



Lab 9.1.7b Configure Access Through the PIX Security Appliance using CLI

Objective

In this lab exercise, the students will complete the following tasks:

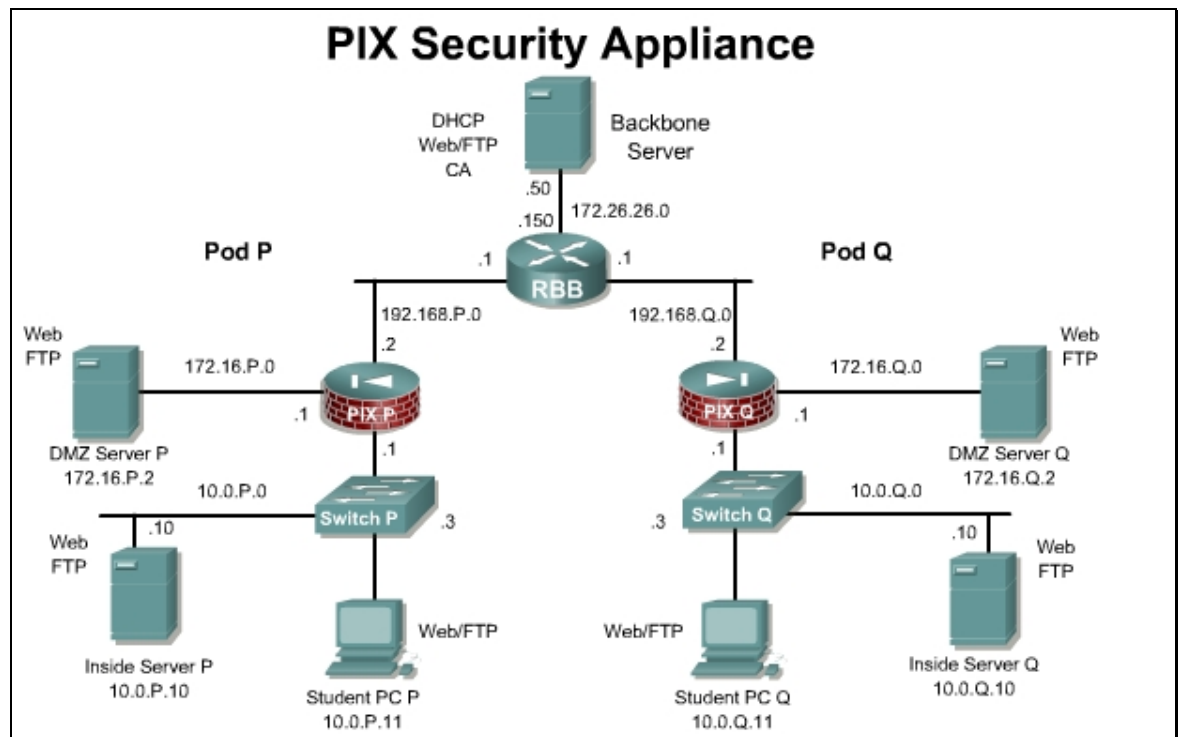
- Configure the PIX Security Appliance to allow inbound traffic to the inside host.
- Configure the PIX Security Appliance to allow inbound traffic to the bastion host.
- Test and verify correct PIX Security Appliance operation.

Scenario

In this exercise, the task is to configure the PIX Security Appliance to protect the internal campus network from outside intruders, while allowing web/ftp access to a DMZ server and web access to one host on the inside.

Topology

This figure illustrates the lab network environment.



Preparation

Begin with the standard lab topology and verify the starting configuration on the pod PIX Security Appliance. Access the PIX Security Appliance console port using the terminal emulator on the student PC. If desired, save the PIX Security Appliance configuration to a text file for later analysis.

