

# PIX/ASA : Cut-through Proxy for Network Access using TACACS+ and RADIUS Server Configuration Example

Document ID: 70992

---

## Introduction

### Prerequisites

- Requirements
- Components Used
- Related Products
- Conventions

### Background Information

#### Configure AAA in PIX

- Sample Configuration of PIX
- Telnet Authentication
- Console Port Authentication
- Enable Authentication
- Enable Secure Authentication of Web Clients
- Configure SSH for Authentication
- Use of the exclude Command
- Configure the RADIUS/TACACS+ Server Using Cisco Secure ACS
- Configure TACACS+ Authorization
- Configure RADIUS Authorization
- Use MAC-Based AAA Exemption
- Configure the Local Database as Fallback Method
- Virtual Telnet

### Verify

### Troubleshoot

- Problem: Unable to get directly into the enable mode after authentication in PIX/ASA
- Solution

### Related Information

---

## Introduction

This document describes how to create AAA-authenticated (Cut-through Proxy) access to a PIX Firewall that runs PIX Software versions 6.3 and later.

The PIX Firewall offers performance that is dramatically better than competing firewalls. It gains speed through a patent pending process called Cut-through Proxies, which is the fastest way for a firewall to authenticate a user.

Unlike a proxy server that must analyze every packet at layer seven of the OSI model, a time- and processing-intensive function, the PIX Firewall first queries a TACACS+ or RADIUS server for authentication. Once approved, the PIX Firewall then establishes a data flow and all traffic thereafter flows directly and quickly between the two parties.

Cut-through Proxies let the PIX Firewall perform dramatically faster than proxy-based servers while maintaining session state. Cut-through Proxy also lowers the cost of ownership by reusing the existing authentication database.

In order to create AAA–authenticated access to a PIX Firewall that runs PIX Software version 5.2 through 6.2, refer to [How To Perform Authentication and Enabling on the Cisco Secure PIX Firewall \(5.2 Through 6.2\)](#). This document also provides information about how to enable authentication, syslogging, and gaining access when the AAA server is down.

In order to learn more about the authentication and authorization command for PIX 6.2, refer to [Authentication and Command Authorization for PIX 6.2](#).

## Prerequisites

### Requirements

There are no specific requirements for this document.

### Components Used

The information in this document is based on these software and hardware versions:

- Cisco Secure PIX Firewall Software versions 6.3 and later
- Cisco Secure Access Control Server (ACS) version 3.2 and later

The information in this document was created from the devices in a specific lab environment. All of the devices used in this document started with a cleared (default) configuration. If your network is live, make sure that you understand the potential impact of any command.

### Related Products

This configuration can also be used with Cisco ASA 5500 Series Security Appliance with 7.x version.

### Conventions

Refer to [Cisco Technical Tips Conventions](#) for more information on document conventions.

## Background Information

You can use access lists to control traffic based on the IP address and protocol. However, you must use authentication and authorization in order to control access and use for specific users or groups. Authentication, which is the process of identifying users, is supported by the PIX Firewall for RADIUS and TACACS+ servers. Authorization identifies the specific permissions for a given user.

If you want to apply authentication and authorization when an internal (local) host initiates a connection to an external (lower security) network, you need to enable it on the internal (higher security) interface. In order to set up authentication and authorization to occur when an external host initiates a connection to an internal host, you need to enable it on the outside interface.

AAA enables the security appliance to determine who the user is (authentication), what the user can do (authorization), and what the user did (accounting).

AAA provides an extra level of protection and control for user access than using Access Control Lists (ACLs) alone. For example, you can create an ACL that allows all outside users to access Telnet on a server on the DMZ network. If you want only some users to access the server and you might not always know IP addresses of these users, you can enable AAA to allow only authenticated and/or authorized users to make it through the

security appliance. The Telnet server also enforces authentication. The security appliance prevents unauthorized users from an attempt to access the server.

You can use authentication alone or with authorization and accounting. Authorization always requires a user to be authenticated first. You can use accounting alone, or with authentication and authorization.

## Configure AAA in PIX

In this section, you are presented with the information to configure the features described in this document.

**Note:** Use the Command Lookup Tool (registered customers only) to obtain more information on the commands used in this section.

You must complete these in order to enable authentication and authorization:

1. Identify the IP address of the authentication server that you will use and determine a server encryption key to be shared by the authentication server and the PIX Firewall.
2. Configure the authentication server with the users that can access the network, the services that they can use, and the hosts that they can access.
3. Configure the PIX Firewall to either enable or disable authentication or authorization.

In addition, you can configure the PIX Firewall in order to control user access to specific hosts or services. However, it is easier to maintain this kind of access control in a single location, at the authentication server. After you enable authentication and authorization, the PIX Firewall prompts users of FTP, Telnet, or HTTP (Web) access. The control of access to a specific system or service is handled by the authentication and authorization server.

**Note:** When you use PIX Firewall version 6.3 or later, you can enable authentication with a user database that you configure locally on your PIX Firewall. The configuration steps are similar to those for the configuration of a RADIUS/TACACS+ server. The differences are noted within each step of this procedure.

Complete these steps in order to enable the PIX Firewall to support user authentication and authorization:

1. For inbound authentication, create the **static** and **access-list** command statements required to permit outside hosts to access servers on the inside network.
2. If the internal network connects to the Internet, create a global address pool of registered IP addresses.
3. Issue the **nat** command with the **access-list** command in order to specify the inside hosts that can start outbound connections.
4. Issue the **aaa-server** command in order to identify the server that handles authentication or authorization.

