Configuring PIX 5.1.x: TACACS+ and RADIUS

Document ID: 4613

Contents

Introduction
Prerequisites
Requirements
Components Used
Conventions
Background Information
Authentication vs. Authorization
What the User Sees with Authentication/Authorization On
Security Server Configurations Used for All Scenarios
Cisco Secure UNIX TACACS Server Configuration
Cisco Secure UNIX RADIUS Server Configuration
Cisco Secure ACS for Windows 2.x RADIUS

EasyACS TACACS+
Cisco Secure 2.x TACACS+
Livingston RADIUS Server Configuration
Merit RADIUS Server Configuration
TACACS+ Freeware Server Configuration
Debugging Steps
Network Diagram
Authentication Debug Examples from PIX
Adding Authorization
Authentication and Authorization Debug Examples from PIX
Adding Accounting
Use of Exclude Command
Max−sessions and Viewing Logged−in Users
Authentication and Enabling on the PIX Itself
Changing the Prompt Users See
Customizing the Message Users See on Success/Failure
Per−user Idle and Absolute Timeouts
Virtual HTTP
Virtual Telnet
Virtual Telnet Logout
Port Authorization
AAA Accounting for Traffic Other Than HTTP, FTP, and Telnet
Extended Authentication (Xauth)
Authentication on the DMZ
Network Diagram
PIX Configuration
Xauth Accounting
Related Information

Introduction

RADIUS and TACACS+ authentication may be done for FTP, Telnet, and HTTP connections. Authentication for other less common protocols can usually be made to work. TACACS+ authorization is supported; RADIUS authorization is not. Changes in PIX 5.1 authentication, authorization, and accounting (AAA) over

the previous version include extended authentication (xauth)−− authentication of IPSec tunnels from the Cisco Secure VPN Client 1.1.

Prerequisites

Requirements

There are no specific requirements for this document.

Components Used

This document is not restricted to specific software and hardware versions.

Conventions

For more information on document conventions, see the Cisco Technical Tips Conventions.

Background Information

Authentication vs. Authorization

- Authentication is who the user is.
- Authorization is what the user can do.
- Authentication *is* valid without authorization.
- Authorization *is not* valid without authentication.
- Accounting is what the user did.

Suppose you have one−hundred users inside and you want only want six of these users to be able to do FTP, Telnet, or HTTP outside the network. You would tell the PIX to authenticate outbound traffic and give all six users id's on the TACACS+/RADIUS security server. With simple authentication, these six users could be authenticated with username and password, then go out. The other ninety−four users could not go out. The PIX prompts users for username/password, then passes their username and password to the TACACS+/RADIUS security server, and depending on the response, opens or denies the connection. These six users could do FTP, Telnet, or HTTP.

But suppose *one* of these six users, "Festus," is not to be trusted. You would like to allow Festus to do FTP, but not HTTP or Telnet to the outside. This means having to add *authorization*, that is, authorizing *what* users can do in addition to authenticating who they are. This is only valid with TACACS+. When we add *authorization* to the PIX, the PIX first sends Festus' username and password to the security server, then sends an authorization request telling the security server what "*command*" Festus is trying to do. With the server set up properly, Festus could be allowed to "ftp 1.2.3.4" but would be denied the ability to HTTP or Telnet anywhere.

What the User Sees with Authentication/Authorization On

When trying to go from inside to outside (or vice versa) with authentication/authorization on:

- **Telnet** − The user sees a username prompt come up, then a request for password. If authentication (and authorization) is successful at the PIX/server, the user is prompted for username and password by the destination host beyond.
- **FTP** − The user sees a username prompt come up. The user needs to enter **local_username@remote_username** for username and **local_password@remote_password** for

password. The PIX sends the local_username and local_password to the local security server, and if authentication (and authorization) is successful at the PIX/server, the remote_username and remote_password are passed to the destination FTP server beyond.

- **HTTP** − A window is displayed in the browser requesting a username and password. If authentication (and authorization) is successful, the user arrives at the destination web site beyond. Keep in mind that *browsers cache usernames and passwords*. If it appears that the PIX should be timing out an HTTP connection but is not doing so, it is likely that re−authentication actually is taking place with the browser shooting the cached username and password to the PIX, which then forwards this to the authentication server. PIX syslog and/or server debug shows this phenomenon. If Telnet and FTP seem to work normally, but HTTP connections do not, this is why.
- **Tunnel** − When attempting to tunnel IPSec traffic into the network with the VPN Client and xauth on, a gray box for "User Authentication for New Connection" is displayed for username/password.

  Note: This authentication is supported beginning with the Cisco Secure VPN Client 1.1. If the **Help** > **About** menu does not show version 2.1.x or later, this does not work.

Security Server Configurations Used for All Scenarios

Cisco Secure UNIX TACACS Server Configuration

In this section, you are presented with the information to configure your security server.

Make sure that you have the PIX IP address or fully−qualified domain name and key in the CSU.cfg file.

```
user = ddunlap {
password = clear "rtp"
default service = permit
}

user = can_only_do_telnet {
password = clear "telnetonly"
service = shell {
cmd = telnet {
permit .*
}
}
}

user = can_only_do_ftp {
password = clear "ftponly"
service = shell {
cmd = ftp {
permit .*
}
}
}

user = httponly {
password = clear "httponly"
service = shell {
cmd = http {
permit .*
}
}
}
```

Cisco Secure UNIX RADIUS Server Configuration

Use the GUI to add the PIX IP address and key to the Network Access Server (NAS) list.

```
user=adminuser {
radius=Cisco {
check_items= {
2="all"
}
reply_attributes= {
6=6
}
}
}
```

Cisco Secure ACS for Windows 2.x RADIUS

Use these steps to configure the Cisco Secure ACS for Windows 2.x RADIUS.

1. Obtain a password in the User Setup GUI section.
2. From the Group Setup GUI section, set attribute 6 (Service−Type) to **Login** or **Administrative**.
3. Add the PIX IP address in the NAS Configuration section GUI.

EasyACS TACACS+

The EasyACS documentation describes setup.

1. In the group section, click **Shell exec** to give exec privileges.
2. To add authorization to the PIX, click on **Deny unmatched IOS commands** at the bottom of the group setup.
3. Select **Add/Edit new command** for each command you wish to allow, for example, **Telnet**.
4. If Telnetting to specific sites is allowed, fill in the IP address(es) in the argument section in the form "permit #.#.#.#". Otherwise, to allow Telnetting, click **Allow all unlisted arguments**.
5. Click **Finish editing command**.
6. Perform steps 1 through 5 for each of the allowed commands (for example, Telnet, HTTP or FTP).
7. Add the PIX IP in the NAS Configuration GUI section.

