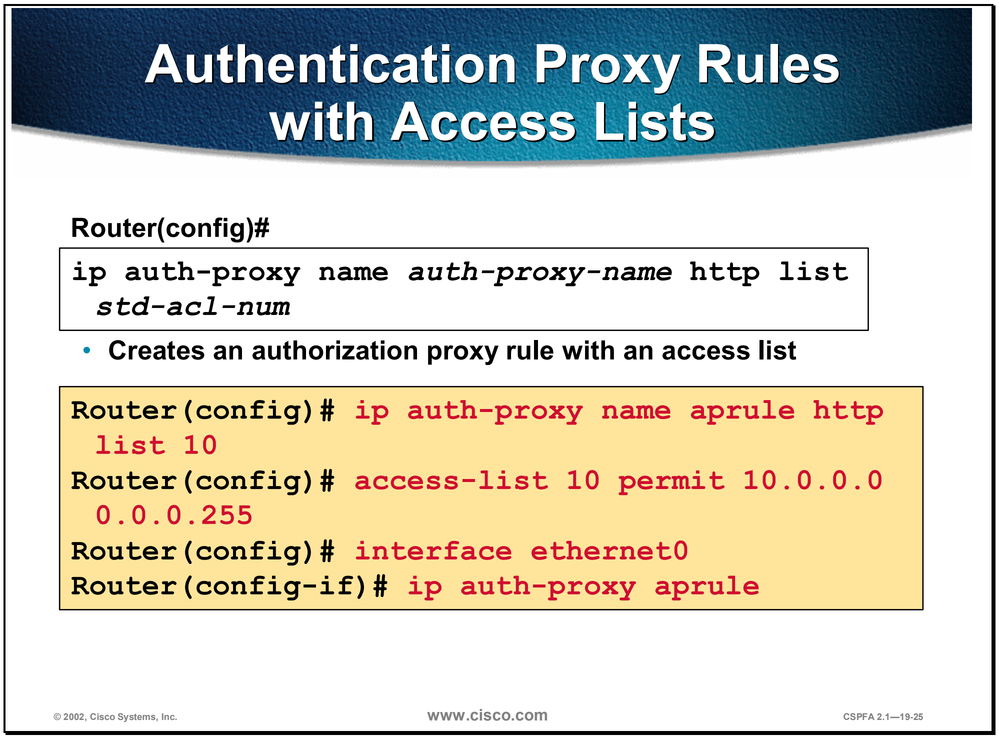
To apply an authentication proxy rule at a firewall interface, use the **ip auth-proxy** interface configuration command. To remove the authentication proxy rules, use the **no** form of this command.

The syntax of the **ip auth-proxy** command is as follows:

**ip auth-proxy *auth-proxy-name***

**no ip auth-proxy *auth-proxy-name***

| | |
|---|---|
| ***auth-proxy-name*** | Specifies the name of the authentication proxy rule to apply to the interface configuration. The authentication proxy rule is established with the **authentication proxy name** command. |



# Authentication Proxy Rules with Access Lists

Router(config)#

```
ip auth-proxy name auth-proxy-name http list
  std-acl-num
```

- Creates an authorization proxy rule with an access list

```
Router(config)# ip auth-proxy name aprule http
  list 10
Router(config)# access-list 10 permit 10.0.0.0
  0.0.0.255
Router(config)# interface ethernet0
Router(config-if)# ip auth-proxy aprule
```

© 2002, Cisco Systems, Inc. www.cisco.com CSPFA 2.1—19-25

You can associate an authentication proxy rule with an ACL, providing control over which hosts use the authentication proxy. To create an authentication proxy rule with ACLs, use the **ip auth-proxy name** global configuration command with the **list** ***std-acl-num*** option. To remove the authentication proxy rules, use the **no** form of this command.