Create a unique server group name.

For example:

```
pix(config)#aaa-server AuthInbound protocol tacacs+
pix(config)#aaa-server AuthInbound (inside) host 10.1.1.1 TheUauthKey
pix(config)#aaa-server AuthOutbound protocol tacacs+
pix(config)#aaa-server AuthOutbound (inside) host 10.1.1.2 TheUauthKey
```

**Note:** This step is not required when you use the LOCAL database for authentication.

**Note:** The RADIUS authorization is provided with the **access-list** command statement as described in the Configure RADIUS Authorization section.

The first command statement in this example creates the AuthInbound authentication group using TACACS+ authentication. The second command statement states that the AuthInbound server is on the inside interface, that the IP address is 10.1.1.1, and the encryption key is TheUauthKey.

The third command statement creates the AuthOutbound authentication group using TACACS+ authentication. The fourth command statement states that the AuthOutbound server is on the inside interface, that the IP address is 10.1.1.2, and the encryption key is TheUauthKey.

5. Issue the **aaa authentication** command in order to enable authentication.

```
pix(config)#aaa authentication include authen_service if_name 0 0 0 0
<server_tag/LOCAL>
```

The **authen\_service** portion of the command represents the type of traffic to include or exclude from authentication. This is based on the service option selected, such as ftp, telnet, http or https.

These are the access authentication service options:

- ◆ enable
- ◆ serial
- ◆ ssh
- ◆ telnet

Specify serial for serial console access, telnet for Telnet access, ssh for SSH access, and enable for enable-mode access. The **cut-through authentication** service options are these:

- ◆ telnet
- ◆ ftp
- ◆ http
- ◆ https
- ◆ icmp/type
- ◆ proto
- ◆ tcp/port
- ◆ udp/port

The variable proto can be any supported IP protocol value or name, such as ip or igmp. Only telnet, ftp, http, or https traffic trigger interactive user authentication. Refer to the **aaa authentication** command section in Cisco PIX Firewall Command Reference for information about this option.

The **if\_name** portion of the command represents the interface name from which users require authentication. If you configured as outside (as configured with the **nameif** command), it enables authentication for connections originated from the outside network to any inside network. Use the LOCAL keyword in order to use the LOCAL database for authentication. In order to use an AAA server, replace the server\_tag with the AAA server group name defined with the **aaa-server** command.

For example:

An IP address of 0 means all hosts. When you set the local IP address to 0, this lets the authentication server decide which hosts are authenticated.

This example enables authentication for any ftp connections originated from the outside network to the any inside network:

```
pix(config)#aaa authentication include ftp outside 0 0 0 0 AuthOutbound
pix(config)#aaa authentication include telnet outside 0 0 0 0 AuthOutbound
pix(config)#aaa authentication include http outside 0 0 0 0 AuthOutbound
pix(config)#aaa authentication include ftp inside 0 0 0 0 AuthInbound
```

This example enables authentication for any telnet connections originated from the inside network to the any outside network:

```
pix(config)#aaa authentication include telnet inside 0 0 0 0 AuthInbound
pix(config)#aaa authentication include http inside 0 0 0 0 AuthInbound
```

**Note:** Be careful to apply authentication only to protocols that can be authenticated. If you use the any keyword to apply authentication, protocols such as Simple Mail Transfer Protocol (SMTP) are prevented from passing through the PIX Firewall.

6. Issue the **aaa authorization** command in order to enable authorization.

```
pix(config)#aaa authorization include authen_service if_name 0 0 0 0
```

The PIX Firewall checks the authorization request with the AAA server, which makes the decision about what services a user can access.

The **authen\_service** portion of the command represents the services that require authorization. Valid values are any, ftp, http, telnet, or protocol/port. Use any to provide authorization for all TCP services. Use the protocol/port form to provide authorization for UDP services.

The **if\_name** portion of the command represents the interface name from which users require authentication. Use interface-name, combined with the local-ip address and the foreign-ip address, in order to determine where access is sought and from whom. The local-ip address is always on the highest security level interface and foreign-ip is always on the lowest.

An IP address of 0 means all hosts. When you set the local IP address to 0, this lets the authorization server decide which hosts are authorized.

**Note:** This step is not required when you use the LOCAL database for authentication.

For example:

```
pix(config)#aaa authorization include ftp outside 0 0 0 0
pix(config)#aaa authorization include telnet outside 0 0 0 0
pix(config)#aaa authorization include http outside 0 0 0 0
pix(config)#aaa authorization include ftp inside 0 0 0 0
pix(config)#aaa authorization include telnet inside 0 0 0 0
pix(config)#aaa authorization include http inside 0 0 0 0
```

Refer to Cisco PIX Firewall Command Reference for more information about the different options available for the authorization and authentication parameters.

## Sample Configuration of PIX

The PIX Firewall lets you define separate groups of TACACS+ or RADIUS servers in order to specify different types of traffic, such as a TACACS+ server for inbound traffic and a RADIUS server for outbound traffic.