Cisco Secure 2.x TACACS+

The user obtains a password in the User Setup GUI section.

1. In the group section, click on **Shell exec** to give exec privileges.
2. To add authorization to the PIX, at the bottom of the group setup, click **Deny unmatched IOS commands**.
3. Select **Add/Edit new command** for each command you wish to allow (for example, **Telnet**).
4. To allow Telnetting to specific sites, enter the IP address in the argument section in the form "permit #.#.#.#". To allow Telnetting to any site, click **Allow all unlisted arguments**.
5. Click **Finish editing command**.
6. Perform steps 1 through 5 for each of the allowed commands (for example, Telnet, HTTP, or FTP).
7. Ensure the PIX IP address is added in the NAS Configuration GUI section.

Livingston RADIUS Server Configuration

Add the PIX IP address and key to the Clients file.

```
adminuser Password="all" User−Service−Type = Shell−User
```

Merit RADIUS Server Configuration

Add the PIX IP address and key to the Clients file.

```
adminuser Password="all" Service−Type = Shell−User
```

TACACS+ Freeware Server Configuration

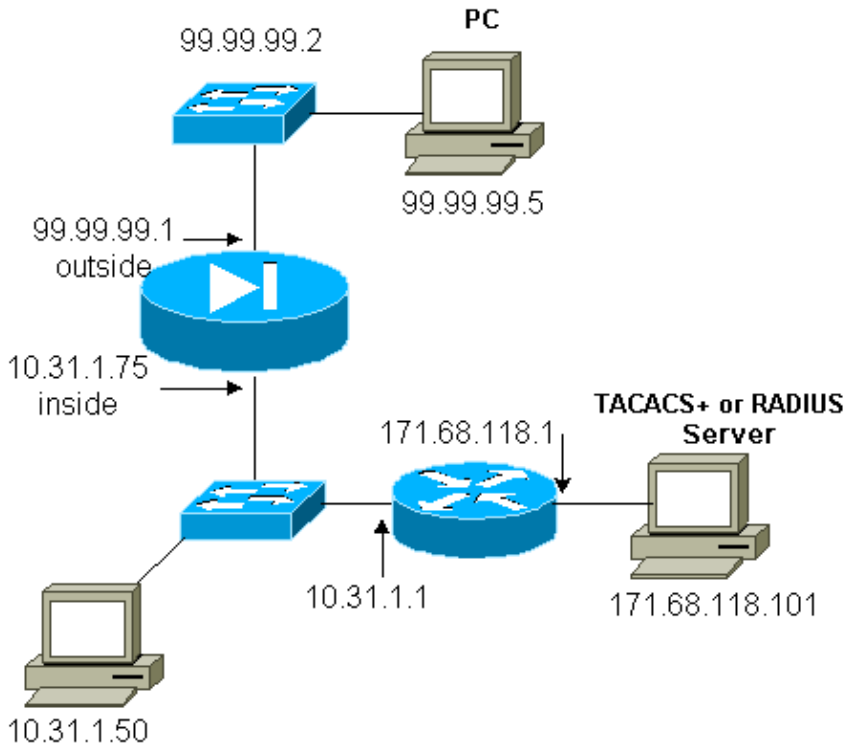
```
key = "cisco"
user = adminuser {
login = cleartext "all"
default service = permit
}

user = can_only_do_telnet {
login = cleartext "telnetonly"
cmd = telnet {
permit .*
}
}

user = httponly {
login = cleartext "httponly"
cmd = http {
permit .*
}
}

user = can_only_do_ftp {
login = cleartext "ftponly"
cmd = ftp {
permit .*
}
}
```

Debugging Steps

Note: Certain **show** commands are supported by the Output Interpreter Tool (registered customers only) , which allows you to view an analysis of **show** command output.

- Make sure the PIX configuration is working before adding AAA. If you cannot pass traffic prior to instituting authentication and authorization, you will not be able to do so afterwards.
- Enable logging in the PIX.
    - ♦ Logging console debugging should not be used on a heavily loaded system.
    - ♦ Logging buffered debugging can be used, then execute the **show logging** command.
    - ♦ Logging can also be sent to a syslog server and examined there.
- Turn on debugging on the TACACS+ or RADIUS servers (all servers have this option).

Network Diagram



PIX Configuration

```
PIX Version 5.1(1)
nameif ethernet0 outside security0
nameif ethernet1 inside security100
nameif ethernet2 pix/intf2 security10
enable password 8Ry2YjIyt7RRXU24 encrypted
passwd 2KFQnbNIdI.2KYOU encrypted
hostname pix3
fixup protocol ftp 21
fixup protocol http 80
fixup protocol smtp 25
fixup protocol h323 1720
fixup protocol rsh 514
fixup protocol sqlnet 1521
names
pager lines 24
no logging timestamp
no logging standby
logging console debugging
no logging monitor
no logging buffered
no logging trap
no logging history
logging facility 20
logging queue 512
interface ethernet0 auto
interface ethernet1 auto
interface ethernet2 auto shutdown
mtu outside 1500
mtu inside 1500
mtu pix/intf2 1500
ip address outside 99.99.99.1 255.255.255.0
ip address inside 10.31.1.75 255.255.255.0
ip address pix/intf2 127.0.0.1 255.255.255.255
no failover
failover timeout 0:00:00
failover ip address outside 0.0.0.0
```

```
failover ip address inside 0.0.0.0
failover ip address pix/intf2 0.0.0.0
arp timeout 14400
global (outside) 1 99.99.99.7−99.99.99.10 netmask 255.255.255.0
nat (inside) 1 10.31.1.0 255.255.255.0 0 0
static (inside,outside) 99.99.99.99 10.31.1.50 netmask 255.255.255.255 0 0
conduit permit icmp any any
conduit permit tcp any any
conduit permit udp any any
route outside 0.0.0.0 0.0.0.0 99.99.99.2 1
route inside 171.68.118.0 255.255.255.0 10.31.1.1 1
route inside 171.68.120.0 255.255.255.0 10.31.1.1 1
timeout xlate 3:00:00 conn 1:00:00 half−closed 0:10:00 udp 0:02:00
timeout rpc 0:10:00 h323 0:05:00
timeout uauth 0:05:00 absolute
aaa−server TACACS+ protocol tacacs+
aaa−server RADIUS protocol radius
aaa−server AuthInbound protocol tacacs+
aaa−server AuthInbound (inside) host 171.68.118.101 cisco timeout 5
aaa−server AuthOutbound protocol radius
aaa−server AuthOutbound (inside) host 171.68.118.101 cisco timeout 5
aaa authentication include telnet outbound 0.0.0.0 0.0.0.0 0.0.0.0 0.0.0.0 AuthOutbound
aaa authentication include telnet inbound 0.0.0.0 0.0.0.0 0.0.0.0 0.0.0.0 AuthInbound
aaa authentication include http outbound 0.0.0.0 0.0.0.0 0.0.0.0 0.0.0.0 AuthOutbound
aaa authentication include http inbound 0.0.0.0 0.0.0.0 0.0.0.0 0.0.0.0 AuthInbound
aaa authentication include ftp outbound 0.0.0.0 0.0.0.0 0.0.0.0 0.0.0.0 AuthOutbound
aaa authentication include ftp inbound 0.0.0.0 0.0.0.0 0.0.0.0 0.0.0.0 AuthInbound
no snmp−server location
no snmp−server contact
snmp−server community public
no snmp−server enable traps
floodguard enable
telnet timeout 5
terminal width 80
Cryptochecksum:b26b560b20e625c9e23743082484caca
: end
[OK]
```