Tools and resources

In order to complete the lab, the following is required:

- Standard PIX Security Appliance lab topology
- Console cable
- HyperTerminal

Additional materials

Further information about the objectives covered in this lab can be found at,

http://www.cisco.com/en/US/products/sw/secursw/ps2120/products_installation_and_configuration_guides_list.html

Command list

In this lab exercise, the following commands will be used. Refer to this list if assistance or help is needed during the lab exercise.

| Command | Description |
|--|--|
| <code>clear xlate</code> | Clears the contents of the translation slots. |
| <code>debug icmp trace</code> | Displays information about Internet Control Message Protocol (ICMP) traffic. |
| <code>global (mapped_interface) nat_id {mapped_ip_address [-mapped_ip_address] [netmask mapped_mask]} interface</code> | Create or delete entries from a pool of global addresses. |
| <code>show arp</code> | Change or view the arp table, and set the arp timeout value. |
| <code>show conn</code> | Display connection information. |
| <code>show xlate</code> | Display current translation and connection slot information. |
| <code>static (real_interface,mapped_interface) {mapped_ip_address interface} {real_ip_address [netmask mask]} {access-list access_list_name} [dns] [norandomseq [nailed]] [[tcp] [max_conns [emb_lim]] [udp udp_max_conns]</code> | Configure a persistent one-to-one address translation rule by mapping a local IP address to a global IP address. This is also known as Static port address translation (Static PAT). Configuration mode. |

Step 1 Verify the starting configuration

Load the startup configuration or configure the following via CLI.

- a. Configure the hostname and domain name.
- b. Configure name to address mappings for:
 - bastionhost at 172.16.P.2
 - insidehost at 10.0.P.11(Where P = pod number)
- c. Configure the inside, outside, and DMZ interface(s).
 - ii. Give each interface the appropriate IP address and name.
 - iii. Enable the Ethernet 0, Ethernet 1, and Ethernet 2 interfaces as 100-Mbps full duplex.
 - iv. Assign all hosts on the inside network to a Network Address Translation (NAT) pool.
 - v. Define a global pool of IP addresses for inside hosts to use on the outside interface. Use IP addresses 192.168.P.32–192.168.P.253.
 - vi. Set a default route for all internal hosts to exit the outside interface.
- d. Enable ASDM access for the inside host.
- e. Test connectivity from the inside to outside using HTTP or FTP.

Step 2 Configure the PIX Security Appliance to Allow Users on the Inside Interface to Access the Bastion Host

Configure the PIX Security Appliance to allow access to the DMZ from the inside network.

- a. Assign one pool of IP addresses for hosts on the public DMZ:

```
PixP(config)# global (dmz) 1 172.16.P.32-172.16.P.253 netmask 255.255.255.0
```

(where P = pod number)

- c. Clear the translation table so that the global IP address will be updated in the table:

```
PixP(config)# clear xlate
```

- d. Write the current configuration to Flash memory:

```
PixP(config)# write memory
```

- e. Test connectivity to the bastion host from the PIX.

```
PixP(config)# ping 172.16.P.2
```

(where P = pod number)

- f. Test web access to the pod bastion host from the pod PC by completing the following substeps:

- i. Open a web browser on the Student PC.
- ii. Use the web browser to access the pod bastion host by entering **http://172.16.P.2**.

(where P = pod number)

The home page of the bastion host should appear on the web browser.

- g. Use the **show arp**, **show conn**, and **show xlate** commands to observe the transaction:

```
PixP(config)# show arp
```

```
outside 192.168.P.1 00e0.1e41.8762
```

```
inside insidehost 00e0.b05a.d509
```

```
dmz bastionhost 00e0.1eb1.78df
```

```
PixP(config)# show xlate
```

```
1 in use, 2 most used
```

```
Global 172.16.P.33 Local insidehost
```

```
PixP(config)# show conn
```

```
2 in use, 2 most used
```

```
TCP out bastionhost:80 in insidehost:1076 idle 0:00:07 Bytes 461  
flags UIO
```

```
TCP out bastionhost:80 in insidehost:1075 idle 0:00:07 Bytes 1441  
flags UIO
```

(where P = pod number)

- h. Test the FTP access to the bastion host from the PC by completing the following substeps:
- i. Establish an FTP session to the bastion host by choosing **Start > Run > ftp 172.16.P.2**. If the following message appears, this indicates the bastion host has been reached:

```
"Connected to 172.16.P.2."
```

(where P = pod number)

- j. Log into the FTP session:

```
User (172.16.P.2(none)): anonymous
```

```
331 Anonymous access allowed, send identity (e-mail name) as  
password.
```

```
Password: cisco
```

(where P = pod number)

- k. Quit the FTP session after connecting and authenticating:

```
ftp> quit
```

Step 3 Configure the PIX Security Appliance to Allow Users on the Outside Interface to Access the Bastion Host

Configure a static translation so that traffic originating from the bastion host always has the same source address on the outside interface of the PIX Security Appliance. Then configure an ACL to allow users on the outside interface to access the bastion host.