This example creates the AuthInbound for TACACS+ authentication, AuthOutbound server groups for RADIUS authentication, and specifies that server 172.68.118.101 on the inside interface provides authentication.

```
pix(config)#aaa-server AuthInbound protocol tacacs+
pix(config)#aaa-server AuthInbound (inside) host 172.68.118.101 cisco timeout 5
pix(config)#aaa-server AuthOutbound protocol radius
pix(config)#aaa-server AuthOutbound (inside) host 172.68.118.101 cisco timeout 5
pix(config)#aaa authentication include ftp outside 0 0 0 0 AuthOutbound
pix(config)#aaa authentication include telnet outside 0 0 0 0 AuthOutbound
pix(config)#aaa authentication include http outside 0 0 0 0 AuthOutbound
pix(config)#aaa authentication include ftp inside 0 0 0 0 AuthInbound
pix(config)#aaa authentication include telnet inside 0 0 0 0 AuthInbound
pix(config)#aaa authentication include http inside 0 0 0 0 AuthInbound
```

From PIX 7.x, you can specify the number of failed attempts allowed for any given server in the server group before that server is deactivated. Issue the **max-failed-attempts** command in AAA server group mode. You must have the AAA server group configured before you can issue this command. For example:

```
hostname(config)#aaa-server svrgrp1 protocol tacacs+
hostname(config-aaa-server-group)#max-failed-attempts 4
```

## Telnet Authentication

You see a request for the PIX password and then a request for the RADIUS or TACACS username and password (stored on the TACACS server – 10.31.1.41 ). For example:

```
pix(config)#aaa-server topix protocol tacacs+
pix(config)#aaa-server topix host 10.31.1.41 cisco timeout 5
pix(config)#aaa authentication telnet console topix
```

## Console Port Authentication

You see a request for the PIX password and then a request for the RADIUS/TACACS username/password (stored on the RADIUS server – 10.31.1.41 ). For example:

```
pix(config)#aaa-server topix protocol radius
pix(config)#aaa-server topix host 10.31.1.41 cisco timeout 5
pix(config)#aaa authentication serial console topix
```

## Enable Authentication

Here, you are prompted for a username and password which is sent to the TACACS or RADIUS server. You can log into the PIX with TACACS or RADIUS and enable through TACACS or RADIUS with the same username/password because the authentication packet for enable is the same as the authentication packet for login.

```
pix(config)#aaa-server topix protocol radius

pix(config)#aaa-server topix host 10.31.1.41 cisco timeout 5

pix(config)#aaa authentication enable console topix
```

## Enable Secure Authentication of Web Clients

PIX Firewall version 6.3 introduces a secured method to exchange usernames and passwords between a web client and a PIX Firewall. This version uses HTTP over the Secure Socket Layer (SSL) (HTTPS). HTTPS encrypts the username and password, and makes the transmission secure.

When you authenticated a web browser using a AAA server on earlier versions of PIX Firewall, the username and password were obtained from the HTTP client in clear text.

Add this keyword to the **aaa** command to enable this feature:

```
pix(config)#aaa authentication secure-http-client
```

The keyword `secure-http-client` enables this feature so the username and password are exchanged securely between HTTP clients and the PIX Firewall.

You must configure AAA authentication and issue this command in order to enable this feature:

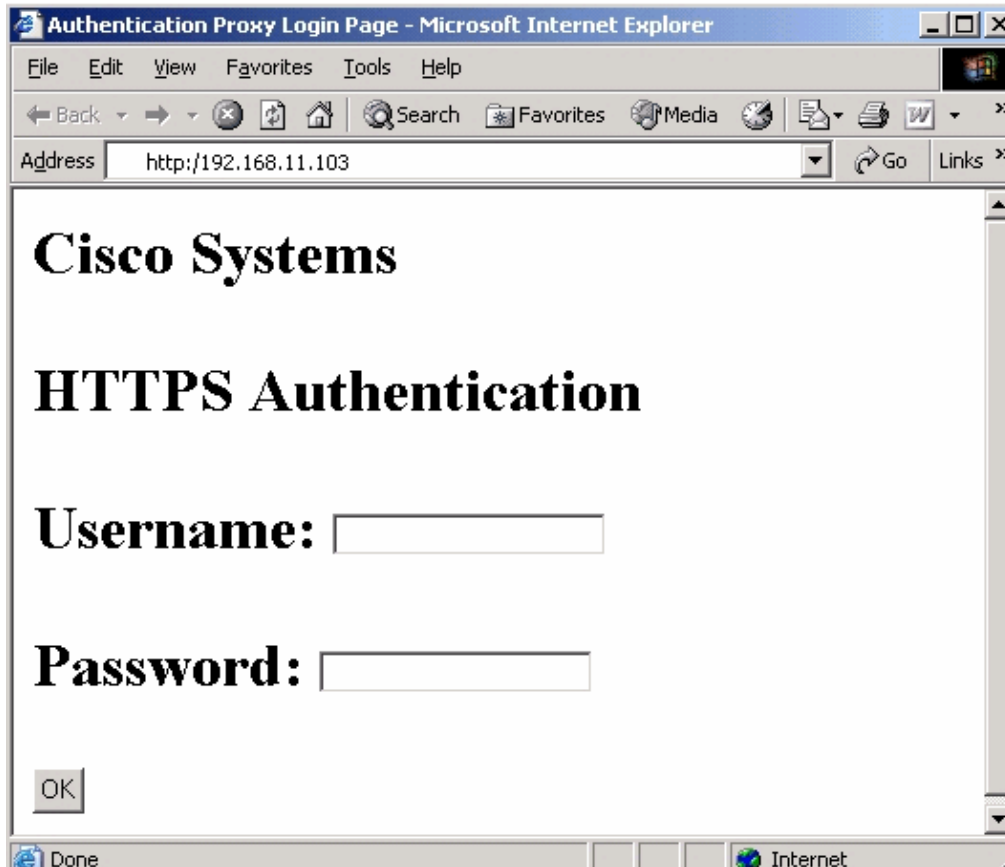
```
pix(config)#aaa authentication include authen_service if_name 0 0 0 0 <server_tag|LOCAL>
```

See the Configure AAA in PIX section for the syntax of this command.