Authentication Debug Examples from PIX

This section shows samples of authentication debugs for various scenarios.

Inbound

The outside user at 99.99.99.2 initiates traffic to inside 10.31.1.50 (99.99.99.99) and is authenticated through TACACS (that is, inbound traffic uses server list "AuthInbound" which includes TACACS server 171.68.118.101).

PIX Debug − Good Authentication − TACACS+

The example below shows a PIX debug with good authentication:

```
109001: Auth start for user '???' from
    99.99.99.2/11008 to 10.31.1.50/23
109011: Authen Session Start: user 'cse', sid 4
109005: Authentication succeeded for user 'cse'
    from 10.31.1.50/23 to 99.99.99.e
302001: Built inbound TCP connection 10 for
    faddr 99.99.99.2/11008 gaddr 99.99.)
```

PIX Debug − Bad Authentication (Username or Password) − TACACS+

The example below shows a PIX debug with bad authentication (username or password). The user sees three username/password sets, followed by this message: `Error: max number of tries exceeded`.

```
109001: Auth start for user '???' from
     99.99.99.2/11010 to 10.31.1.50/23
  109006: Authentication failed for user '' from
     10.31.1.50/23 to 99.99.99.2/11010 on
     interface outside
```

PIX Debug − Can Ping Server, No Response − TACACS+

The example below shows a PIX debug where the server is pingable, but not speaking to the PIX. The user sees the username once, but the PIX never asks for a password (this is on Telnet). The user sees `Error: Max number of tries exceeded`.

```
109001: Auth start for user '???' from 99.99.99.2/11011
     to 10.31.1.50/23
  109002: Auth from 10.31.1.50/23 to 99.99.99.2/11011 failed
     (server 171.68.118.101 failed) on interface outside
  109002: Auth from 10.31.1.50/23 to 99.99.99.2/11011 failed
     (server 171.68.118.101 failed) on interface outside
  109002: Auth from 10.31.1.50/23 to 99.99.99.2/11011 failed
     (server 171.68.118.101 failed) on interface outside
  109006: Authentication failed for user '' from 10.31.1.50/23
     to 99.99.99.2/11011 on interface outside
```

PIX Debug − Unable to Ping Server − TACACS+

The example below shows a PIX debug where the server is not pingable. The user sees the username once, but the PIX never asks for a password (this is on Telnet). The following messages are displayed: `Timeout to TACACS+ server` and `Error: Max number of tries exceeded` (a bogus server was swapped in the configuration).

```
111005: console end configuration: OK
  109001: Auth start for user '???' from
     99.99.99.2/11012 to 10.31.1.50/23
  109002: Auth from 10.31.1.50/23 to 99.99.99.2/11012
     failed (server 1.1.1.1 failed) on interface outside
  109002: Auth from 10.31.1.50/23 to 99.99.99.2/11012
     failed (server 1.1.1.1 failed) on interface outside
  109002: Auth from 10.31.1.50/23 to 99.99.99.2/11012
     failed (server 1.1.1.1 failed) on interface outside
  109006: Authentication failed for user '' from
     10.31.1.50/23 to 99.99.99.2/11012 on interface
     outside
```

PIX Debug − Good Authentication − RADIUS

The example below shows a PIX debug with good authentication:

```
109001: Auth start for user '???' from
     10.31.1.50/11008 to 99.99.99.2/23
  109011: Authen Session Start: user 'pixuser', sid 8
  109005: Authentication succeeded for user
     'pixuser' from 10.31.1.50/11008 to
     99.99.99.2/23 on interface inside
  302001: Built outbound TCP connection 16 for faddr
     99.99.99.2/23 gaddr 99.99.99.99/11008
     laddr 10.31.1.50/11008 (pixuser)
```

PIX Debug − Bad Authentication (Username or Password) − RADIUS

The example below shows a PIX debug with bad authentication (username or password). The user sees the request for a username and password, and has three opportunities to enter these. When the entry is unsuccessful, the following message is displayed: `Error: max number of tries exceeded`.

```
109001: Auth start for user '???' from 10.31.1.50/11010
     to 99.99.99.2/23
  109006: Authentication failed for user ''
     from 10.31.1.50/11010 to 99.99.99.2/23
     on interface inside
```

PIX Debug − Can Ping Server, Daemon Down − RADIUS

The example below shows a PIX debug where the server is pingable, but the daemon is down and will not communicate with the PIX. The user sees username, then password, the `RADIUS server failed` message, and the `Error: Max number of tries exceeded`. error message.

```
109001: Auth start for user '???' from 10.31.1.50/11011
     to 99.99.99.2/23
  ICMP unreachable (code 3) 171.68.118.101 > 10.31.1.75
  1ICMP unreachable (code 3) 171.68.118.101 > 10.31.1.75
  ICMP unreachable (code 3) 171.68.118.101 > 10.31.1.75
  ICMP unreachable (code 3) 171.68.118.101 > 10.31.1.75
  09002: Auth from 10.31.1.50/11011 to 99.99.99.2/23
     failed (server 171.68.118.101 failed) on interface inside
  109002: Auth from 10.31.1.50/11011 to 99.99.99.2/23 failed
     (server 171.68.118.101 failed) on interface inside
  109002: Auth from 10.31.1.50/11011 to 99.99.99.2/23 failed
     (server 171.68.118.101 failed) on interface inside
  109006: Authentication failed for user '' from 10.31.1.50/11011
     to 99.99.99.2/23 on interface inside
```