- a. Create a static translation for the pod bastion host. Use the hostname configured in a previous lab step for the bastion host at 172.16.P.2.

```
PixP(config)# static (dmz,outside) 192.168.P.11 bastionhost
```

(where P = pod number)

- b. Configure an ACL to allow users on the outside interface to ping the bastion host.

```
PixP(config)# access-list OUTSIDE_ACCESS_IN permit icmp any any echo
```

```
PixP(config)# access-group OUTSIDE_ACCESS_IN in interface outside
```

- c. Ping a peer bastion host from the internal host as allowed by the ACL through the static:

```
C:\> ping 192.168.Q.11
```

(where Q = peer pod number)

- ```
PixP(config)# show xlate
```
- 2 in use, 2 most used
- Global 172.16.P.34 Local insidehost
- Global 192.168.P.11 Local bastionhost
- (where P = pod number)
- e. Test the web access to the bastion hosts of peer pod groups by completing the following substeps. The tests should fail.
- Open a web browser on the client PC.
  - Use the web browser to access the bastion host of the peer pod group by entering **http://192.168.Q.11**.  
(where Q = peer pod number)
  - Have a peer pod attempt to access their peer bastion host in the same way.
    - Why did the connection fail?

- ```
PixP(config)# access-list OUTSIDE_ACCESS_IN permit tcp any host
192.168.P.11 eq www

PixP(config)# access-list OUTSIDE_ACCESS_IN permit tcp any host
192.168.P.11 eq ftp
```

- Copyright © 2005, Cisco Systems, Inc.

- iii. Use the **show arp**, **show conn**, and **show xlate** commands to observe the transaction.

Step 4 Configure the PIX Security Appliance to Allow Users on the Outside Interface to Access the Inside Host

- a. Configure a static translation so that traffic originating from the student PC always has the same source address on the outside interface of the PIX Security Appliance. Then configure an ACL to allow users on the outside interface to access the student PC.
- b. Create a static translation from the outside PIX Security Appliance interface to the internal host, and create an ACL to allow web connections from the outside to the PC on the inside:

```
PixP(config)# static (inside,outside) 192.168.P.10 insidehost
PixP(config)# access-list OUTSIDE_ACCESS_IN permit tcp any host
192.168.P.10 eq www
```

(where P = the pod number)

- c. Turn on Internet Control Message Protocol (ICMP) monitoring at the PIX Security Appliance:

```
PixP(config)# debug icmp trace
debug icmp trace enabled at level 1
```

- d. Clear the translation table:

```
PixP(config)# clear xlate
```

- e. Ping the static outside address of the peer inside host to test the translation. Observe the source and destination of the packets at the console of the PIX Security Appliance:

```
C:\> ping 192.168.P.1
```

(where Q = peer pod number)

Note the example display for PixP:

```
ICMP echo request (len 72 id 5 seq 0) 10.0.Q.11 > 192.168.P.10
ICMP echo reply (len 72 id 5 seq 0) insidehost > 10.0.Q.11
ICMP echo request (len 72 id 5 seq 1) 10.0.Q.11 > 192.168.P.10
ICMP echo reply (len 72 id 5 seq 1) insidehost > 10.0.Q.11
ICMP echo request (len 72 id 5 seq 2) 10.0.Q.11 > 192.168.P.10
ICMP echo reply (len 72 id 5 seq 2) insidehost > 10.0.Q.11
ICMP echo request (len 72 id 5 seq 3) 10.0.Q.11 > 192.168.P.10
ICMP echo reply (len 72 id 5 seq 3) insidehost > 10.0.Q.11
ICMP echo request (len 72 id 5 seq 4) 10.0.Q.11 > 192.168.P.10
ICMP echo reply (len 72 id 5 seq 4) insidehost > 10.0.Q.11
```