This feature also supports authentication of clients that access secure (HTTPS) websites.

**Note:** When you enable AAA authentication, `secure-http-client` is not required to authenticate HTTPS sessions.

After you enable this feature, when a user accesses a web page that requires authentication, the PIX Firewall requests a username and password for HTTPS authentication.



**Note:** The **auth-prompt** command is issued in this example in order to customize the Cisco Systems text field. This field is blank if you do not enter a string with the **auth-prompt** command. Refer to Cisco PIX Firewall Command Reference for the detailed syntax of this command.

After the user enters a valid username and password, an Authentication Successful window appears and closes automatically. If the user fails to enter a valid username and password, an Authentication Failed window appears.

A maximum of 16 concurrent HTTPS authentications are allowed. If all 16 HTTPS authentication processes run, a new connection that requires authentication does not succeed. An authentication process starts when the PIX Firewall receives the username and password from the browser and ends when it receives the authentication result from the AAA server. The length of time required to complete each authentication process depends on the response time from the authentication source. If the LOCAL database is used, it is very fast. If a RADIUS or TACACS+ server is used, it depends on the server response time.

When the **uauth timeout 0** command is configured (the uauth timeout is set to 0), HTTPS authentication might not work. If a browser initiates multiple TCP connections to load a web page after HTTPS authentication, the first connection is let through, but the subsequent connections trigger authentication. As a result, users are continuously presented with an authentication page, even if the correct username and password are entered each time. The workaround is to set the uauth timeout to 1 second with the **timeout uauth 0:0:1** command. However, this workaround opens a 1-second window of opportunity that might allow non-authenticated users to go through the firewall if they come from the same source IP address.

If a web browser launches an HTTPS web page request while secure authentication is in process for a previous HTTP request, the HTTPS request triggers a second secure authentication process, even if secure authentication is not specifically enabled for HTTPS. Once the authentication process for either web page is completed successfully, the request that remains can be completed by reloading the page.

Because HTTPS authentication occurs on the SSL port 443, do not use the **access-list** command to block traffic from the HTTP client to HTTP server on port 443. Also, if you configure static PAT for web traffic on port 80, you must also configure a static entry for SSL port 443. In this example, the first line configures static PAT for web traffic and the second line must be added to support the HTTPS authentication configuration in configuration mode:

```
static (inside,outside) tcp 10.132.16.200 www 10.130.16.10 www
static (inside,outside) tcp 10.132.16.200 443 10.130.16.10 443
```

HTTP users see a pop-up window generated by the browser if the **aaa authentication secure-http-client** command is not configured. If the **aaa authentication secure-http-client** command is configured, a form loads in the browser to collect the username and password. In either case, if you enter an incorrect password, you are prompted again. When the web server and the authentication server are on different hosts, issue the **virtual** command to receive the correct authentication behavior.

## Configure SSH for Authentication

PIX 5.2 added Secure Shell (SSH) version 1 support. SSH 1 is based on a November, 1995, IETF draft. SSH versions 1 and 2 are not compatible with each other. Refer to Secure Shell (SSH) Frequently Asked Questions for more information about SSH.

The PIX is considered the SSH server. Traffic from SSH clients (that is, boxes running SSH) to the SSH server (the PIX) is encrypted. Some SSH version 1 clients are listed in the PIX 5.2 release notes. Cisco lab tests were done with F-secure SSH 1.1 on NT and version 1.2.26 for Solaris.

**Note:** For PIX 7.x, refer to the Allowing SSH Access section of Managing System Access. Also, refer to the sample configuration in PIX/ASA 7.x: SSH/Telnet on the Inside and Outside Interface Configuration Example.

Complete these steps in order to configure AAA authenticated SSH:

1. Make sure you can Telnet to PIX with AAA on, but without SSH:

```
pix(config)#aaa-server AuthOutbound protocol radius (or tacacs+)
pix(config)#aaa authentication telnet console AuthOutbound
pix(config)#aaa-server AuthOutbound host 172.18.124.111 cisco
```

**Note:** When SSH is configured, the **telnet 172.18.124.114 255.255.255.255** command is not needed because the **ssh 172.18.124.114 255.255.255.255 inside** command is issued on the PIX. Both commands are included for testing purposes.

2. Issue these commands in configuration mode in order to add SSH:

```
hostname goss-d3-pix515b
domain-name rtp.cisco.com
ca gen rsa key 1024
```

```
!--- Caution: The RSA key is not saved without
!--- the ca save all command.
!--- The write mem command does not save it.
!--- In addition, if the PIX has undergone a write erase
!--- or has been replaced, then cutting and pasting
!--- the old configuration does not generate the key.
!--- You must re-enter the ca gen rsa key command.
!--- If there is a secondary PIX in a failover pair, the write standby
!--- command does not copy the key from the primary to the secondary.
!--- You must also generate and save the key on the secondary device.
```

```
ssh 172.18.124.114 255.255.255.255 inside
ssh timeout 60
aaa authen ssh console AuthOutbound
logging trap debug
logging console debug
```