PIX Debug − Unable to Ping Server or Key/Client Mismatch − RADIUS

The example below shows a PIX debug where the server is not pingable or there is a Client/key mismatch. The user sees a username, password, the `Timeout to RADIUS server` message, and the `Error: Max number of tries exceeded` message a bogus server was swapped in the configuration).

```
109001: Auth start for user '???' from 10.31.1.50/11012
     to 99.99.99.2/23
  109002: Auth from 10.31.1.50/11012 to 99.99.99.2/23 failed
     (server 1.1.1.1 failed) on interface inside
  109002: Auth from 10.31.1.50/11012 to 99.99.99.2/23 failed
     (server 1.1.1.1 failed) on interface inside
  109002: Auth from 10.31.1.50/11012 to 99.99.99.2/23 failed
     (server 1.1.1.1 failed) on interface inside
  109006: Authentication failed for user '' from 10.31.1.50/11012
     to 99.99.99.2/23 on interface inside
```

Adding Authorization

If you decide to add authorization, since authorization is not valid without authentication, you need to require authorization for the same source and destination range.

```
aaa authorization telnet inbound 0.0.0.0 0.0.0.0 0.0.0.0 0.0.0.0 AuthInbound
aaa authorization http inbound 0.0.0.0 0.0.0.0 0.0.0.0 0.0.0.0 AuthInbound
aaa authorization ftp inbound 0.0.0.0 0.0.0.0 0.0.0.0 0.0.0.0 AuthInbound
```

Note that you do not add authorization for outgoing because outgoing traffic is authenticated with RADIUS, and RADIUS authorization is not valid.

Authentication and Authorization Debug Examples from PIX

PIX Debug − Good Authentication and Successful Authorization − TACACS+

The example below shows a PIX debug with good authentication and successful authorization:

```
109001: Auth start for user '???' from 99.99.99.2/11016
   to 10.31.1.50/23
109011: Authen Session Start: user 'cse', Sid 11
109005: Authentication succeeded for user 'cse'
   from 10.31.1.50/23 to 99.99.99.2/11016 on interface outside
109011: Authen Session Start: user 'cse', Sid 11
109007: Authorization permitted for user 'cse' from
   99.99.99.2/11016 to 10.31.1.50/23 on interface outside
302001: Built inbound TCP connection 19 for faddr 99.99.99.2/11016
   gaddr 99.99.99.99/23 laddr 10.31.1.50/23 (cse)
```

PIX Debug − Good Authentication, Failed Authorization − TACACS+

The example below shows the PIX debug with good authentication but failed authorization. Here the user also sees the message `Error: Authorization Denied`.

```
109001: Auth start for user '???' from
    99.99.99.2/11017 to 10.31.1.50/23
109011: Authen Session Start: user 'httponly',
   Sid 12
109005: Authentication succeeded for user 'httponly'
   from 10.31.1.50/23 to 99.99.99.2/11017 on
   interface outside
109008: Authorization denied for user 'httponly' from
   10.31.1.50/23 to 99.99.99.2/11017 on interface outside
```

Adding Accounting

TACACS+

```
aaa accounting include any inbound
0.0.0.0 0.0.0.0 0.0.0.0 0.0.0.0 AuthInbound
```

TACACS+ freeware output:

```
Tue Feb 22 08:52:20 2000  10.31.1.75 cse PIX
   99.99.99.2 start task_id=0x14
   foreign_ip=99.99.99.2 local_ip=10.31.1.50
   cmd=telnet
Tue Feb 22 08:52:25 2000  10.31.1.75 cse PIX
   99.99.99.2 stop task_id=0x14
   foreign_ip=99.99.99.2 local_ip=10.31.1.50
   cmd=telnet elapsed_time=5
   bytes_in=39 bytes_out=126
```

RADIUS

```
aaa accounting include any outbound
0.0.0.0 0.0.0.0 0.0.0.0 0.0.0.0 AuthOutbound
```

Merit RADIUS output:

```
Tue Feb 22 08:56:17 2000
        Acct-Status-Type = Start
        NAS-IP-Address = 10.31.1.75
        Login-IP-Host = 10.31.1.50
        Login-TCP-Port = 23
        Acct-Session-Id = 0x00000015
        User-Name = pixuser

        Tue Feb 22 08:56:24 2000
        Acct-Status-Type = Stop
        NAS-IP-Address = 10.31.1.75
        Login-IP-Host = 10.31.1.50
        Login-TCP-Port = 23
        Acct-Session-Id = 0x00000015
        Username = pixuser
        Acct-Session-Time = 6
        Acct-Input-Octets = 139
        Acct-Output-Octets = 36
```

Use of Exclude Command

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

```
aaa authentication exclude telnet inbound 0.0.0.0 0.0.0.0 99.99.99.100
 255.255.255.255 AuthInbound

aaa authorization exclude telnet inbound 0.0.0.0 0.0.0.0 99.99.99.100 255.255.255.255
AuthInbound
```

Max-sessions and Viewing Logged-in Users

Some TACACS+ and RADIUS servers have "max-session" or "view logged-in users" features. The ability to do max-sessions or check logged-in users is dependent on accounting records. When there is an accounting "start" record generated but no "stop" record, the TACACS+ or RADIUS server assumes the person is still logged in (that is, the user has a session through the PIX).

This works well for Telnet and FTP connections because of the nature of the connections. This does not work well for HTTP because of the nature of the connection. In the following example, a different network configuration is used, but the concepts are the same.

User Telnets through the PIX, authenticating on the way:

```
 171.68.118.100/1200 to 9.9.9.25 /23
(pix) 109011: Authen Session Start: user
   'cse', Sid 3
(pix) 109005: Authentication succeeded for user
   'cse' from 171.68.118.100/12 00 to 9.9.9.25/23
(pix) 302001: Built TCP connection 5 for faddr
   9.9.9.25/23 gaddr 9.9.9.10/12 00
   laddr 171.68.118.100/1200 (cse)
(server start account) Sun Nov 8 16:31:10 1998
   rtp-pinecone.rtp.cisco.com cse
PIX 171.68.118.100 start task_id=0x3 foreign_ip=9.9.9.25
   local_ip=171.68.118.100 cmd=telnet
```

Because the server has seen a start record but no stop record, at this point in time, the server shows that the

Telnet user is logged in. If the user attempts another connection that requires authentication (perhaps from another PC), and if max−sessions is set to **1** on the server for this user (assuming the server supports max−sessions), the connection is refused by the server.