The syntax of the **ip auth-proxy name** with ACLs command is as follows:

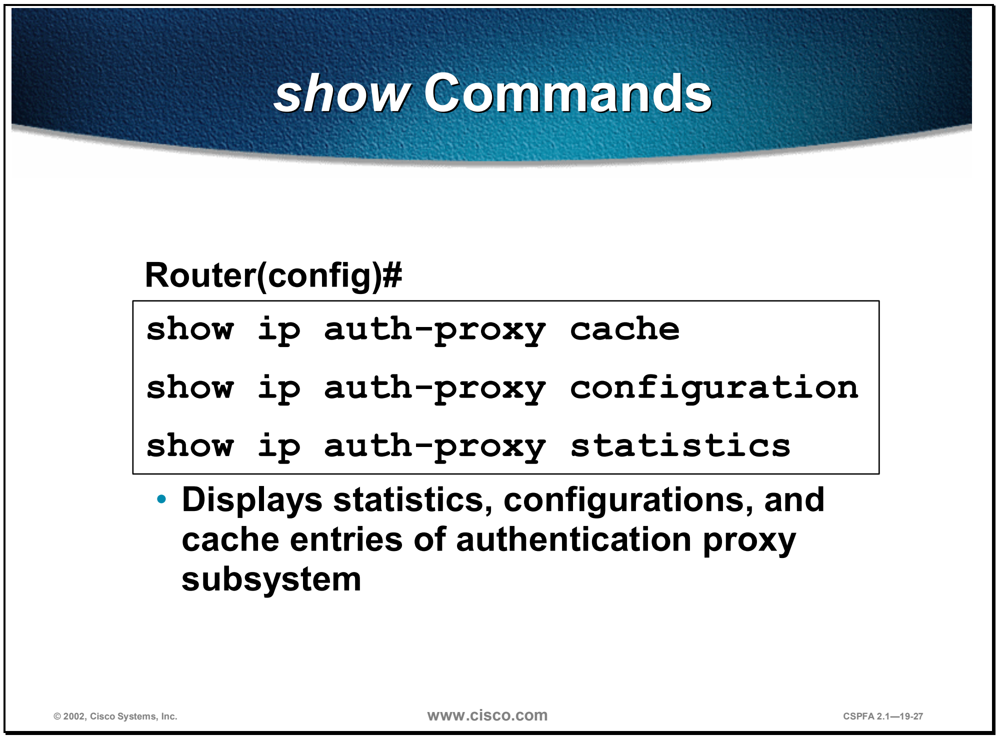
**ip auth-proxy name *auth-proxy-name* http list *std-acl-num***

**no ip auth-proxy name *auth-proxy-name***

| | |
|---|---|
| ***auth-proxy-name*** | Associates a name with an authentication proxy rule. Enter a name of up to 16 alphanumeric characters. |
| **list *std-acl-num*** | Specifies a standard ACL to use with the authentication proxy. With this option, the authentication proxy is applied only to those hosts in the standard ACL If no list is specified, all connections initiating HTTP traffic arriving at the interface are subject to authentication. |

# Test and Verify the Configuration

This section discusses the procedures for testing and verifying the authentication proxy configuration.



## *show* Commands

Router(config)#

```
show ip auth-proxy cache
show ip auth-proxy configuration
show ip auth-proxy statistics
```

- **Displays statistics, configurations, and cache entries of authentication proxy subsystem**