3. Issue the **show ca mypubkey rsa** command in config mode:

```
goss-d3-pix(config)#show ca mypubkey rsa
% Key pair was generated at: 08:22:25 Aug 14 2000
Key name: goss-d3-pix.rtp.cisco.com
Usage: General Purpose Key
Key Data:
 30819f30 0d06092a 864886f7 0d010101 05000381 8d003081 89028181 00ad4bc
e9c174d5 0657a0f3 c94e4b6d 32ac8500 6b84e754 59e20df4 f28c257d 131af21d
4c0a8f4c e79d8b6d a3520faa 1a42d577 c6adfe51 9d96fa62 f3be07fb 01e082d7
133cecff bf24f653 bc690b11 ee222070 413c1920 d02321f8 4fc3c5f1 f0c6e077
81e93184 af55438b dcdca34 c0a5f5ad 87c435ef
 67170674 4d5ba51e 6d020301 0001
% Key pair was generated at: 08:27:18 Aug 14 2000
Key name: goss-d3-pix.rtp.cisco.com.server
Usage: Encryption Key
Key Data:
 307c300d 06092a86 4886f70d 01010105 00036b00 30680261 00d4f61b ec45843a
4ad9266d b125ee26 efc63cc4 e5e9cda4 9418ee53 6e4d16cf 3d0dc864 4d4830c8
fa7f110e 8a5761ed 4ca73ea7 5d405862 6f3150df 9eb0d11e 9c4d3563 95ff51ae
6711d60b 9a1415e4 19201d3f 03b455ea c1df9a41 b3a5a73f 4f020301 0001
```

4. Issue a Telnet from the Solaris station:

```
rtp-evergreen#./ssh -c 3des -l cisco -v 172.18.124.157
```

**Note:** The username on the RADIUS/TACACS+ server is cisco and 172.18.124.157 is the destination.

## Use of the exclude Command

If you add another host outside (at 99.99.99.100) to the network, and this host is trusted, you can exclude them from authentication and authorization with these commands:

```
pix(config)#aaa authentication exclude telnet inbound 0.0.0.0 0.0.0.0 99.99.99.100
255.255.255.255 AuthInbound
```

```
pix(config)#aaa authorization exclude telnet inbound 0.0.0.0 0.0.0.0 99.99.99.100
255.255.255.255 AuthInbound
```

## Configure the RADIUS/TACACS+ Server Using Cisco Secure ACS

Complete these steps in order to configure RADIUS and TACACS+ in a Cisco Secure ACS.

1. You must configure the PIX to locate the CSACS in order to check the user credentials.

For example:

```
pix(config)#aaa-server AuthOutbound protocol radius
pix(config)#aaa-server AuthOutbound (inside) host 171.68.118.101 testkey
```

2. Choose **Network Configuration** on the left and click **Add Entry** to add an entry for the PIX in either the TACACS+ or RADIUS server database. Choose the server database according to the PIX configuration.



# Network Configuration

Select

- User Setup
- Group Setup
- Shared Profile Components
- Network Configuration
- System Configuration
- Interface Configuration
- Administration Control
- External User Databases
- Reports and Activity
- Online Documentation

AAA Clients		
AAA Client Hostname	AAA Client IP Address	Authenticate Using
<a href="#">PIX</a>	172.16.5.1	TACACS+ (Cisco IOS)
<a href="#">PIX-A</a>	172.16.1.85	RADIUS (Cisco IOS/PIX)
<a href="#">VPN3000</a>	172.16.5.2	TACACS+ (Cisco IOS)
<a href="#">WLC</a>	172.16.1.31	RADIUS (Cisco Aironet)
<a href="#">WLC Main</a>	172.16.1.50	RADIUS (Cisco Aironet)

Add Entry

Search

- Enter **172.16.1.85** in the IP address field, and enter **test** for the shared secret Key field. Choose **RADIUS (Cisco IOS/PIX)** in the Authenticate Using drop-down box. Click **Submit**.



# Network Configuration

Edit

- User Setup
- Group Setup
- Shared Profile Components
- Network Configuration
- System Configuration
- Interface Configuration
- Administration Control
- External User Databases
- Reports and Activity
- Online Documentation

## Add AAA Client

AAA Client Hostname	<input type="text" value="pix7"/>
AAA Client IP Address	<input type="text" value="172.16.1.85"/>
Key	<input type="text" value="test"/>
Authenticate Using	<input type="text" value="RADIUS (Cisco IOS/PIX)"/>
<input type="checkbox"/> Single Connect TACACS+ AAA Client (Record stop in accounting on failure).	
<input type="checkbox"/> Log Update/Watchdog Packets from this AAA Client	
<input type="checkbox"/> Log RADIUS Tunneling Packets from this AAA Client	
<input type="checkbox"/> Replace RADIUS Port info with Username from this AAA Client	

The key is used to authenticate between the PIX and CSACS server (RADIUS Server). If you want to select the TACACS+ protocol for authentication, then choose **TACACS+(Cisco IOS)** in the Authenticate Using drop down menu.



# Network Configuration

Edit

- User Setup
- Group Setup
- Shared Profile Components
- Network Configuration
- System Configuration
- Interface Configuration
- Administration Control
- External User Databases
- Reports and Activity
- Online Documentation

## Add AAA Client

AAA Client Hostname	<input type="text" value="pix7"/>
AAA Client IP Address	<input type="text" value="172.16.1.1"/>
Key	<input type="text" value="test"/>
Authenticate Using	<input type="text" value="TACACS+ (Cisco IOS)"/>
<input type="checkbox"/> Single Connect TACACS+ AAA Client (Record stop in accounting on failure).	
<input type="checkbox"/> Log Update/Watchdog Packets from this AAA Client	
<input type="checkbox"/> Log RADIUS Tunneling Packets from this AAA Client	
<input type="checkbox"/> Replace RADIUS Port info with Username from this AAA Client	

4. Enter the username in the User field in the Cisco Secure database, then click **Add/Edit**.

In this example, the username is user1.