The user goes about their Telnet or FTP business on the target host, then exits (spends ten minutes there):

```
pix) 302002: Teardown TCP connection 5 faddr
     9.9.9.25/80 gaddr 9.9.9.10/128
  1 laddr 171.68.118.100/1281 duration 0:00:00
     bytes 1907 (cse)
  (server stop account) Sun Nov 8 16:41:17 1998
     rtp−pinecone.rtp.cisco.com cse
  PIX 171.68.118.100 stop task_id=0x3 foreign_ip=9.9.9.25
     local_ip=171.68.118.100
  cmd=telnet elapsed_time=5 bytes_in=98 bytes_out=36
```

Whether uauth is 0 (that is, authenticate every time) or more (authenticate once and not again during uauth period), an accounting record is cut for every site accessed.

HTTP works differently due to the nature of the protocol. Below is an example of HTTP:

The user browses from 171.68.118.100 to 9.9.9.25 through the PIX:

```
(pix) 109001: Auth start for user '???' from
     171.68.118.100/1281 to 9.9.9.25 /80
  (pix) 109011: Authen Session Start: user 'cse', Sid 5
  (pix) 109005: Authentication succeeded for user
     'cse' from 171.68.118.100/12 81 to 9.9.9.25/80
  (pix) 302001: Built TCP connection 5 for faddr
     9.9.9.25/80 gaddr 9.9.9.10/12 81 laddr
     171.68.118.100/1281 (cse)
  (server start account) Sun Nov 8 16:35:34 1998
     rtp−pinecone.rtp.cisco.com cse
  PIX 171.68.118.100 start task_id=0x9 foreign_ip=9.9.9.25
     local_ip=171.68.118.100 cmd=http
  (pix) 302002: Teardown TCP connection 5 faddr
     9.9.9.25/80 gaddr 9.9.9.10/128
  1 laddr 171.68.118.100/1281 duration 0:00:00
     bytes 1907 (cse)
  (server stop account) Sun Nov 8 16:35.35 1998
     rtp−pinecone.rtp.cisco .com cse
  PIX 171.68.118.100 stop task_id=0x9 foreign_ip =9.9.9.25
  local_ip=171.68.118.100 cmd=http elapsed_time=0
     bytes_ in=1907 bytes_out=223
```

The user reads the downloaded web page.

The start record is posted at 16:35:34 and the stop record at 16:35:35. This download took one second (that is, there was less than one second between the start and the stop record). Is the user still logged in to the web site and the connection still open when the user is reading the web page? No. Will max−sessions or view logged−in users work here? No, because the connection time (the time between the "Built" and "Teardown") in HTTP is too short. The start and stop record is sub−second. There is not a start record without a stop record since the records occur at virtually the same instant. There will still be start and stop record sent to the server for every transaction whether uauth is set for 0 or something larger. However, max−sessions and view logged−in users will not work due to the nature of HTTP connections.

Authentication and Enabling on the PIX Itself

The previous discussion concerns authenticating Telnet (and HTTP, FTP) traffic through the PIX. Ensure Telnet to the PIX works without authentication on:

```
telnet 10.31.1.5 255.255.255.255
passwd ww
```

Then add the command to authenticate users Telnetting to the PIX:

```
aaa authentication telnet console AuthInbound
```

When users Telnet to the PIX, they are prompted for the Telnet password (**WW**). The PIX also requests the TACACS+ or RADIUS username and password. In this case since the AuthInbound server list is used, the PIX requests the TACACS+ username and password.

If the server is down, you can access the PIX by entering **pix** for the username, and then the enable password (**enable password** *whatever*). With the command:

```
aaa authentication enable console AuthInbound
```

The user is prompted for a username and password which is sent to the TACACS or RADIUS server. In this case since the AuthInbound server list is used, the PIX requests the TACACS+ username and password.

Since the authentication packet for enable is the same as the authentication packet for login, if the user can log in to the PIX with TACACS or RADIUS, they can enable through TACACS or RADIUS with the same username/password. This problem has been assigned Cisco bug ID CSCdm47044 (registered customers only) .

If the server is down, you can access PIX enable mode by inputting **pix** for the username and the normal enable password from the PIX (**enable password** *whatever*). If **enable password** *whatever* is not in the PIX configuration, enter **pix** for the username and press **Enter**. If the enable password is set but not known, a password recovery disk needs to be built to reset the password.

Changing the Prompt Users See

If you have the command:

```
auth-prompt PIX_PIX_PIX
```

users going through the PIX see the following sequence:

```
PIX_PIX_PIX [at which point one would enter the username]
   Password:[at which point one would enter the password]
```

On arrival at the ultimate destination, users would see the Username: and Password: prompt displayed by the destination box. This prompt only affects users going *through* the PIX, not to the PIX.

Note: There are no accounting records cut for access to the PIX.

Customizing the Message Users See on Success/Failure

If youh have the commands:

```
auth-prompt accept "GOOD_AUTH"
auth-prompt reject "BAD_AUTH"
```

then users see the following sequence on a failed/successful login through the PIX:

```
PIX_PIX_PIX
   Username: asjdkl
   Password: "BAD_AUTH"
   "PIX_PIX_PIX"
   Username: cse
   Password: "GOOD_AUTH"
```

Per−user Idle and Absolute Timeouts

This function is currently not working and the problem has been assigned Cisco bug ID CSCdp93492 (registered customers only) .

