Configuring PIX 5.0.x: TACACS+ and RADIUS

Document ID: 13820

Contents

Introduction

RADIUS and TACACS+ authentication may be done for FTP, Telnet, and HTTP connections. Authentication for other less common TCP protocols can usually be made to work.

TACACS+ authorization is supported. RADIUS authorization is not. Changes in the PIX 5.0 authentication, authorization, and accounting (AAA) over the previous version include AAA accounting for traffic other than HTTP, FTP, and Telnet.

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, refer to the Cisco Technical Tips Conventions.

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.

As an example, assume 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. Tell the PIX to authenticate outbound traffic and give all six users IDs on the TACACS+/RADIUS security server. With simple *authentication*, these six users can be authenticated with username and password, then go out. The other ninety−four users are unable to go out. The PIX prompts users for username/password, then passes their username and password to the TACACS+/RADIUS security server. Depending on the response, it opens or denies the connection. These six users can do FTP, Telnet, or HTTP.

On the other hand, assume *one* of these three users, "Terry," is not to be trusted. You would like to allow Terry to do FTP, but not HTTP or Telnet to the outside. This means you need to add *authorization*. That is, authorizing *what* users can do in addition to authenticating *who* they are. When you add *authorization* to the PIX, the PIX first sends Terry's username and password to the security server, then sends an authorization request telling the security server what "*command*" Terry is trying to do. With the server set up properly, Terry can be allowed to "FTP 1.2.3.4" but is denied the ability to "HTTP" or "Telnet" anywhere.

What the User Sees with Authentication/Authorization On

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

- **Telnet** – The user sees a username prompt display, followed by 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 displayed in the browser that requests 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 will show this phenomenon. If Telnet and FTP seem to work normally, but HTTP connections do not, this is why.

Security Server Configurations Used for All Scenarios

Cisco Secure UNIX TACACS Server Configuration

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 graphical user interface (GUI) to add the PIX IP and key to the network access server (NAS) list.

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

Cisco Secure Windows 2.x RADIUS

Follow these steps:

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 in the NAS Configuration 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 **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 you want to allow Telnet to specific sites, enter the IP(s) in the argument section in the form "permit #.#.#.#". To allow Telnet to all sites, 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 **Shell exec** (to give exec privileges).
2. To add authorization to the PIX, click **Deny unmatched IOS commands** at the bottom of the group setup.
3. Select **Add/Edit new command** for each command you want to allow (for example, Telnet).
4. If you want to allow Telnet to specific sites, enter permit IP(s) in the argument rectangle (for example, "permit 1.2.3.4"). To allow Telnet to all sites, click **Allow all unlisted arguments**.
5. Click **finish editing command**.

6. Perform the previous steps for each of the allowed commands (for example, Telnet, HTTP and/or FTP).
7. Add the PIX IP in the NAS Configuration GUI section.

Livingston RADIUS Server Configuration

Add the PIX IP and key to clients file.

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

Merit RADIUS Server Configuration

Add the PIX IP and key to the clients file.

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

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

- Make sure that the PIX configurations work before you add AAA.

  ♦ If you cannot pass traffic before instituting authentication and authorization, you will not be able to do so afterwards.
- Enable logging in the PIX

  ♦ The **logging console debugging** command *should not* be used on a heavy loaded system.
  ♦ The **logging buffered debugging** command can be used. Output from the **show logging** or **logging** commands can be sent to a syslog server and examined.
- Make sure that debugging is on for the TACACS+ or RADIUS servers. All servers have this option.

Network Diagram

Outside:

192.1.1.1

192.1.1.254

PIX

Inside:

10.31.1.200

10.31.1.1

10.31.1.5

171.68.118.1

171.68.118.143

171.68.118.133

Tacacs Server

Radius Server
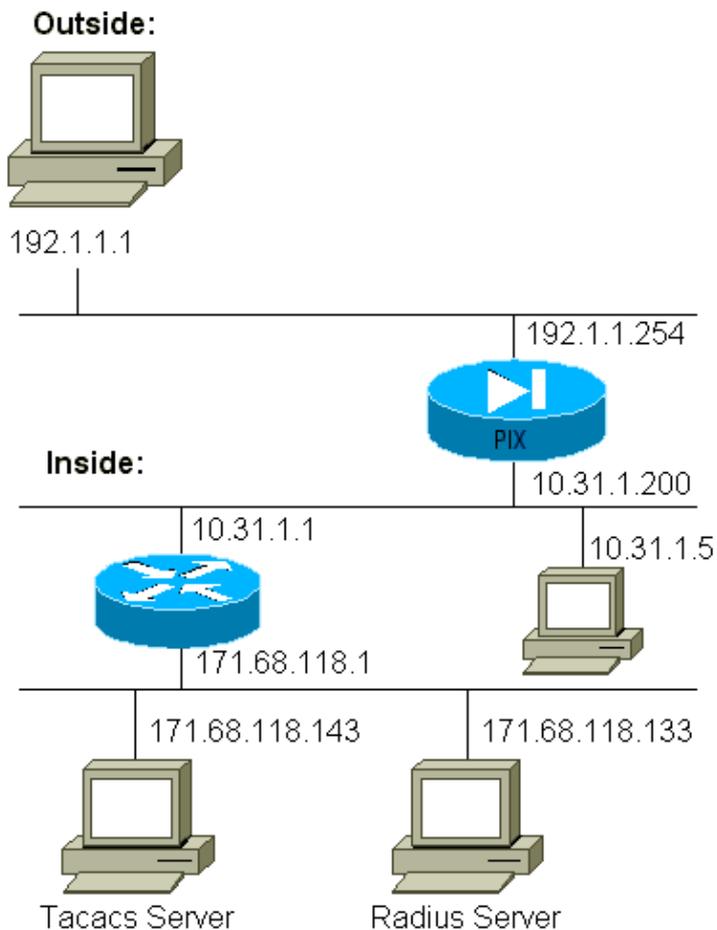
PIX Configuration

```
pix-5# write terminal
 nameif ethernet0 outside security0
 nameif ethernet1 inside security100
 enable password 8Ry2YjIyt7RRXU24 encrypted
 passwd 2KFQnbNIdI.2KYOU encrypted
 hostname pixfirewall
 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
 name 1.1.1.1 abcd
 name 1.1.1.2 a123456789
 name 1.1.1.3 a123456789123456
 pager lines 24
 logging timestamp
 no logging standby
 logging console debugging
 no logging monitor
 logging buffered debugging
 no logging trap
 logging facility 20
 logging queue 512
 interface ethernet0 auto
 interface ethernet1 auto
```

```
mtu outside 1500
mtu inside 1500
ip address outside 192.1.1.254 255.255.255.0
ip address inside 10.31.1.200 255.255.255.0
no failover
failover timeout 0:00:00
failover ip address outside 0.0.0.0
failover ip address inside 0.0.0.0
arp timeout 14400
global (outside) 1 192.1.1.10-192.1.1.20 netmask 255.255.255.0
static (inside,outside) 192.1.1.25 171.68.118.143 netmask 255.255.255.255 0 0
static (inside,outside) 192.1.1.30 10.31.1.5 netmask 255.255.255.255 0 0
conduit permit tcp any any
conduit permit icmp any any
conduit permit udp any any
no rip outside passive
no rip outside default
no rip inside passive
no rip inside default
route inside 171.68.118.0 255.255.255.0 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 0:00:00 absolute
aaa-server TACACS+ protocol tacacs+
aaa-server RADIUS protocol radius
aaa-server AuthInbound protocol tacacs+
aaa-server AuthInbound (inside) host 171.68.118.143 cisco timeout 5
aaa-server AuthOutbound protocol radius
aaa-server AuthOutbound (inside) host 171.68.118.133 cisco timeout 5
aaa authentication telnet outbound 0.0.0.0 0.0.0.0 0.0.0.0 0.0.0.0 AuthOutbound
aaa authentication telnet inbound 0.0.0.0 0.0.0.0 0.0.0.0 0.0.0.0 AuthInbound
aaa authentication http outbound 0.0.0.0 0.0.0.0 0.0.0.0 0.0.0.0 AuthOutbound
aaa authentication http inbound 0.0.0.0 0.0.0.0 0.0.0.0 0.0.0.0 AuthInbound
aaa authentication ftp outbound 0.0.0.0 0.0.0.0 0.0.0.0 0.0.0.0 AuthOutbound
aaa authentication 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
telnet timeout 5
terminal width 80
Cryptochecksum:fef4bfc9801d7692dce0cf227fe7859b
: end
```

Authentication Debug Examples from PIXAuthentication Debug Examples from PIX

In these debug examples:

Outbound

The inside user at 10.31.1.5 initiates traffic to outside 192.1.1.1 and is authenticated through TACACS+. The outbound traffic uses server list "AuthOutbound" which includes RADIUS server 171.68.118.133.

Inbound

The outside user at 192.1.1.1 initiates traffic to inside 10.31.1.5 (192.1.1.30) and is authenticated through TACACS. Inbound traffic uses server list "AuthInbound" which includes TACACS server 171.68.118.143).

PIX Debug – Good Authentication – TACACS+

This example shows a PIX debug with good authentication:

```
pixfirewall# 109001: Auth start for user "???" from 192.1.1.1/13155
    to 10.31.1.5/23
 109011: Authen Session Start: user 'pixuser', sid 6
 109005: Authentication succeeded for user 'pixuser' from 10.31.1.5/23
    to 192.1.1.1/13155
 109012: Authen Session End: user 'pixuser', Sid 6, elapsed 1 seconds
 302001: Built inbound TCP connection 6 for faddr 192.1.1.1/13155
    gaddr 192.1.1.30/23 laddr 10.31.1.5/23 (pixuser)
```

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

This example shows PIX debug with bad authentication (username or password). The user sees four username/password sets and the message "`Error: max number of tries exceeded.`"

```
pixfirewall# 109001: Auth start for user '???' from 192.1.1.1/13157
    to 10.31.1.5/23
 109006: Authentication failed for user '' from 10.31.1.5/23
    to 192.1.1.1/13157
```