# User Setup

Select

- User Setup
- Group Setup
- Shared Profile Components
- Network Configuration
- System Configuration
- Interface Configuration
- Administration Control
- External User Databases
- Reports and Activity
- Online Documentation

User:

List users beginning with letter/number:

[A](#) [B](#) [C](#) [D](#) [E](#) [F](#) [G](#) [H](#) [I](#) [J](#) [K](#) [L](#) [M](#)  
[N](#) [O](#) [P](#) [Q](#) [R](#) [S](#) [T](#) [U](#) [V](#) [W](#) [X](#) [Y](#) [Z](#)  
[0](#) [1](#) [2](#) [3](#) [4](#) [5](#) [6](#) [7](#) [8](#) [9](#)

5. In the next window, enter the password for user1.

In this example, the password is also password1. You can map the user account to a group if you wish. When you have finished, click **Submit**.



# User Setup

- User Setup
- Group Setup
- Shared Profile Components
- Network Configuration
- System Configuration
- Interface Configuration
- Administration Control
- External User Databases
- Reports and Activity
- Online Documentation

### Supplementary User Info

Real Name:

Description:

### User Setup

Password Authentication:

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

Password:

Confirm Password:

Separate (CHAP/MS-CHAP/ARAP)

Password:

Confirm Password:

When a token server is used for authentication, supplying a separate CHAP password for a token card user allows CHAP authentication. This is

## Configure TACACS+ Authorization

You can configure the security appliance to perform network access authorization with TACACS+. You specify the ACLs that authorization rules must match in order to identify the traffic to be authorized. Alternatively, you can identify the traffic directly in authorization rules themselves.

The use of ACLs to identify traffic to be authorized can greatly reduced the number of authorization commands you must enter. This is because each authorization rule you enter can specify only one source, destination subnet and service, whereas an ACL can include many entries.

Authentication and authorization statements are independent. However, any unauthenticated traffic matched by an authorization statement is denied. For authorization to succeed, a user must first authenticate with the security appliance. Because a user at a given IP address only needs to authenticate one time for all rules and types, if the authentication session has not expired, authorization can occur even if the traffic is matched by an authentication statement.

After a user authenticates, the security appliance checks the authorization rules for matching traffic. If the traffic matches the authorization statement, the security appliance sends the username to the TACACS+ server. The TACACS+ server responds to the security appliance with a permit or a deny for that traffic, based on the user profile. The security appliance enforces the authorization rule in the response.

See the documentation for your TACACS+ server for information on how to configure network access authorizations for a user.

Complete these steps in order to configure TACACS+ authorization:

1. Enable authentication.

Refer to Enabling Network Access Authentication for more information. If you have already enabled authentication, continue to the next step.

2. Issue the **access-list** command in order to create an ACL that identifies the source addresses and destination addresses of traffic you want to authorize.

For steps, refer to Adding an Extended Access List. The permit ACEs mark matching traffic for authorization, while deny entries exclude matching traffic from authorization. The ACL you use for authorization matching should contain rules that are equal to, or a subset of, the rules in the ACL used for authentication matching.

**Note:** If you have configured authentication and want to authorize all the traffic being authenticated, you can use the same ACL you created for use with the **aaa authentication match** command.

3. Issue this command in order to enable authorization:

```
hostname/contexta(config)#aaa authorization match acl_name
interface_name server_group
```

The **acl\_name** portion of the command is the name of the ACL you created in step 2, the **interface\_name** portion of the command is the name of the interface as specified with the **nameif** command or by default, and the **server\_group** portion of the command is the AAA server group you created when you enabled authentication.

**Note:** Alternatively, you can use the **aaa authorization include** command. This identifies traffic within the command. However, you cannot use both methods in the same configuration. Refer to Cisco Security Appliance Command Reference for more information.

These commands authenticate and authorize inside Telnet traffic. Telnet traffic to servers other than 10.165.201.5 can be authenticated alone, but traffic to 10.165.201.5 requires authorization.

```
hostname/contexta(config)#access-list TELNET_AUTH extended permit tcp any any
eq telnet
```

```
hostname/contexta(config)#access-list SERVER_AUTH extended permit tcp any host
10.165.201.5 eq telnet
```

```
hostname/contexta(config)#aaa-server AuthOutbound protocol tacacs+
```

```
hostname/contexta(config-aaa-server-group)#exit
```

```
hostname/contexta(config)#aaa-server AuthOutbound (inside) host 10.1.1.1
```

```
hostname/contexta(config-aaa-server-host)#key TACPlusUauthKey
```

```
hostname/contexta(config-aaa-server-host)#exit
```

```
hostname/contexta(config)#aaa authentication match TELNET_AUTH inside AuthOutbound
```

```
hostname/contexta(config)#aaa authorization match SERVER_AUTH inside AuthOutbound
```

## Configure RADIUS Authorization

The PIX Firewall allows a RADIUS server to send user group attributes to the PIX Firewall in the RADIUS authentication response message.

