

# PIX Firewall Emulator

## Introduction

The emulator is shown in Figure 1. A PIX firewall focuses on filtering network traffic in terms of firewall rules. Figure 2 shows an example with three ports. One port connects to the untrusted network (E0 - outside), one port connects to the trusted network (E1 - inside) and one port connects to the third network (E2 - inf2), which typically defines the DMZ. In this case the public access servers will be placed in the DMZ, so that external traffic from the outside network will be streamed off into the DMZ. Thus firewall rules can be applied for outside-to-DMZ, outside-to-inside, and inside-to-DMZ. The PIX device contains many security functions:

- Firewall rules. These are contained within ACLs (using the **access-list** and **access-group** commands), and block or permit traffic.
- Port blocking. These use the **fixup** command to change, enable or disable network services.
- Intrusion detection. These use the ip audit command to detect intrusions.
- Shunning. This, along with intrusion detection, allows a defined response to an intrusion.
- Failover. This allows other devices to detect that a PIX device has crashed, and that another device needs to take its place.

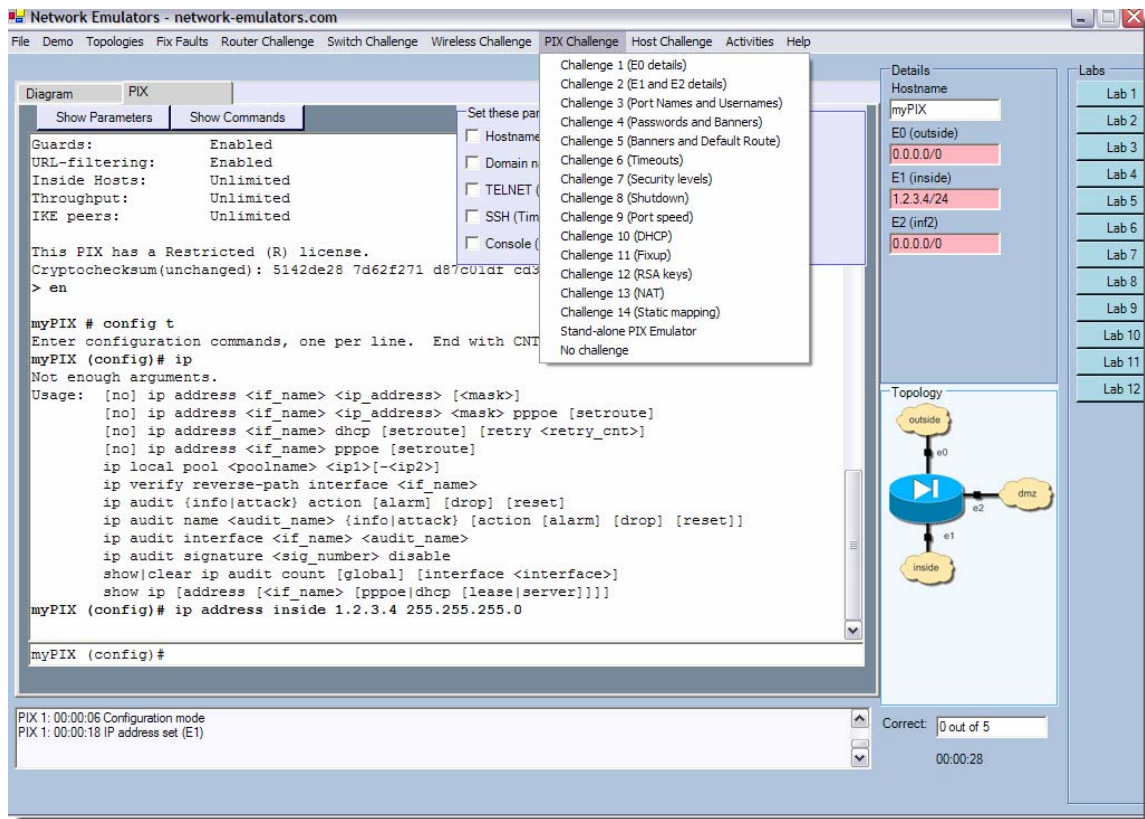


Figure 1: PIX Emulator

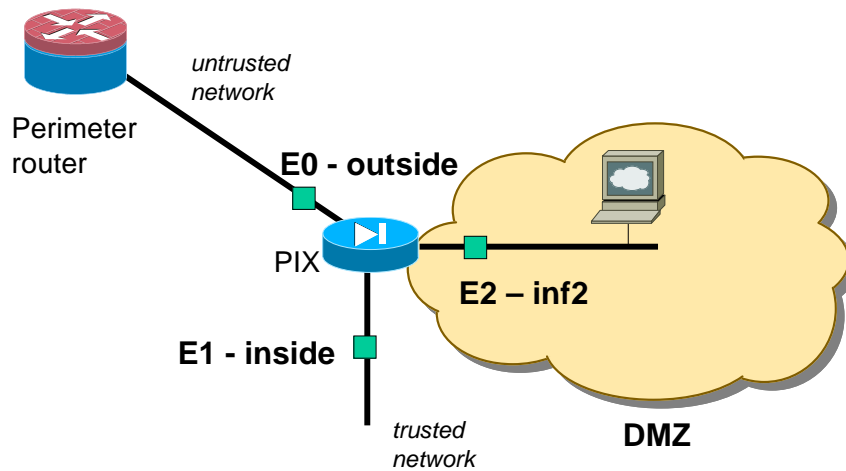


Figure 2 PIX firewall

## Tutorial 1 (Basic Configuration)

1. You should start in the user mode:

```
>
```

2. View the commands available in this mode with:

```
> ?
```

3. Go into the EXEC mode using the enable command.

```
> enable
```

**How does the prompt change?**

4. From the EXEC mode go into the Global Configuration Mode, and use the hostname command to change the hostname to MyPIX.

```
# ?
# config t
(config)# hostname ?
(config)# hostname MyPIX
(config)# password ?
(config)# password cisco
(config)# enable password ?
(config)# enable password cisco