Virtual HTTP

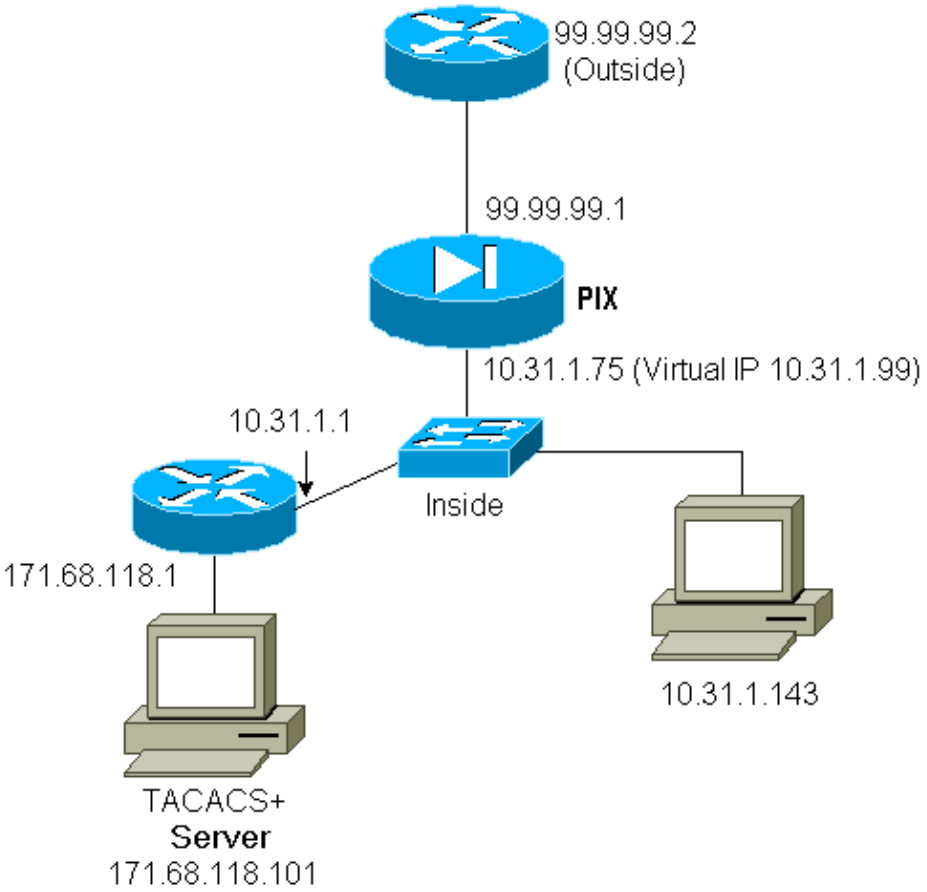
If authentication is required on sites outside the PIX as well as on the PIX itself, unusual browser behavior can sometimes be observed, since browsers cache the username and password.

To avoid this, you can implement virtual HTTP by adding an RFC 1918 address (that is, an address that is unroutable on the Internet, but valid and unique for the PIX inside network) to the PIX configuration using the following command:

```
virtual http #.#.#.# [warn]
```

When the user tries to go outside the PIX, authentication is required. If the warn parameter is present, the user receives a redirect message. The authentication is good for the length of time in the uauth. As indicated in the documentation, do not set the **timeout uauth** command duration to 0 seconds with virtual HTTP; this prevents HTTP connections to the real web server.

Virtual HTTP Outbound Example



PIX Configuration Virtual HTTP Outbound:

```
ip address outside 99.99.99.1 255.255.255.0
ip address inside 10.31.1.75 255.255.255.0
global (outside) 1 99.99.99.7-99.99.99.10 netmask 255.255.255.0
timeout uauth 01:00:00
aaa authentication include http outbound 0.0.0.0 0.0.0.0 0.0.0.0 0.0.0.0 AuthOutbound
aaa-server RADIUS protocol radius
aaa-server AuthOutbound protocol radius
aaa-server AuthOutbound (inside) host 171.68.118.101 cisco timeout 5
virtual http 10.31.1.99
```

Virtual Telnet

It is possible to configure the PIX to authenticate all inbound and outbound, but it is not a good idea because some protocols, such as mail, are not easily authenticated. When a mail server and Client try to communicate through the PIX when all traffic through the PIX is being authenticated, PIX syslog for unauthenticatible protocols shows messages such as:

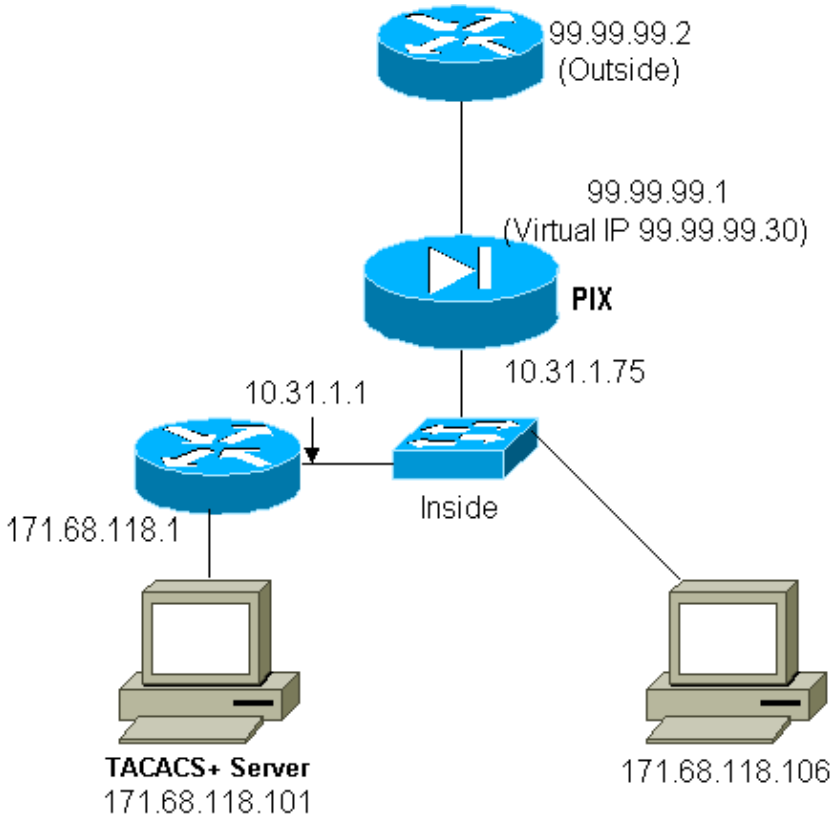
```
109013: User must authenticate before using
    this service
109009: Authorization denied from 171.68.118.106/49
    to 9.9.9.10/11094    (not authenticated)
```

However, if there is really a need to authenticate some kind of unusual service, this can be done by use of the **virtual telnet** command. This command allows authentication to occur to the virtual Telnet IP address. After this authentication, the traffic for the unusual service can go to the real server.

In this example, you want TCP port 49 traffic to flow from outside host 99.99.99.2 to inside host 171.68.118.106. Since this traffic is not really authenticatable, set up a virtual Telnet. For virtual Telnet, there

must be an associated static. Here, both 99.99.99.20 and 171.68.118.20 are virtual addresses.