The administrator first defines access lists on the PIX Firewall for each user group. For example, there could be access lists for each department in an organization, sales, marketing, engineering, and so on. The administrator then lists the access list in the group profile in the Cisco version of RADIUS, called CiscoSecure.

The PIX Firewall requests authentication of the user by the RADIUS server. If the user is authorized, the RADIUS server returns a confirming authorization response message to the PIX Firewall with vendor specific attribute 11 (filter-id) set to the access list for the given group of the user. RADIUS attribute 11 cannot be used to pass this information.

The PIX Firewall also provides the same functionality for TACACS+ in order to maintain consistency.

**Note:** Access lists can be used with either RADIUS or TACACS, but authorizing FTP, HTTP, or Telnet is only possible with TACACS+.

Issue these **access-list** command statements in order to restrict users in a department to three servers and deny everything else:

```
access-list eng permit ip any server1 255.255.255.255
access-list eng permit ip any server2 255.255.255.255
access-list eng permit ip any server3 255.255.255.255
access-list eng deny ip any any
```

In this example, the vendor-specific attribute string in the CiscoSecure configuration has been set to `acl=eng`. Use this field in the CiscoSecure configuration in order to identify the access list identification name. The PIX Firewall receives the `acl=acl_ID` string from CiscoSecure, extracts the ACL identifier, and puts it in the `uauth` entry of the user.

When a user tries to open a connection, the PIX Firewall checks the access list in the `uauth` entry of the user and , permits or denies the connection. This depends on the permit or deny status of the access list match. When a connection is denied, the PIX Firewall generates a corresponding syslog message. If there is no match, then the implicit rule is to deny.

Because the source IP of a given user can vary depending on where they are logging in from, set the source address in the **access-list** command statement to any, and the destination address to identify the network services to which user is permitted or denied access.

**Note:** The **aaa authorization** command does not require a separate RADIUS option.

## Use MAC-Based AAA Exemption

PIX Firewall versions 6.3 and later let you use MAC addresses to bypass authentication for devices, such as Cisco IP Phones, that do not support AAA authentication. You need to identify the MAC addresses on the inside (higher security) interface in order to use this feature. The PIX Firewall uses the MAC address and the IP address, that has been dynamically assigned to the MAC address, in order to bypass the AAA server for traffic that matches. Authorization services are automatically disabled when you bypass authentication. The accounting records are still generated (if enabled), but the username is not displayed.

Create a list of MAC addresses to be exempted from AAA authentication and then assign the list to a AAA server in order to enable MAC-based AAA exemption.

**Note:** This feature cannot be applied on the outside or lower security level interface.

1. Issue this command in configuration mode in order to define a list of MAC addresses:

```
mac-list mcl-id deny | permit mac mac-mask
```

Issue this command as many times as necessary to define all the MAC addresses you want to add to the list. Replace `mcl-id` with the identifier of the MAC list.

Use the `permit` option to identify the MAC addresses to be exempted from authentication. Use the `deny` option to prevent the bypassing of authentication. Replace `mac` with a partial MAC address that can be used to select a group of devices based on a common portion of the hardware address, such as the vendor ID. Replace `mac-mask` with a mask that identifies the portion of the MAC address that should be used for matching.

For example, this command bypasses authentication for a single MAC address:

```
mypix(config)#mac-list adc permit 00a0.c95d.0282 ffff.ffff.ffff
```

In this example, the mask `FFFF.FFFF.FFFF` instructs the PIX Firewall to match all 12 digits (six bytes) in the preceding hexadecimal address.

This command bypasses authentication for all Cisco IP Phones, which have the hardware ID `0003.E3`:

```
mypix(config)#mac-list adc permit 0003.E300.0000 FFFF.FF00.0000
```

2. Issue this command in order to apply the MAC list to the AAA server:

```
mypix(config)#aaa mac-exempt match<mcl-id>
```

Replace `mcl-id` with the identifier for the MAC list that you want to apply.

For example, this command applies the MAC-list `adc` to the AAA server:

```
mypix(config)#aaa mac-exempt match<adc>
```

3. Issue this command in order to view the current entries in a specific MAC list:

```
mypix#show mac-list [mcl-id]
```

If you omit the MAC list identifier, the system displays all currently configured MAC lists.

4. Issue this command in order to clear all the entries on a MAC list:

```
mypix#clear mac-list [mclid]
```

If you omit the MAC list identifier, the system clears all the currently configured MAC lists.

## Configure the Local Database as Fallback Method

This section describes how to manage users in the local database. You can use the local database for CLI access authentication, privileged mode authentication, command authorization, network access authentication, and VPN authentication and authorization. You cannot use the local database for network access

authorization. The local database does not support accounting.

For multiple context mode, you can configure usernames in the system execution space in order to provide individual logins using the login command. However, you cannot configure any **aaa** commands in the system execution space.

**Note:** If primary authentication server, for example, ACS, is up and reachable, but the username does not exist in the database, then ASA does not allow fallback to local database to search for this user. ASA falls back to LOCAL database or secondary database only if primary server goes down.

Complete these steps:

1. Issue these commands in order to define a user account in the local database:

- ◆ For PIX 6.3:

```
hostname(config)#username username [{nopassword | password  
password} [encrypted]] [privilege level]}
```

- ◆ For PIX/ASA 7.x:

Syntax

```
hostname(config)#username user password password1 encrypted  
pix(config)#aaa authentication {telnet | ssh | http | serial} console  
{LOCAL | server_group [LOCAL]}
```

**Note:** If you use a TACACS+ or RADIUS server group for authentication, you can configure the security appliance to use the local database as a fallback method if the AAA server is unavailable. Specify the server group name followed by LOCAL (LOCAL is case sensitive). Cisco recommends that you use the same username and password in the local database as the AAA server because the security appliance prompt does not give any indication which method is being used.