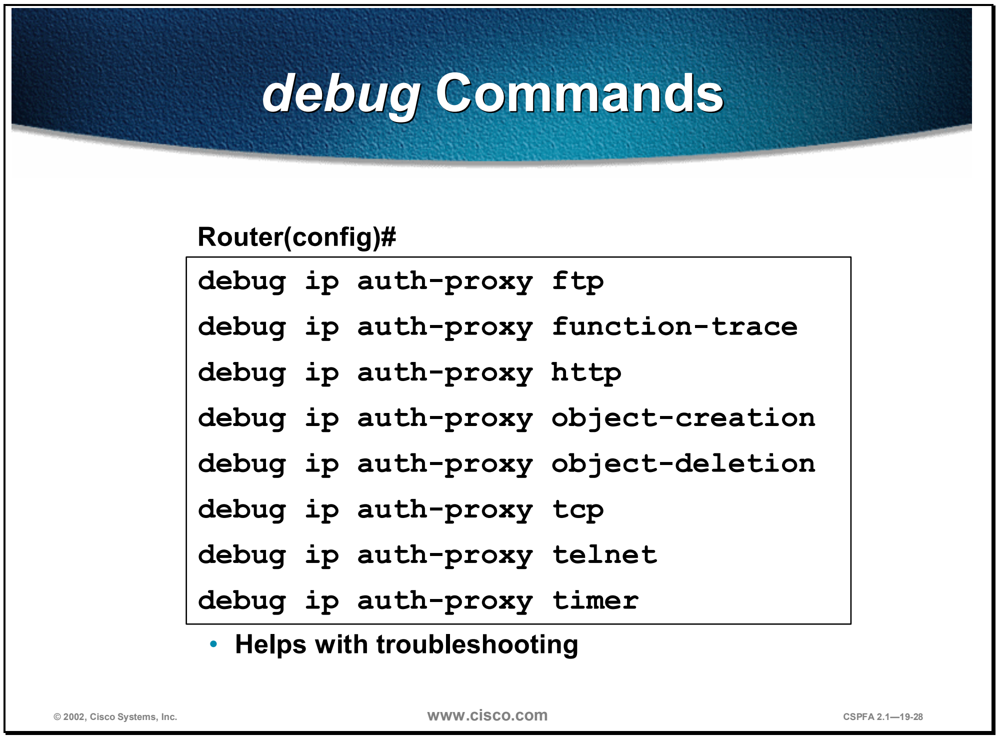
© 2002, Cisco Systems, Inc. www.cisco.com CSPFA 2.1—19-27

Use the **show ip auth-proxy** command to display the authentication proxy entries, the running authentication proxy configuration, or the authentication proxy statistics.

The syntax of the **show ip auth-proxy** command is as follows:

**show ip auth-proxy cache** | **configuration** | **statistics**

| | |
|---|---|
| **cache** | Lists the host IP address, the source port number, the timeout value for the authentication proxy, and the state for connections using authentication proxy. If the authentication proxy state is HTTP_ESTAB, the user authentication was successful. |
| **configuration** | Displays all authentication proxy rules configured on the router. |
| **statistics** | Displays all the router statistics related to the authentication proxy. |



## *debug* Commands

Router(config)#

```
debug ip auth-proxy ftp
debug ip auth-proxy function-trace
debug ip auth-proxy http
debug ip auth-proxy object-creation
debug ip auth-proxy object-deletion
debug ip auth-proxy tcp
debug ip auth-proxy telnet
debug ip auth-proxy timer
```