Virtual Telnet Inbound



PIX Configuration Virtual Telnet Inbound

```
ip address outside 99.99.99.1 255.255.255.0
ip address inside 10.31.1.75 255.255.255.0
static (inside,outside) 99.99.99.20 171.68.118.20 netmask 255.255.255.255 0 0
static (inside,outside) 99.99.99.30 171.68.118.106 netmask 255.255.255.255 0 0
conduit permit tcp host 99.99.99.20 eq telnet any
conduit permit tcp host 99.99.99.30 eq tacacs any
aaa-server TACACS+ protocol tacacs+
aaa-server Incoming protocol tacacs+
aaa-server Incoming (inside) host 171.68.118.101 cisco timeout 5
aaa authentication include telnet inbound 0.0.0.0 0.0.0.0 0.0.0.0 0.0.0.0
Incoming
aaa authentication include tcp/49 inbound 0.0.0.0 0.0.0.0 0.0.0.0 0.0.0.0
Incoming
virtual telnet 99.99.99.20
```

PIX Debug Virtual Telnet Inbound

The user at 99.99.99.2 must first authenticate by Telnetting to the 99.99.99.20 address on the PIX:

```
109001: Auth start for user '???' from
    99.99.99.2/22530 to 171.68.118.20/23
 109011: Authen Session Start: user 'cse', Sid 13
 109005: Authentication succeeded for user
    'cse' from 171.68.118.20/23 to
    99.99.99.2/22530 on interface outside
```

After the successful authentication, the **show uauth** command shows the user has "time on the meter":

```
pixfirewall# show uauth
                        Current    Most Seen
Authenticated Users       1          2
Authen In Progress        0          1
user 'cse' at 99.99.99.2, authenticated
   absolute   timeout: 0:05:00
   inactivity timeout: 0:00:00
```

And when the device at 99.99.99.2 wants to send TCP/49 traffic to the device at 171.68.118.106:

```
302001: Built inbound TCP connection 16
    for faddr 99.99.99.2/11054 gaddr
    99.99.99.30/49 laddr 171.68.118.106/49 (cse)
```

Authorization can be added:

```
aaa authorization include tcp/49 inbound
0.0.0.0 0.0.0.0 0.0.0.0 0.0.0.0 AuthInbound
```

so that when TCP/49 traffic is attempted through the PIX, the PIX also sends the authorization query to the server:

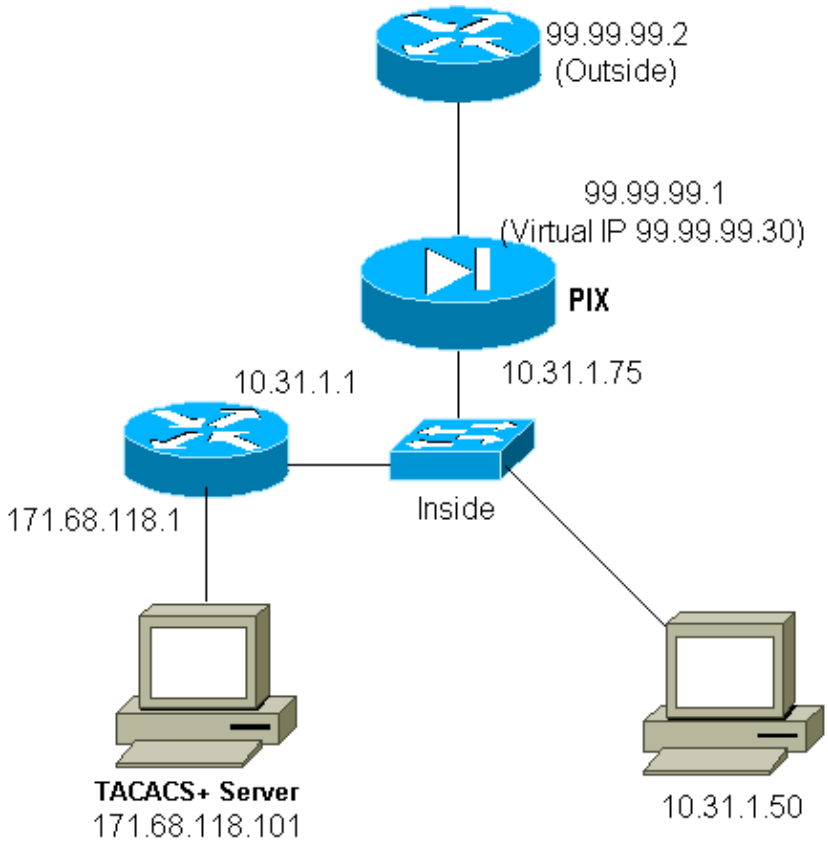
```
109007: Authorization permitted for user 'cse'
    from 99.99.99.2/11057 to 171.68.118.106/49
    on interface outside
```

On the TACACS+ server, this is seen as:

```
service=shell,
  cmd=tcp/49,
  cmd-arg=171.68.118.106
```

Virtual Telnet Outbound

Since outbound traffic is allowed by default, no static is required for use of virtual Telnet outbound. In the following example, the inside user at 10.31.1.50 Telnets to virtual 99.99.99.30 and authenticates; the Telnet connection is immediately dropped. Once authenticated, TCP traffic is allowed from 10.31.1.50 to the server at 99.99.99.2:



PIX Configuration Virtual Telnet Outbound:

```
ip address outside 99.99.99.1 255.255.255.0
ip address inside 10.31.1.75 255.255.255.0
global (outside) 1 99.99.99.7-99.99.99.10 netmask 255.255.255.0
timeout uauth 0:05:00 absolute
aaa-server RADIUS protocol radius
aaa-server AuthOutbound protocol radius
aaa-server AuthOutbound (inside) host 171.68.118.101 cisco timeout 5
aaa authentication include telnet outbound 0.0.0.0 0.0.0.0 0.0.0.0 0.0.0.0 AuthOutbound
aaa authentication include tcp/49 outbound 0.0.0.0 0.0.0.0 0.0.0.0 0.0.0.0 AuthOutbound
virtual telnet 99.99.99.30
```

Note: There is no authorization since this is RADIUS.

PIX Debug Virtual Telnet Outbound:

```
109001: Auth start for user '???' from 10.31.1.50/11034
    to 99.99.99.30/23
 109011: Authen Session Start: user 'pixuser', Sid 16
 109005: Authentication succeeded for user 'pixuser'
    from 10.31.1.50/11034 to 99.99.99.30/23 on interface
    inside
 302001: Built outbound TCP connection 18 for faddr
    99.99.99.2/49 gaddr 99.99.99.8/11036 laddr
    10.31.1.50/11036 (pixuser)
 302002: Teardown TCP connection 18 faddr 99.99.99.2/49
    gaddr 99.99.99.8/11036 laddr 10.31.1.50/11036
    duration 0:00:02 bytes 0 (pixuser)
```

Virtual Telnet Logout

When users Telnet to the virtual Telnet IP address, the **show uauth** command shows their uauth. If the users want to prevent traffic from going through after their sessions are finished when there is time left in the uauth, they need to Telnet to the virtual Telnet IP address again. This toggles the session off.