Example:

```
pix(config)#aaa authentication ssh console TACACS+ LOCAL
```

You can alternatively use the local database as your main method of authentication (with no fallback) by entering LOCAL alone. For example, issue this command in order to define a user account in the local database and to perform local authentication for an SSH connection:

Example:

```
pix(config)#aaa authentication ssh console LOCAL
```

2. From PIX 7.x, issue this command in order to configure a local user account with VPN attributes:

```
hostname/contexta(config)#username username attributes
```

When you issue a **username attributes** command, you enter the username mode. These are the commands available in this mode that need to be issued in order to configure the user profile:

- ◆ **group-lock**
- ◆ **password-storage**
- ◆ **vpn-access-hours**
- ◆ **vpn-filter**
- ◆ **vpn-framed-ip-address**
- ◆ **vpn-group-policy**

- ◆ **vpn-idle-timeout**
- ◆ **vpn-session-timeout**
- ◆ **vpn-simultaneous-logins**
- ◆ **vpn-tunnel-protocol**
- ◆ **webvpn**

## Virtual Telnet

Although you can configure network access authentication for any protocol or service (see the **aaa authentication match** or **aaa authentication include** command), you can authenticate directly with HTTP, Telnet, or FTP only. A user must first authenticate with one of these services before other traffic that requires authentication is allowed through. If you do not want to allow HTTP, Telnet, or FTP through the security appliance, but want to authenticate other types of traffic, you can configure virtual Telnet. The user Telnets to a given IP address configured on the security appliance, and the security appliance provides a Telnet prompt.

In order to configure a virtual Telnet server, issue this command:

```
hostname(config)#virtual telnet <ip_address>
```

The *ip\_address* argument sets the IP address for the virtual Telnet server. Make sure this address is an unused address that is routed to the security appliance.

You must configure authentication for Telnet access to the virtual Telnet address as well as the other services you want to authenticate using the **authentication match** or **aaa authentication include** command.

When an unauthenticated user connects to the virtual Telnet IP address, the user is challenged for a username and password, and then authenticated by the AAA server. Once authenticated, the user sees the message "Authentication Successful." Then, the user can successfully access other services that require authentication.

For inbound users (from lower security to higher security), you must also include the virtual Telnet address as a destination interface in the access list applied to the source interface. Also, you must add a **static** command for the virtual Telnet IP address, even if NAT is not required (using the **no nat-control** command). An identity NAT command is typically used (where you translate the address to itself).

For outbound users, there is an explicit permit for traffic. However, if you apply an access list to an inside interface, make sure to allow access to the virtual Telnet address. A static statement is not required.

In order to log out from the security appliance, reconnect to the virtual Telnet IP address. You are prompted to log out.

This example shows how to enable virtual Telnet along with AAA authentication for other services:

```
hostname(config)#virtual telnet 10.165.202.129
hostname(config)#access-list ACL-IN extended permit tcp any host 10.165.200.225
eq smtp
hostname(config)#access-list ACL-IN

!--- This is the SMTP server on the inside

hostname(config)#access-list ACL-IN extended permit tcp any host 10.165.202.129
eq telnet
hostname(config)#access-list ACL-IN

!--- This is the virtual Telnet address

hostname(config)#access-group ACL-IN in interface outside
```

```
hostname(config)#static (inside, outside) 10.165.202.129 10.165.202.129 netmask
255.255.255.255
hostname(config)#access-list AUTH extended permit tcp any host 10.165.200.225
eq smtp
hostname(config)#access-list AUTH

!--- This is the SMTP server on the inside

hostname(config)#access-list AUTH extended permit tcp any host 10.165.202.129
eq telnet
hostname(config)#access-list AUTH

!--- This is the virtual Telnet address

hostname(config)#aaa-server ACS protocol tacacs+
hostname(config)#aaa-server Acs (inside) host 10.1.1.1 TheUauthKey
hostname(config)#aaa authentication match AUTH outside ACS
hostname(config)#aaa authorization match AUTH outside ACS
```

## Verify

There is currently no verification procedure available for the configurations in this document.

## Troubleshoot

This section provides information you can use to troubleshoot your configuration.

### Problem: Unable to get directly into the enable mode after authentication in PIX/ASA

You are not able to get directly into the enable mode after authentication in PIX/ASA.

### Solution

It is not possible to get directly into the enable mode as this is not supported on PIX/ASA. You have to get into the enable mode manually.

---

## Related Information

- [Cisco Secure PIX Firewall Command References](#)
- [Cisco PIX 500 Series Security Appliances](#)
- [Remote Authentication Dial-In User Service \(RADIUS\) Support page](#)
- [Terminal Access Controller Access Control System \(TACACS+\) Support page](#)
- [Request for Comments \(RFCs\)](#)
- [Technical Support & Documentation – Cisco Systems](#)

---

[Contacts & Feedback](#) | [Help](#) | [Site Map](#)

© 2008 – 2009 Cisco Systems, Inc. All rights reserved. [Terms & Conditions](#) | [Privacy Statement](#) | [Cookie Policy](#) | [Trademarks of Cisco Systems, Inc.](#)

---

Updated: Oct 13, 2008

Document ID: 70992

---