```

**How does the prompt change?**

- Exit from the Global Configuration Mode using `exit`, and list the current running-config with `show running-config`.

```
(config) # exit
# show running-conf
```

**Outline some of the settings in the running-config:**

### Showing version and activation-key

- Use the **show version**, and **show activation-key** commands to display the details of the system:

```
# show version
Cisco PIX Firewall Version 6.3(1)
Cisco PIX Device Manager Version 3.0(1)

Compiled on Wed 19-Mar-03 11:49 by morlee

pixfirewall up 222 days 0 hours

Hardware:   PIX-515E, 32 MB RAM, CPU Pentium II 433 MHz
Flash E28F128J3 @ 0x300, 16MB
BIOS Flash AM29F400B @ 0xffffd8000, 32KB

0: ethernet0: address is 000d.6585.77cb, irq 10
1: ethernet1: address is 000d.6585.77cc, irq 11
2: ethernet2: address is 0002.b3cc.782a, irq 11
Licensed Features:
Failover:           Disabled
VPN-DES:            Enabled
VPN-3DES-AES:       Enabled
Maximum Interfaces: 3
Cut-through Proxy: Enabled
Guards:             Enabled
URL-filtering:      Enabled
Inside Hosts:       Unlimited
Throughput:         Unlimited
IKE peers:          Unlimited

This PIX has a Restricted (R) license.

Serial Number: 807290127 (0x301e450f)
Running Activation Key: 0x34a51644 0x8429686c 0xeb739343 0x2c42ff31
Configuration last modified by enable_15 at 04:25:55.894 UTC Sun Jan 22
2006

# show activation-key
```

```
Serial Number: 807290127 (0x301e450f)

Running Activation Key: 0x34a51644 0x8429686c 0xeb739343 0x2c42ff31
Licensed Features:
Failover:           Disabled
VPN-DES:            Enabled
VPN-3DES-AES:       Enabled
Maximum Interfaces: 3
Cut-through Proxy: Enabled
Guards:             Enabled
URL-filtering:      Enabled
Inside Hosts:       Unlimited
Throughput:         Unlimited
IKE peers:          Unlimited

This PIX has a Restricted (R) license.

The flash activation key is the SAME as the running key.
```