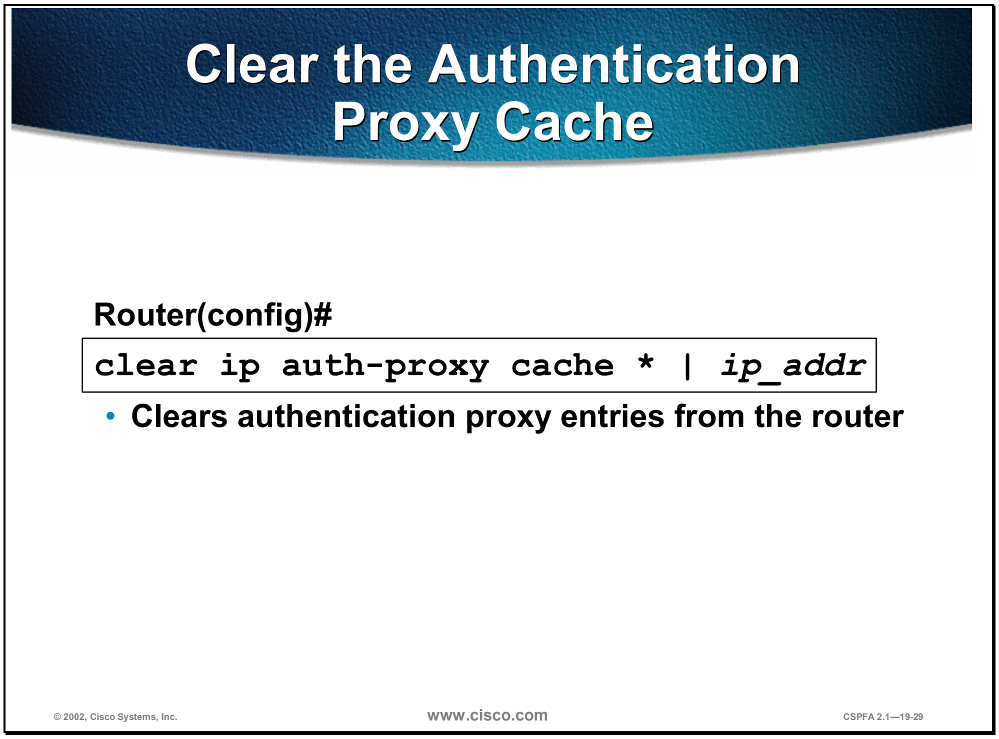
- **Helps with troubleshooting**

© 2002, Cisco Systems, Inc. www.cisco.com CSPFA 2.1—19-28

The syntax of the **debug ip auth-proxy** command is as follows:

**debug ip auth-proxy ftp** | **function-trace** | **http** | **object-creation** | **object-deletion** | **tcp** | **telnet** | **timer**

| | |
|---|---|
| **ftp** | Displays FTP events related to the authentication proxy. |
| **function-trace** | Displays the authentication proxy functions. |
| **http** | Displays HTTP events related to the authentication proxy. |
| **object-creation** | Displays additional entries to the authentication proxy cache. |
| **object-deletion** | Displays deletion of cache entries for the authentication proxy. |
| **tcp** | Displays TCP events related to the authentication proxy. |
| **telnet** | Displays Telnet-related authentication proxy events. |
| **timer** | Displays authentication proxy timer-related events. |



## Clear the Authentication Proxy Cache

Router(config)#

```
clear ip auth-proxy cache * | ip_addr
```

- Clears authentication proxy entries from the router

© 2002, Cisco Systems, Inc. www.cisco.com CSPFA 2.1—19-29

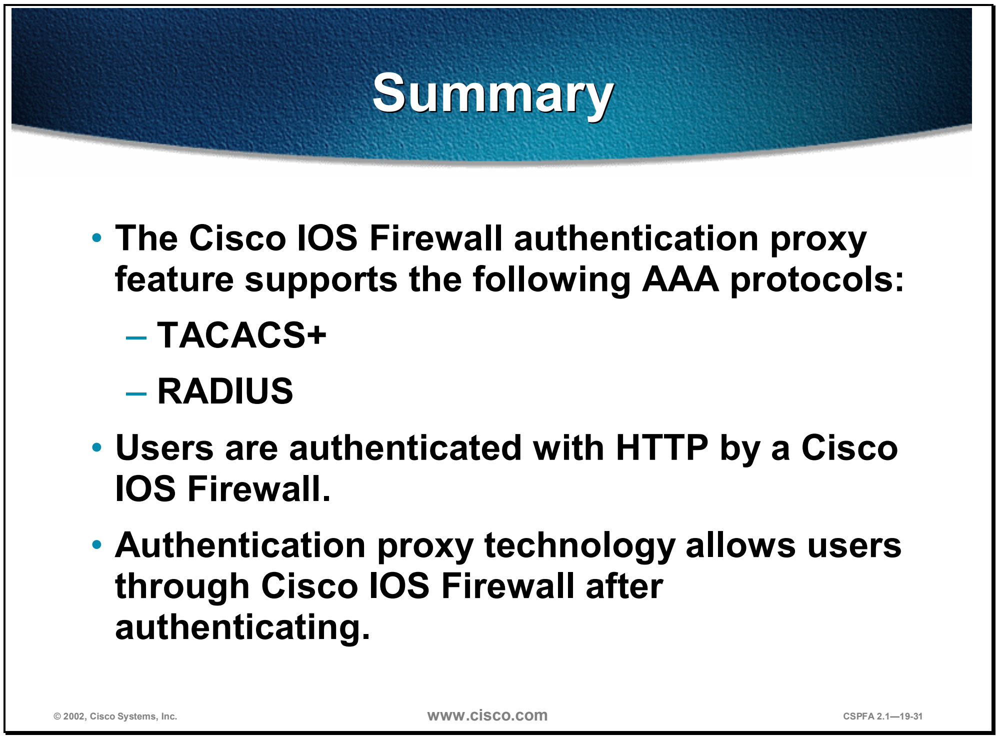
The syntax of the **clear ip auth-proxy cache** command is as follows:

**clear ip auth-proxy cache * |** ***ip_addr***

| | |
|---|---|
| ***** | Clears all authentication proxy entries, including user profiles and dynamic ACLs. |
| ***ip_addr*** | Clears the authentication proxy entry, including user profiles and dynamic ACLs, for the specified IP address. |

# Summary

This section summarizes what you have learned in this chapter.



## Summary

- The Cisco IOS Firewall authentication proxy feature supports the following AAA protocols:
  - TACACS+
  - RADIUS
- Users are authenticated with HTTP by a Cisco IOS Firewall.
- Authentication proxy technology allows users through Cisco IOS Firewall after authenticating.

© 2002, Cisco Systems, Inc. www.cisco.com CSPFA 2.1—19-31

# Lab Exercise—Configure Authentication Proxy on a Cisco Router

Complete the following lab to practice what you learned in this chapter.

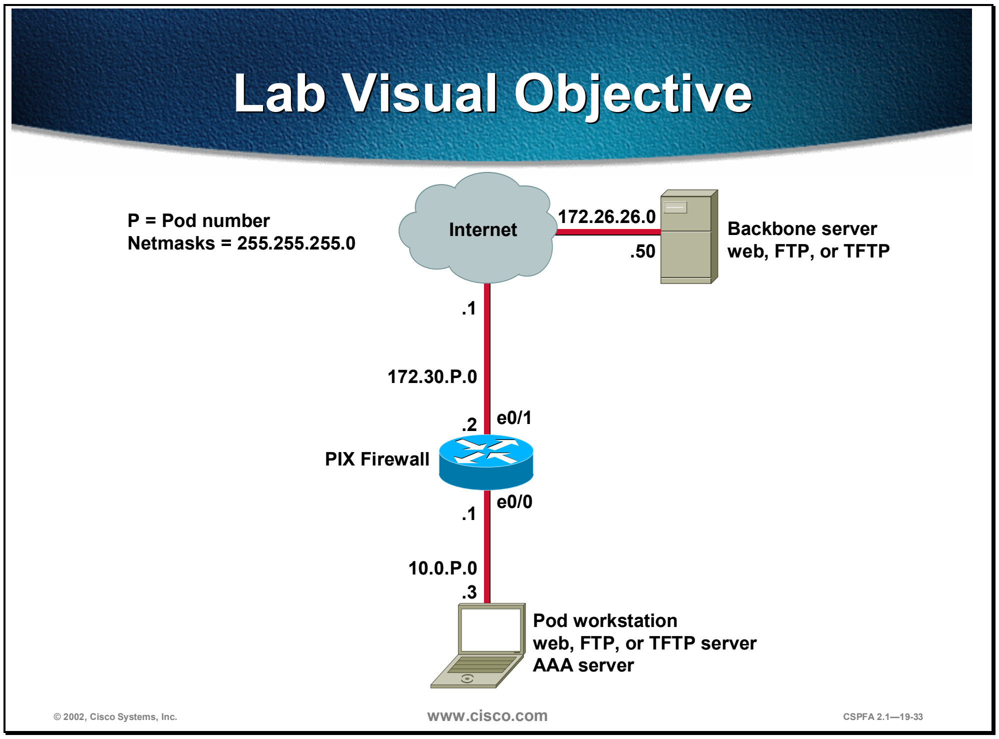
## Objectives

In this lab exercise you will complete the following tasks:

- Configure CSACS NT.
- Configure AAA.
- Configure an authentication proxy.
- Test and verify an authentication proxy.