PIX debug – Can Ping Server, No Response – TACACS+

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

```
Auth start for user '???' from 192.1.1.1/13159 to
    10.31.1.5/23
 pixfirewall# 109002: Auth from 10.31.1.5/23 to 192.1.1.1/13159
    failed (server 171.68.118.143 failed)
 109002: Auth from 10.31.1.5/23 to 192.1.1.1/13159 failed
    (server 171.68.118.143 failed)
 109002: Auth from 10.31.1.5/23 to 192.1.1.1/13159 failed
    (server 171.68.118.143 failed)
 109006: Authentication failed for user '' from 10.31.1.5/23
    to 192.1.1.1/13159
```

PIX Debug – Unable to Ping Server – TACACS+

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

```
 109001: Auth start for user '???' from 192.1.1.1/13158
    to 10.31.1.5/23
 109002: Auth from 10.31.1.5/23 to 192.1.1.1/13158 failed
    (server 171.68.118.143 failed)
 109002: Auth from 10.31.1.5/23 to 192.1.1.1/13158 failed
    (server 171.68.118.143 failed)
 109002: Auth from 10.31.1.5/23 to 192.1.1.1/13158 failed
    (server 171.68.118.143 failed)
 109006: Authentication failed for user '' from 10.31.1.5/23
    to 192.1.1.1/13158
```

PIX Debug – Good Authentication – RADIUS

This example shows a PIX debug with good authentication:

```
109001: Auth start for user '???' from 10.31.1.5/11074
    to 192.1.1.1/23
 109011: Authen Session Start: user 'pixuser', Sid 7
 109005: Authentication succeeded for user 'pixuser'
    from 10.31.1.5/11074 to 192.1.1.1/23
 109012: Authen Session End: user 'pixuser', Sid 7,
    elapsed 1 seconds
 302001: Built outbound TCP connection 7 for faddr 192.1.1.1/23
    gaddr 192.1.1.30/11074 laddr 10.31.1.5/11074 (pixuser)
```

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

This example shows a PIX debug with bad authentication (username or password). The user sees a request for Username and Password. The user has three opportunities for successful Username/Password entry.

```
 – 'Error: max number of tries exceeded'
 pixfirewall# 109001: Auth start for user '???' from
    192.1.1.1/13157 to 10.31.1.5/23
 109001: Auth start for user '???' from 10.31.1.5/11075
    to 192.1.1.1/23
 109002: Auth from 10.31.1.5/11075 to 192.1.1.1/23 failed
    (server 171.68.118.133 failed)
 109002: Auth from 10.31.1.5/11075 to 192.1.1.1/23 failed
    (server 171.68.118.133 failed)
 109002: Auth from 10.31.1.5/11075 to 192.1.1.1/23 failed
    (server 171.68.118.133 failed)
 109006: Authentication failed for user '' from 10.31.1.5/11075
    to 192.1.1.1/23
```

Ping Debug – Can Ping Server, Daemon Down – RADIUS

This example 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, password, and the messages "RADIUS server failed" and "Error: Max number of tries exceeded."

```
 pixfirewall# 109001: Auth start for user '???'
    from 10.31.1.5/11076 to 192.1.1.1/23
 109002: Auth from 10.31.1.5/11076 to 192.1.1.1/23 failed
    (server 171.68.118.133 failed)
 109002: Auth from 10.31.1.5/11076 to 192.1.1.1/23 failed
    (server 171.68.118.133 failed)
 109002: Auth from 10.31.1.5/11076 to 192.1.1.1/23 failed
    (server 171.68.118.133 failed)
 109006: Authentication failed for user '' from 10.31.1.5/11076
    to 192.1.1.1/23
```

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

This example shoes a PIX debug where the server is not pingable or there is a key/client mismatch. The user sees Username, password, and the messages "Timeout to RADIUS server" and "Error: Max number of tries exceeded" (a bogus server was swapped in the configuration).

```
 109001: Auth start for user '???' from 10.31.1.5/11077
    to 192.1.1.1/23
 109002: Auth from 10.31.1.5/11077 to 192.1.1.1/23 failed
    (server 100.100.100.100 failed)
```

```
109002: Auth from 10.31.1.5/11077 to 192.1.1.1/23 failed
   (server 100.100.100.100 failed)
109002: Auth from 10.31.1.5/11077 to 192.1.1.1/23 failed
   (server 100.100.100.100 failed)
109006: Authentication failed for user '' from 10.31.1.5/11077
   to 192.1.1.1/23
```

Add Authorization

If you decide to add authorization, you will require authorization for the same source and destination range (since authorization is not valid without authentication):

```
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 authorization is not added 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+

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

```
109011: Authen Session Start: user 'pixuser', Sid 8
 109007: Authorization permitted for user 'pixuser'
    from 192.1.1.1/13160 to 10.31.1.5/23
 109012: Authen Session End: user 'pixuser', Sid 8,
    elapsed 1 seconds
 302001: Built inbound TCP connection 8 for faddr 192.1.1.1/13160
    gaddr 192.1.1.30/23 laddr 10.31.1.5/23 (pixuser)
```

PIX Debug – Good Authentication, Failed Authorization – TACACS+

This example shows a PIX debug with good authentication but with failed authorization. Here the user also sees the message "Error: Authorization Denied."