- To set the activation key:

```
# config t
(config) # activation-key ffff1111 22224444 12345678 00ff00ff
```

### Using the show command

- Complete the following command:

```
# ?
# show ?
# show nameif
# show version
# show interface
# show processes
# show conn
# show fixup
# show aaa
# show aaa-server
# show blocks
# show domain-name
# show history
# show traffic
# show memory
# show clock
# show terminal
# show timeout
# show ua
```

Using the information from above what are the following:

**How much memory does it have?**

**What is version of the PIX firewall software?**

**What is the version of the BIOS?**

**Which ports does the PIX device have?**

9. The IP addresses that are set can be shown at any time with:

```
# show ip
```

10. The show route commands shows the routes that have been setup. It will add a static route for every port that has been created. In this case there is only one port which has a default IP address (the inside port which has an IP address of 10.0.0.1 and a subnet mask of 255.255.255.0):

```
# show route
```

11. It should be seen that this gives:

```
inside 10.0.0.0 255.255.255.0 10.0.0.1 CONNECT static
```

**Programming the ports**

12. Program the three ports of the PIX with:

```
# nameif
# config t
(config)# ?
(config)# ip ?
(config)# ip address inside 192.168.1.1 255.255.255.0
(config)# ip address outside 10.1.1.1 255.255.0.0
(config)# ip address inf2 192.168.2.1 255.255.0.0
(config)# exit
# show ip
# show running
```

**Ping the newly defined ports. Are they responding?**

13. Program the three ports of the PIX with:

```
# nameif
# config t
(config)# ?
(config)# interface ?
(config)# interface e0 auto
(config)# interface e1 auto
(config)# interface e2 auto
(config)# exit
# show running
```

**Ping the newly defined ports. Are they responding?**

## Setting the domain name and time-outs

14. Set the domain-name with:

```
# config t
(config)# domain-name ?
(config)# domain-name fred.com
(config)# exit
# show domain-name
# show running
```

15. To set the time-outs:

```
# config t
(config)# telnet ?
(config)# telnet timeout 10
(config)# ssh ?
(config)# ssh timeout 10
(config)# console ?
(config)# console timeout 5
(config)# exit
# show running
```

16. To enable a WWW server:

```
# config t
(config)# http ?
(config)# http server enable
(config)# exit
# show running
```

17. To disable the WWW server:

```
# show http
# config t
(config)# no http server enable
(config)# exit
# show running
```

18. To enable a user:

```
# config t
(config)# username ?
(config)# username fred password fred
(config)# exit
# show running
```

19. To enable banners:

```
# config t
(config)# banner ?
(config)# banner motd # hello #
(config)# banner exec # welcome to exec #
(config)# banner login # welcome to PIX #
```

```
# show running
```

20. To disable banners:

```
# config t
(config)# no banner motd
(config)# no banner exec
(config)# no banner login
# show running
```

21. To change the IF name of a port:

```
# nameif
# config t
(config)# nameif ?
(config)# nameif e2 dmz security40
(config)# exit
# nameif
# show running
```

### Setting a static route

22. A static route is setup for each of the IP addresses that have been setup. For example:

```
# config t
(config)# ip address inside 192.168.1.1 255.255.255.0
(config)# ip address outside 10.1.1.1 255.255.0.0
(config)# ip address inf2 192.168.2.1 255.255.0.0
(config)# exit
(config)# show route
```

gives:

```
inside 192.168.1.0 255.255.255.0 192.168.1.1 1 CONNECT static
outside 10.1.0.0 255.255.0.0 10.1.1.1 1 CONNECT static
inf2 192.168.0.0 255.255.0.0 192.168.2.1 1 CONNECT static
```