## Lab Visual Objective

The following figure displays the configuration you will use to complete in this lab exercise.



Lab Visual Objective

© 2002, Cisco Systems, Inc. www.cisco.com CSPFA 2.1—19-33

### Directions

Your task in this exercise is to configure authentication proxy on a Cisco router. Work with your lab partner to perform the following steps in this lab exercise:

- Task 1—Configure CSACS NT

- Task 2—Configure AAA
- Task 3—Configure an Authentication Proxy
- Task 4—Test and Verify an Authentication Proxy

## Task 1—Configure CSACS NT

To configure CSACS NT, complete the following steps:

**Step 1** On your workstation, open Cisco Secure ACS from the desktop.

**Step 2** Click **Interface Configuration** on the far left column of CSACS to go to the Interface Configuration window.

**Step 3** Click **TACACS+ (Cisco)** to configure this option.

**Step 4** Scroll down until you find New Services.

**Step 5** Select the first line under New Service and enter **auth-proxy** under service.

**Step 6** Choose **Advanced TACACS+ features**.

**Step 7** Click **Submit** to submit your changes.

**Step 8** Click **Group Setup** to open the Group Setup window.

**Step 9** Select **Default Group (1 user)** under the Group drop-down menu.

**Step 10** Click **Edit Settings** to go to the Group Setup for this group.

**Step 11** Scroll down until you find the auth-proxy check box followed by the Custom attributes check box. Check both the **auth-proxy** check box and **the Custom attributes** check box.

**Step 12** Enter the following in the Custom attributes box.

```
proxyacl#1=permit tcp any any
priv-lvl=15
```

**Step 13** Click **Submit + Restart** to submit your changes and restart CSACS. Wait for the interface to return to the Group Setup main window.

## Task 2—Configure AAA

To configure AAA, complete the following steps:

**Step 1** On your router, enter global configuration mode.

```
Router# configure terminal
```

**Step 2** Enable AAA.

```
Router(config)# aaa new-model
```

**Step 3** Specify the authentication protocol.

```
Router(config)# aaa authentication login default group tacacs+
```

**Step 4** Specify the authorization protocol.

```
Router(config)# aaa authorization auth-proxy default group tacacs+
```

**Step 5** Define the TACACS+ server and its key.

```
Router(config)# tacacs-server host 10.0.P.3 (P=pod num.)
Router(config)# tacacs-server key secretkey
```

**Step 6** Clear the previously applied ACL on the inside interface.

```
Router(config)# no access-list 101
```

**Step 7** Define a new ACL to allow TACACS+ traffic to the inside interface from your AAA server. Also allow outbound ICMP traffic and CBAC traffic (FTP and WWW). Block all other inside-initiated traffic.

```
Router(config)# access-list 101 permit tcp host 10.0.P.3 eq tacacs host 10.0.P.1
Router(config)# access-list 101 permit icmp any any
Router(config)# access-list 101 permit tcp 10.0.P.0 0.0.0.255 any eq ftp
Router(config)# access-list 101 permit tcp 10.0.P.0 0.0.0.255 any eq www
Router(config)# access-list 101 deny ip any any

(where P = pod number, and Q = peer pod number)
```

**Step 8** Enable the router's HTTP server for AAA.

```
Router(config)# ip http server
Router(config)# ip http authentication aaa
```

## Task 3—Configure an Authentication Proxy

To configure authentication proxy, complete the following steps:

**Step 1** Define an authentication proxy rule.

```
Router(config)# ip auth-proxy name APRULE http auth-cache-time 5
```