- f. Observe the source, destination, and translated addresses on the PIX Security Appliance console.
- g. Test web access to a peer pod inside host as allowed by the static and ACL configured in this task by completing the following substeps:
 - i. Open a web browser on the Student PC.
 - ii. Use the web browser to access the inside host of the peer pod by entering **http://192.168.Q.10.**
(where Q = peer pod number)

- h. Turn off the ICMP debugging:

```
PixP(config)#no debug icmp trace
```

- i. Write the current configuration to the terminal and verify the previously entered commands are correct. After verifying the configuration, use the **write memory** to save the configuration to Flash memory. The configuration should appear similar to the following:

```
PixP(config)# write terminal
: Saved
:
PIX Version 7.0(1)
names
name 172.16.P.2 bastionhost
name 10.0.P.11 insidehost
!
interface Ethernet0
  speed 100
  nameif outside
  security-level 0
  ip address 192.168.P.2 255.255.255.0
!
interface Ethernet1
  speed 100
  nameif inside
  security-level 100
  ip address 10.0.P.1 255.255.255.0
!
interface Ethernet2
  speed 100
  nameif dmz
  security-level 50
  ip address 172.16.P.1 255.255.255.0
!
enable password 8Ry2YjIyt7RRXU24 encrypted
passwd 2KFQnbNIdI.2KYOU encrypted
hostname PixP
domain-name cisco.com
ftp mode passive
access-list OUTSIDE_ACCESS_IN extended permit icmp any any echo
access-list OUTSIDE_ACCESS_IN extended permit tcp any host
192.168.P.11 eq www
```

```

access-list OUTSIDE_ACCESS_IN extended permit tcp any host
192.168.P.11 eq ftp
access-list OUTSIDE_ACCESS_IN extended permit tcp any host
192.168.P.10 eq www
pager lines 24
mtu outside 1500
mtu inside 1500
mtu dmz 1500
no failover
monitor-interface outside
monitor-interface inside
monitor-interface dmz
asdm image flash:/asdm
no asdm history enable
arp timeout 14400
global (outside) 1 192.168.P.32-192.168.P.253 netmask 255.255.255.0
global (dmz) 1 172.16.P.32-172.16.P.253 netmask 255.255.255.0
nat (inside) 1 0.0.0.0 0.0.0.0
static (dmz,outside) 192.168.P.11 bastionhost netmask
255.255.255.255
static (inside,outside) 192.168.P.10 insidehost netmask
255.255.255.255
access-group OUTSIDE_ACCESS_IN in interface outside
route outside 0.0.0.0 0.0.0.0 192.168.P.1 1
timeout xlate 3:00:00
timeout conn 1:00:00 half-closed 0:10:00 udp 0:02:00 icmp 0:00:02
timeout sunrpc 0:10:00 h323 0:05:00 h225 1:00:00 mgcp 0:05:00
timeout mgcp-pat 0:05:00 sip 0:30:00 sip_media 0:02:00
timeout uauth 0:05:00 absolute
http server enable
http insidehost 255.255.255.255 inside
no snmp-server location
no snmp-server contact
snmp-server enable traps snmp
telnet timeout 5
ssh timeout 5
console timeout 0
dhcpd address 10.0.P.32-10.0.P.253 inside
dhcpd lease 3600
dhcpd ping_timeout 50
dhcpd domain cisco.com

```



```
dhcpcd enable inside
!
class-map inspection_default
  match default-inspection-traffic
!
!
policy-map global_policy
  class inspection_default
    inspect dns maximum-length 512
    inspect ftp
    inspect h323 h225
    inspect h323 ras
    inspect netbios
    inspect rsh
    inspect rtsp
    inspect skinny
    inspect esmtp
    inspect sqlnet
    inspect sunrpc
    inspect tftp
    inspect sip
    inspect xdmcp
  !
service-policy global_policy global
Cryptochecksum:599d3ee100d62cfb3db8fe1790a77fcb
: end
Pixl(config)#
```