After first authentication:

```
pix3# show uauth
                        Current    Most Seen
Authenticated Users       1          2
Authen In Progress        0          1
user 'pixuser' at 10.31.1.50, authenticated
   absolute   timeout: 0:05:00
   inactivity timeout: 0:00:00
pix3# 109001: Auth start for user 'pixuser' from
   10.31.1.50/11038 to 99.99.99.30/23
109005: Authentication succeeded for user 'pixuser'
   from 10.31.1.50/11038 to 99.99.99.30/23 on
   interface inside
```

After second authentication (that is, the hole is toggled closed):

```
pix3# show uauth
                        Current    Most Seen
Authenticated Users       0          2
Authen In Progress        0          1
```

Port Authorization

Authorization is allowed for port ranges (like TCP/30−100). If virtual Telnet is configured on the PIX and authorization for a range of ports, once the hole is opened with virtual Telnet, the PIX issues a **tcp/30−100** command to the TACACS+ server for authorization:

```
static (inside,outside) 99.99.99.75 10.31.1.50 netmask 255.255.255.255 0 0
conduit permit tcp host 99.99.99.75 host 99.99.99.2
static (inside,outside) 99.99.99.75 10.31.1.50 netmask 255.255.255.255 0 0
virtual telnet 99.99.99.75
aaa authentication include any inbound 0.0.0.0 0.0.0.0 0.0.0.0 0.0.0.0 AuthInbound
aaa authorization include tcp/30−100 inbound 0.0.0.0 0.0.0.0 0.0.0.0 0.0.0.0 AuthInbound
virtual telnet 99.99.99.30
```

TACACS+ Freeware Server Configuration:

```
user = anyone {
         login = cleartext "anyone"
         cmd = tcp/30−100 {
         permit 10.31.1.50
         }
         }
```

AAA Accounting for Traffic Other Than HTTP, FTP, and Telnet

After making sure virtual Telnet worked to allow TCP/49 traffic to the host inside the network, we decided we wanted accounting for this, so we added:

```
aaa accounting include any inbound
0.0.0.0 0.0.0.0 0.0.0.0 0.0.0.0 AuthInbound
```

This results in having an accounting record cut when the tcp/49 traffic goes through (this example is from the TACACS+ freeware):

```
Sun Feb 27 05:24:44 2000 10.31.1.75 cse PIX
    99.99.99.2 start task_id=0x14 foreign_ip=99.99.99.2 local_ip=171.68.118.106
    cmd=tcp/49
```

Extended Authentication (Xauth)

Sample Configurations

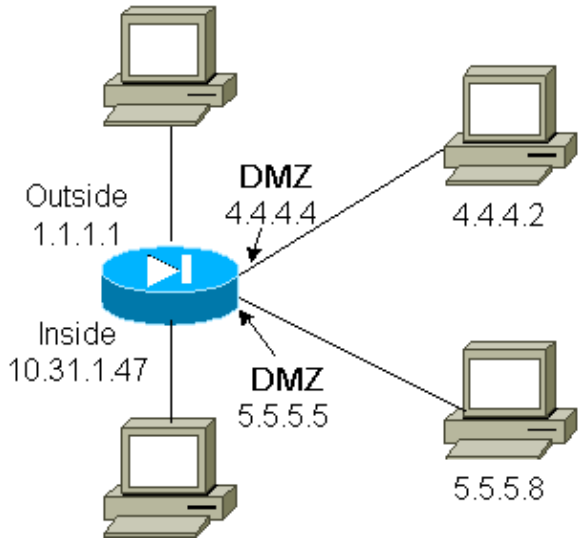
- Terminating IPSec Tunnels on Multiple Cisco Secure PIX Firewall Interfaces with Xauth
- IPSec Between Cisco Secure PIX Firewall and a VPN Client with Extended Authentication

Authentication on the DMZ

To authenticate users going from one DMZ interface to another, tell the PIX to authenticate traffic for the named interfaces. On our PIX the arrangement is:

```
least secure

PIX outside (security0) = 1.1.1.1

pix/intf4 (DMZ – security20) = 4.4.4.4 & device 4.4.4.2

pix/intf5 (DMZ – security25) = 5.5.5.5 & device 5.5.5.8

(static to 4.4.4.15)

PIX inside (security100) = 10.31.1.47

 most secure
```

Network Diagram



PIX Configuration

We want to authenticate Telnet traffic between pix/intf4 and pix/intf5:

```
nameif ethernet0 outside security0
nameif ethernet1 inside security100
(nameif ethernet2 pix/intf2 security10
nameif ethernet3 pix/intf3 security15)
nameif ethernet4 pix/intf4 security20
nameif ethernet5 pix/intf5 security25
ip address outside 1.1.1.1 255.255.255.0
ip address inside 10.31.1.47 255.255.255.0
(ip address pix/intf2 127.0.0.1 255.255.255.255
ip address pix/intf3 127.0.0.1 255.255.255.255)
ip address pix/intf4 4.4.4.4 255.255.255.0
ip address pix/intf5 5.5.5.5 255.255.255.0
static (pix/intf5,pix/intf4) 4.4.4.15 5.5.5.8 netmask 255.255.255.255 0 0
aaa authentication telnet pix/intf4 5.5.5.0 255.255.255.0
4.4.4.0 255.255.255.0 AuthInbound
aaa authentication telnet pix/intf5 5.5.5.0 255.255.255.0
4.4.4.0 255.255.255.0 AuthInbound
aaa-server TACACS+ protocol tacacs+
aaa-server AuthInbound protocol tacacs+
aaa-server AuthInbound (inside) host 171.68.118.101 cisco timeout 5
```

Xauth Accounting

If the **sysopt connection permit−ipsec** command, not the **sysopt ipsec pl−compatible** command, is configured in the PIX with xauth, accounting is valid for TCP connections, but not ICMP or UDP.

Related Information

- **PIX Product Support Page**
- **Documentation for PIX Firewall**
- **PIX Command Reference**
- **RADIUS Support Page**
- **RADIUS in IOS Documentation**
- **Requests for Comments (RFCs)**
- **Cisco Secure UNIX Support Page**
- **Documentation for Cisco Secure ACS for UNIX**
- **Documentation for Cisco Secure ACS for Windows**
- **Cisco Secure ACS for Windows Support Page**
- **Technical Support − Cisco Systems**

Updated: May 14, 2009 Document ID: 4613