**Step 2** Apply the authentication proxy rule to the inside interface.

```
Router(config)# interface ethernet 0/0
Router(config-if)# ip auth-proxy APRULE
Router(config-if)# end
```

## Task 4—Test and Verify an Authentication Proxy

To test and verify authentication proxy, complete the following steps:

**Step 1** On your router, use the **show access-list** command to check your access lists. Fill in the blanks below using the output from this command.

```
Router# show access-list
Extended IP access list 101
```

___________________________________________________________

___________________________________________________________

___________________________________________________________

```
Extended IP access list 102
```

___________________________________________________________

___________________________________________________________

___________________________________________________________

___________________________________________________________

**Step 2** On your router, use the **show ip inspect** command to see CBAC sessions. Fill in the blanks below using the output from this command.

```
Router# show ip inspect sessions
```

___________________________________________________________

___________________________________________________________

**Step 3** Use the **show ip auth-proxy configuration** command to verify the authorization proxy configuration. Fill in the blanks below using the output from this command.

```
Router# show ip auth-proxy configuration
Authentication global cache time is _____ minutes
Authentication Proxy Rule Configuration
 Auth-proxy name ____________
    http list not specified auth-cache-time _____ minutes
```

**Step 4** Use the **show ip auth-proxy statistics** command to verify the authorization proxy statistics. Fill in the blanks below using the output from this command.

```
Router# show ip auth-proxy statistics
Authentication Proxy Statistics
    proxied client number _____
```

**Step 5** Use the **show ip auth-proxy cache** command to verify the authorization proxy configuration. Fill in the blanks below using the output from this command.

```
Router# show ip auth-proxy cache

__________________________________________________________________
```

**Step 6** From your workstation command prompt, ping the backbone server.

```
C:\> ping 172.26.26.50
Pinging 172.26.26.50 with 32 bytes of data:

Reply from 172.26.26.50: bytes=32 time=34ms TTL=125
Reply from 172.26.26.50: bytes=32 time=34ms TTL=125
Reply from 172.26.26.50: bytes=32 time=34ms TTL=125
Reply from 172.26.26.50: bytes=32 time=36ms TTL=125
```

**Step 7** Use your web browser to connect to the backbone web server. In the URL field enter the following.

```
http://172.26.26.50
```

**Step 8** Enter the following when the web browser prompts you for a username and password.

```
Username: aaauser
Password: aaapass
```

**Step 9** Use the **show access-list** command to check your ACLs. Fill in the blanks below using the output from this command.

```
Router# show access-list
Extended IP access list 101
__________________________________________________________________
__________________________________________________________________
__________________________________________________________________
Extended IP access list 102
__________________________________________________________________
__________________________________________________________________
__________________________________________________________________
__________________________________________________________________
__________________________________________________________________
__________________________________________________________________
__________________________________________________________________
__________________________________________________________________
```

On your router, use the **show ip inspect sessions** command to see CBAC sessions:

```
Router# show ip inspect sessions
_________________________________________________________________
_________________________________________________________________
_________________________________________________________________
_________________________________________________________________
_________________________________________________________________
```

**Step 10** Use the **show ip auth-proxy statistics** command to verify the authorization proxy statistics. Fill in the blanks below using the output from this command.

```
Router# show ip auth-proxy statistics
Authentication Proxy Statistics
    proxied client number _____
```

**Step 11** Use the **show ip auth-proxy cache** command to verify the authorization proxy configuration. Fill in the blanks below using the output from this command.

```
Router# show ip auth-proxy cache
_________________________________________________________________
_________________________________________________________________
```

# Contact VSEC Training

VSEC Training values your opinion. Let us know what you think about this course, any suggestions you have for this course, or any inaccuracies that you find in the course material. Send an e-mail to *vsec-tng@cisco.com*. You must include one of the following statements in the subject line of your e-mail, which summarizes your message:

- Kudos
- Technical inaccuracies
- Grammatical/style inaccuracies
- General suggestions