```
109001: Auth start for user '???' from 192.1.1.1/13162
    to 10.31.1.5/23
 109011: Authen Session Start: user 'userhttp', Sid 10
 109005: Authentication succeeded for user 'userhttp'
    from 10.31.1.5/23 to 192.1.1.1/13162
 109008: Authorization denied for user 'userhttp'
    from 10.31.1.5/23 to 192.1.1.1/13162
 109012: Authen Session End: user 'userhttp', Sid 10,
    elapsed 1 seconds
 302010: 0 in use, 2 most used
```

Add Accounting

TACACS+

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

Debug look the same whether accounting is on or off. However, at the time of the "Built," a "start" accounting record is sent. At the time of the "Teardown," a "stop" accounting record is sent.

The TACACS+ accounting records look like this output (these are from Cisco Secure NT, hence the comma–delimited format):

```
04/26/2000,01:31:22,pixuser,Default Group,192.1.1.1,
  start,,,,,,,0x2a,,PIX,10.31.1.200,telnet,6,
  Login,1,,,1,,,,,,,,,,,local_ip=10.31.1.5 foreign_ip=192.1.1.1,
  ,, ,,,,,,,,,zekie,,,,,,,,^
04/26/2000,01:31:26,pixuser,Default Group,192.1.1.1,stop,4,
  ,36,82,,,0x2a,,PIX,10.31.1.200,telnet,6,
  Login,1,,,1,,,,,,,,,,,local_ip=10.31.1.5 foreign_ip=192.1.1. 1,
  ,,,,,,,,,,,,zekie,,,,,,,
```

RADIUS

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

Debug looks the same whether accounting is on or off. However, at the time of the "Built," a "start" accounting record is sent. At the time of the "Teardown," a "stop" accounting record is sent.

RADIUS accounting records look like this output (these are from Cisco Secure UNIX; ones in Cisco Secure NT may be comma–delimited instead):

```
radrecv: Request from host a1f01c8 code=4, id=18, length=65
Acct-Status-Type = Start
Client-Id = 10.31.1.200
Login-Host = 10.31.1.5
Login-TCP-Port = 23
Acct-Session-Id = "0x0000002f"
User-Name = "pixuser"
Sending Accounting Ack of id 18 to a1f01c8 (10.31.1.200)
radrecv: Request from host a1f01c8 code=4, id=19, length=83
Acct-Status-Type = Stop
Client-Id = 10.31.1.200
Login-Host = 10.31.1.5
Login-TCP-Port = 23
Acct-Session-Id = "0x0000002f"
Username = "pixuser"
Acct-Session-Time = 7
```

Use of Except Command

In our network, if we decide that a particular source and/or destination does not need authentication, authorization, or accounting, we can do something like this output:

```
aaa authentication except inbound 192.1.1.1 255.255.255.255
   0.0.0.0 0.0.0.0 AuthInbound
```

If you are "excepting" a box from authentication and have authorization on, you must also except the box from authorization.

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 (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 this example output, a different network configuration is used, but the concepts are the same.

The user Telnets through the PIX, authenticating on the way:

```
(pix) 109001: Auth start for user '???' from 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
```

Since 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 on with the Telnet or FTP business on the target host, then exits (spends 10 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 (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. This output shows 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 posted at 16:35:34, and the stop record posted 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 they are 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 will not be 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 do not work due to the nature of HTTP connections.

Authentication and Enabling on the PIX Itself

The previous discussion described authenticating Telnet (and HTTP, FTP) traffic *through* the PIX. We make sure Telnet *to* the PIX works *without* authentication on:

```
    telnet 10.31.1.5 255.255.255.255 passwd ww

    aaa authentication telnet console AuthInbound
```

When users Telnet to the PIX, they are prompted for the Telnet password (**ww**). Then the PIX also requests the TACACS+ (in this case, since the "AuthInbound" server list is used) or RADIUS username and password. If the server is down, you can get into the PIX by entering **pix** for the username, and then the enable password (**enable password** *whatever*) to gain access.

With this command:

```
    aaa authentication enable console AuthInbound
```

the user is prompted for a username and password, which is sent to the TACACS (in this case, since the "AuthInbound" server list is used, the request goes to the TACACS server) or RADIUS server. 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) .

Authentication on the Serial Console

The **aaa authentication serial console AuthInbound** command requires authentication verification in order to access the serial console of the PIX.

When the user performs configuration commands from the console, syslog messages are cut (assuming the PIX is configured to send syslog at the debug level to a syslog host). This is an example of what is displayed on the syslog server:

```
    logmsg: pri 245, flags 0, from [10.31.1.200.2.2], msg Nov 01 1999
     03:21:14: %PIX-5-111008: User 'pixuser' executed the 'logging' command.
```

Change the Prompt that Users See

If you have the **auth–prompt PIX_PIX_PIX** command, users that go through the PIX see this sequence:

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

Upon arrival at the ultimate destination box, the "Username:" and "Password:" prompt is displayed. This prompt affects only users going *through* the PIX, not *to* the PIX.

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

Customize the Message Users See on Success/Failure

If you have the commands:

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

users see this 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

Idle and absolute uauth timeouts can be sent down from the TACACS+ server on a per−user basis. If all the users in your network are to have the same "timeout uauth," do not implement this! But if you need different uauths per−user, continue to read.

In this example, the **timeout uauth 3:00:00** command is used. Once a person authenticates, they do not have to re−authenticate for three hours. However, if you set up a user with this profile and have TACACS AAA *authorization* on in the PIX, the idle and absolute timeouts in the user profile override the timeout uauth in the PIX for that user. This does not mean that the Telnet session through the PIX is disconnected after the idle/absolute timeout. It just controls whether re−authentication takes place.

This profile comes from TACACS+ freeware:

```
user = timeout {
default service = permit
login = cleartext "timeout"
service = exec {
timeout = 2
idletime = 1
}
}
```

After authentication, execute a **show uauth** command on the PIX:

```
pix-5# show uauth
                        Current     Most Seen
 Authenticated Users       1            1
 Authen In Progress        0            1
 user 'timeout' at 10.31.1.5, authorized to:
    port 11.11.11.15/telnet
    absolute   timeout: 0:02:00
```

```
      inactivity timeout: 0:01:00
```

After the user sits idle for one minute, the debug on the PIX shows:

```
109012: Authen Session End: user 'timeout', Sid 19, elapsed 91 seconds
```

The user has to re−authenticate when it returns to the same target host or a different host.
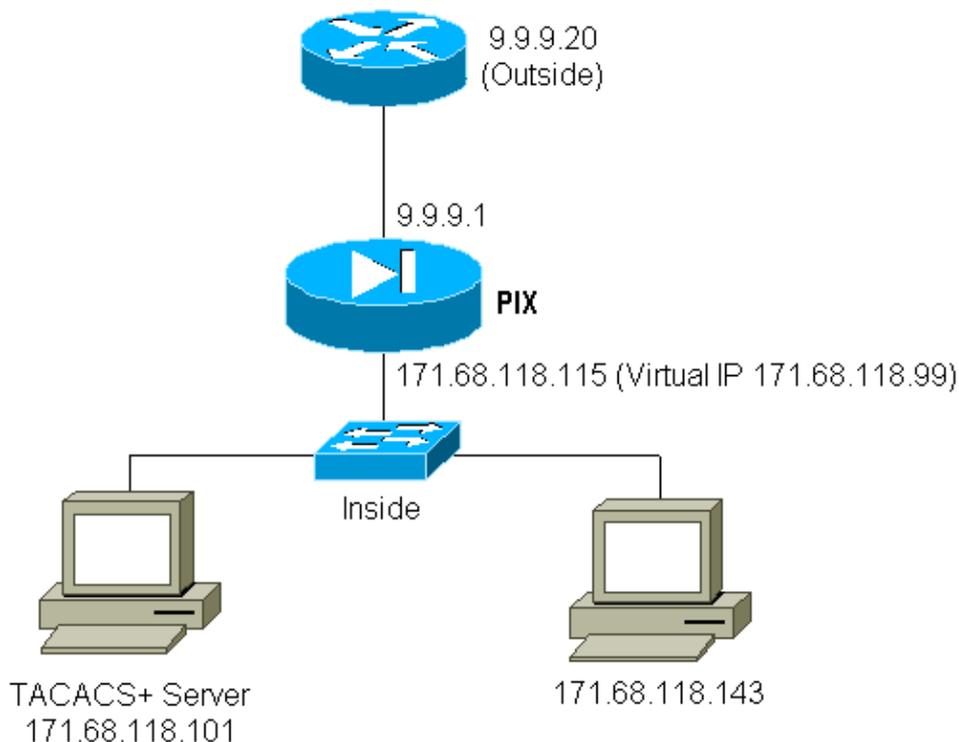
Virtual HTTP

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 (an address that is unroutable on the Internet, but valid and unique for the PIX inside network) to the PIX configuration using this 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 Diagram

PIX Configuration Virtual HTTP Outbound

```
ip address outside 9.9.9.1 255.255.255.0
 ip address inside 171.68.118.115 255.255.255.0
 global (outside) 1 9.9.9.5−9.9.9.9 netmask 255.0.0.0
 timeout uauth 01:00:00
```

```
aaa-server TACACS+ protocol tacacs+
aaa-server AuthOutbound protocol tacacs+
aaa-server AuthOutbound (inside) host 171.68.118.101 cisco timeout 10
aaa authentication any outbound 0.0.0.0 0.0.0.0 0.0.0.0 0.0.0.0 AuthOutbound
virtual http 171.68.118.99
auth-prompt THIS_IS_PIX_5
```

Virtual Telnet

It is possible to configure the PIX to authenticate all inbound and outbound traffic, but it is not a good idea to do so. This is 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 unauthenticatable protocols show messages such as:

```
109001: Auth start for user '???' from 9.9.9.10/11094
   to 171.68.118.106/25
109009: Authorization denied from 171.68.118.106/49 to
   9.9.9.10/11094 (not authenticated
```

Since mail and some other services are not interactive enough to authenticate, one solution is to use the **except** command for authentication/authorization (authenticate all except for source/destination of the mail server/client).

If there is a real 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. After this authentication, the traffic for the unusual service can go to the real server.

In this example, we want TCP port 49 traffic to flow from outside host 9.9.9.10 to inside host 171.68.118.106. Since this traffic is not really authenticatable, we set up a virtual Telnet. For inbound virtual Telnet, there must be an associated static. Here, both 9.9.9.20 and 171.68.118.20 are virtual addresses.

Virtual Telnet Inbound Diagram
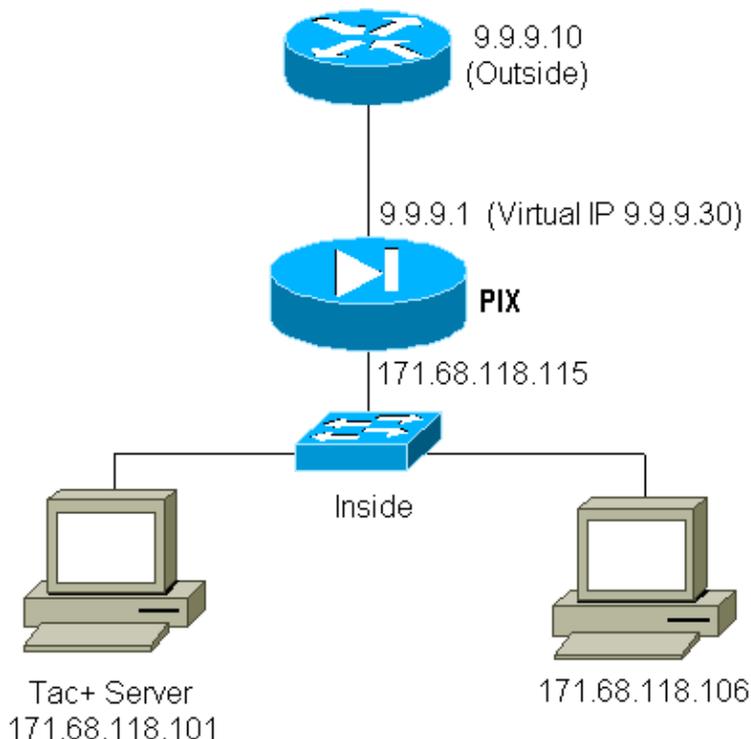
PIX Configuration Virtual Telnet Inbound

```
ip address outside 9.9.9.1 255.255.255.0
 ip address inside 171.68.118.115 255.255.255.0
 static (inside,outside) 9.9.9.20 171.68.118.20 netmask 255.255.255.255 0 0
 static (inside,outside) 9.9.9.30 171.68.118.106 netmask 255.255.255.255 0 0
 conduit permit tcp host 9.9.9.20 eq telnet any
 conduit permit tcp host 9.9.9.30 eq tacacs any
 aaa-server TACACS+ protocol tacacs+
 aaa-server AuthInbound protocol tacacs+
 aaa-server AuthInbound (inside) host 171.68.118.101 cisco timeout 5
 AAA authentication any inbound 0.0.0.0 0.0.0.0 0.0.0.0 0.0.0.0 AuthInbound
 virtual telnet 9.9.9.20
```

TACACS+ Server User Configuration Virtual Telnet Inbound

```
user = pinecone {
default service = permit
        login = cleartext "pinecone"
service = exec {
        timeout = 10
        idletime = 10
        }
}
```

PIX Debug Virtual Telnet Inbound

The user at 9.9.9.10 must first authenticate by Telnetting to the 9.9.9.20 address on the PIX:

```
pixfirewall# 109001: Auth start for user '???' from 9.9.9.10/11099
    to 171.68.118.20/23
 109011: Authen Session Start: user 'pinecone', Sid 13
 109005: Authentication succeeded for user 'pinecone'
    from 171.68.118.20/23 to 9.9.9.10/1470
```

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

```
pixfirewall# show uauth
                        Current     Most Seen
 Authenticated Users       1           1
 Authen In Progress        0           1
 user 'pinecone' at 9.9.9.10, authenticated
    absolute   timeout: 0:10:00
    inactivity timeout: 0:10:00
```

Here, the device at 9.9.9.10 wants to send TCP/49 traffic to the device at 171.68.118.106:

```
pixfirewall# 109001: Auth start for user 'pinecone' from 9.9.9.10/11104
    to 171.68.118.20/23
 109011: Authen Session Start: user 'pinecone', Sid 14
 109005: Authentication succeeded for user 'pinecone' from 171.68.118.20/23
    to 9.9.9.10/1470
 302001: Built TCP connection 23 for faddr 9.9.9.10/11104 gaddr 9.9.9.30/49
  laddr 171.68.118.106/49 (pinecone)
 302002: Teardown TCP connection 23 faddr 9.9.9.10/11104 gaddr 9.9.9.30/49
  laddr 171.68.118.106/49 duration 0:00:10 bytes 179 (pinecone)
```

Virtual Telnet Outbound

Since outbound traffic is allowed by default, no static is required for use of virtual Telnet outbound. In this

example, the inside user at 171.68.118.143 Telnets to virtual 9.9.9.30 and authenticates. The Telnet connection is immediately dropped. Once authenticated, TCP traffic is allowed from 171.68.118.143 to the server at 9.9.9.10:
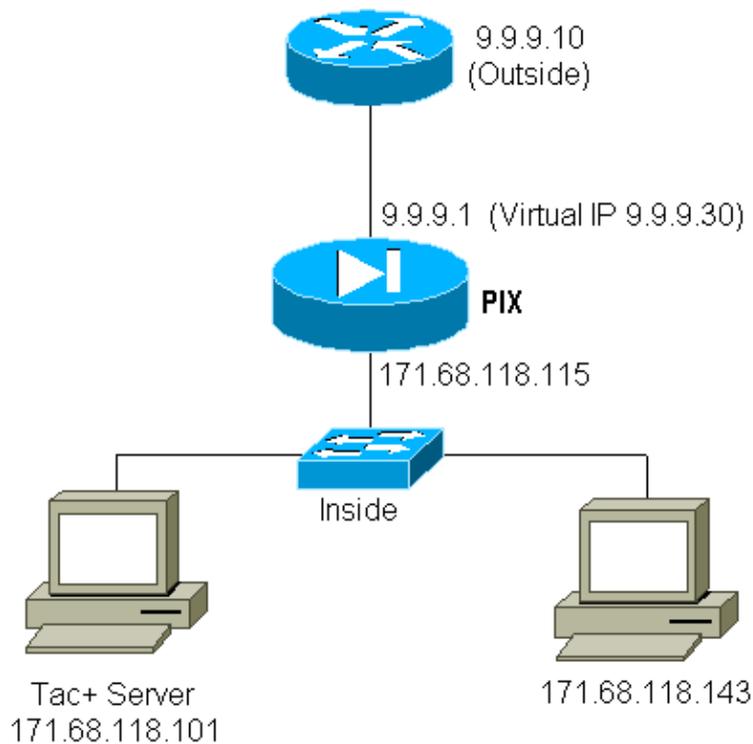
PIX Configuration Virtual Telnet Outbound

```
ip address outside 9.9.9.1 255.255.255.0
 ip address inside 171.68.118.115 255.255.255.0
 global (outside) 1 9.9.9.5-9.9.9.9 netmask 255.0.0.0
 timeout uauth 00:05:00
 aaa-server TACACS+ protocol tacacs+
 aaa-server AuthOutbound protocol tacacs+
 aaa-server AuthOutbound (inside) host 171.68.118.101 cisco timeout 10
 AAA authentication any outbound 0.0.0.0 0.0.0.0 0.0.0.0 0.0.0.0 AuthOutbound
 virtual telnet 9.9.9.30
```

PIX Debug Virtual Telnet Outbound

```
109001: Auth start for user '???' from 171.68.118.143/1536
     to 9.9.9.30/23
 109011: Authen Session Start: user 'timeout_143', Sid 25
 109005: Authentication succeeded for user 'timeout_143' from
    171.68.118.143/1536 to 9. 9.9.30/23
 302001: Built TCP connection 46 for faddr 9.9.9.10/80 gaddr
    9.9.9.30/1537 laddr 171.68 .118.143/1537 (timeout_143)
 304001: timeout_143@171.68.118.143 Accessed URL 9.9.9.10:/
 302001: Built TCP connection 47 for faddr 9.9.9.10/80 gaddr
    9.9.9.30/1538 laddr 171.68 .118.143/1538 (timeout_143)
 302002: Teardown TCP connection 46 faddr 9.9.9.10/80 gaddr
    9.9.9.30/1537 laddr 171.68. 118.143/1537 duration 0:00:03
    bytes 625 (timeout_143)
 304001: timeout_143@171.68.118.143 Accessed URL 9.9.9.10:/
 302002: Teardown TCP connection 47 faddr 9.9.9.10/80 gaddr
    9.9.9.30/1538 laddr 171.68. 118.143/1538 duration 0:00:01
    bytes 2281 (timeout_143)
 302009: 0 in use, 1 most used
```

Virtual Telnet Logout

When the user Telnets to the virtual Telnet IP, the **show uauth** command shows the uauth.

If the user wants to prevent traffic from going through after the session is finished (when there is time left in the uauth), the user needs to Telnet to the virtual Telnet IP again. This toggles the session off.

Port Authorization

You can require authorization on a range of ports. In this example, authentication was still required for all outbound, but only authorization was required for TCP ports 23–49.

PIX Configuration

```
AAA authentication any outbound 0.0.0.0 0.0.0.0 0.0.0.0 0.0.0.0 AuthOutbound
AAA authorization tcp/23-49 outbound 0.0.0.0 0.0.0.0
   0.0.0.0 0.0.0.0 AuthOutbound
```

When the Telnet was done from 171.68.118.143 to 9.9.9.10, authentication and authorization occurred because Telnet port 23 is in the 23–49 range.

When an HTTP session is done from 171.68.118.143 to 9.9.9.10, you still have to authenticate, but the PIX does not ask the TACACS+ server to authorize HTTP because 80 is not in the 23–49 range.

TACACS+ Freeware Server Configuration

```
user = telnetrange {
        login = cleartext "telnetrange"
        cmd = tcp/23-49 {
        permit 9.9.9.10
        }
        }
```

Note that the PIX sends "`cmd=tcp/23-49`" and "`cmd-arg=9.9.9.10`" to the TACACS+ server.

Debug on the PIX

```
109001: Auth start for user '???' from 171.68.118.143/1051
    to 9.9.9.10/23
 109011: Authen Session Start: user 'telnetrange', Sid 0
 109005: Authentication succeeded for user 'telnetrange'
    from 171.68.118.143/1051 to 9. 9.9.10/23
 109011: Authen Session Start: user 'telnetrange', Sid 0
 109007: Authorization permitted for user 'telnetrange'
    from 171.68.118.143/1051 to 9.9 .9.10/23
 302001: Built TCP connection 0 for faddr 9.9.9.10/23
    gaddr 9.9.9.5/1051 laddr 171.68.1 18.143/1051 (telnetrange)
 109001: Auth start for user '???' from 171.68.118.143/1105
    to 9.9.9.10/80
 109001: Auth start for user '???' from 171.68.118.143/1110
    to 9.9.9.10/80
 109011: Authen Session Start: user 'telnetrange', Sid 1
 109005: Authentication succeeded for user 'telnetrange'
    from 171.68.118.143/1110 to 9. 9.9.10/80
 302001: Built TCP connection 1 for faddr 9.9.9.10/80 gaddr 9.9.9.5/1110
  laddr 171.68.1 18.143/1110 (telnetrange)
 302001: Built TCP connection 2 for faddr 9.9.9.10/80 gaddr 9.9.9.5/1111
  laddr 171.68.1 18.143/1111 (telnetrange)
```

```
302002: Teardown TCP connection 1 faddr 9.9.9.10/80 gaddr 9.9.9.5/1110
 laddr 171.68.11 8.143/1110 duration 0:00:08 bytes 338 (telnetrange)
304001: timeout_143@171.68.118.143 Accessed URL 9.9.9.10:/
302002: Teardown TCP connection 2 faddr 9.9.9.10/80 gaddr 9.9.9.5/1111
 laddr 171.68.11 8.143/1111 duration 0:00:01 bytes 2329 (telnetrange)
```

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

PIX software version 5.0 changes the traffic accounting functionality. Accounting records can now be cut for traffic other than HTTP, FTP, and Telnet, once authentication is completed.
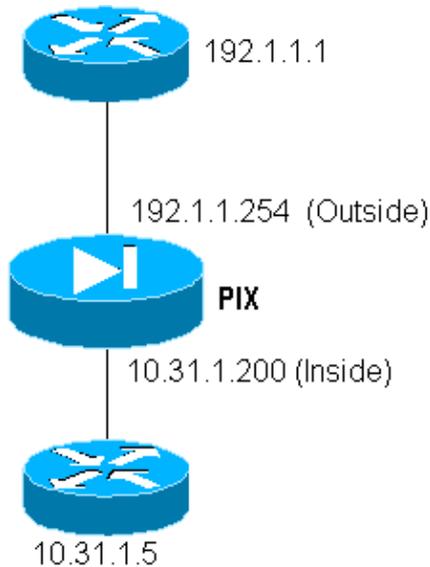
To TFTP−copy a file from the outside router (192.1.1.1) to the inside router (10.31.1.5), add virtual Telnet to open up a hole for the TFTP process:

```
virtual telnet 192.1.1.30
static (inside,outside) 192.1.1.30 10.31.1.5 netmask 255.255.255.255 0 0
conduit permit udp any any
AAA authentication udp/69 inbound 0.0.0.0 0.0.0.0 0.0.0.0 0.0.0.0 AuthInbound
AAA authorization udp/69 inbound 0.0.0.0 0.0.0.0 0.0.0.0 0.0.0.0 AuthInbound
AAA accounting udp/0 inbound 0.0.0.0 0.0.0.0 0.0.0.0 0.0.0.0 AuthInbound
```

Next, Telnet from the outside router at 192.1.1.1 to virtual IP 192.1.1.30 and authenticate to the virtual address which allows UDP to traverse the PIX. In this example, the **copy tftp flash** process was started from outside to inside:

```
302006: Teardown UDP connection for faddr 192.1.1.1/7680
     gaddr 192.1.1.30/69 laddr 10.31.1.5/69
```

For every **copy tftp flash** on the PIX (there were three during this IOS copy), an accounting record is cut and sent to the authentication server. Following is an example of a TACACS record on Cisco Secure Windows):

```
Date,Time,Username,Group-Name,Caller-Id,Acct-Flags,elapsed_time,
    service,bytes_in,bytes_out,paks_in,paks_out,
    task_id,addr,NAS-Portname,NAS-IP-Address,cmd
 04/28/2000,03:08:26,pixuser,Default Group,192.1.1.1,start,,,,,,,
    0x3c,,PIX,10.31.1.200,udp/69
```

Related Information

- **Documentation for PIX Firewall**
- **PIX Command Reference**
- **PIX Product Support Page**
- **Requests for Comments (RFCs)**
- **Technical Support – Cisco Systems**

Updated: Sep 26, 2008   Document ID: 13820