23. To add a route:

```
# config t
(config)# route ?
(config)# route inside 10.1.1.0 255.255.255.0 10.0.0.3
# show running
```

The additional line in the running configuration is:

```
route inside 192.1.1.0 255.255.255.0 192.1.1.3 1
```

24. To show all the routes:

```
# show route
```

```
inside 192.1.1.0 255.255.255.0 192.1.1.3 1 OTHER static
inside 192.168.1.0 255.255.255.0 192.168.1.1 1 CONNECT static
```

```
outside 10.1.0.0 255.255.0.0 10.1.1.1 1 CONNECT static
inf2 192.168.0.0 255.255.0.0 192.168.2.1 1 CONNECT static
```

25. To get rid of a route:

```
# config t
(config)# no route inside 10.1.1.0 255.255.255.0 10.0.0.3
# show running
```

## RIP routing

26. For dynamic routing, the RIP command can be used:

```
# config t
(config)# rip
(config)# rip ?
(config)# rip outside passive version 2 authentication md5 HKEY abc
(config)# exit
# show running
```

The PIX device accepts RIP version 2, which supports MD5 authentication. In this case the key is set to “abc”.

## Network Time

27. To set the IP address of the NTP server:

```
# config t
(config)# ntp ?
(config)# ntp server 10.0.0.30
# show running
```

28. To get rid of access to the NTP server:

```
# config t
(config)# ntp ?
(config)# no ntp server 10.0.0.30
# show running
```

## Fixup

29. To configure the protocols that are enabled or disabled on the PIX firewall:

```
# show fixup
# config t
(config)# fixup ?
```

30. To disable the FTP protocol on the PIX device, then:

```
(config)# no fixup protocol ftp 23
(config)# exit
# show fixup
```

31. To enable it:

```
(config)# fixup protocol ftp 23
(config)# exit
# show fixup
```

32. To change the port that the PIX device listens for HTTP traffic:

```
(config)# fixup protocol http 8080
(config)# exit
# show fixup
```

### CPU Usage

33. To show CPU usage:

```
# show cpu use
```

### DHCP

34. To use the DHCP daemon:

```
# config t
(config)# dhcpd ?
```

### Debug

35. To use the debug options:

```
# debug ?
```

### ARP

36. To use the arp options:

```
# arp ?
# show arp
```

### Showing details

37. To show the details of the EEPROM:

```
# show eeprom
```

38. To show details of aaa-server:

```
# show aaa-server
```

### NAT

39. To show the details of the NAT and global commands:

```
# config t
(config)# nat ?
(config)# global ?
```

40. To setup NAT on the inside interface to use the network addresses from 10.0.0.1 to 10.0.0.254:

```
(config)# nat (inside) 1 10.0.0.0 255.255.255.0
(config)# exit
# show running
```

41. Next we could setup NAT in the DMZ so that it uses addresses from 172.16.0.1 to 172.16.0.254:

```
(config)# nameif e2 dmz security50
(config)# nat (dmz) 1 172.16.0.0 255.255.255.0
(config)# exit
# show nat
# show running
```

42. Finally we can assign the addresses on the outside and within the DMZ to be globally available addresses:

```
# config t
(config)# global (outside) 1 192.168.0.20-192.168.0.254 netmask
                255.255.255.0
(config)# global (dmz) 1 172.16.0.0-172.16.0.254 netmask 255.255.255.0
(config)# exit
# show global
# show running
```

The **global** command assigns a public address to internal hosts which are available through the firewall.

## ICMP

43. To show the details of the ICMP command, and to deny a ping response from the PIX from devices outside our network:

```
# config t
(config)# icmp ?
(config)# icmp deny any echo outside
```

## DHCPD

44. To show the details of the DHCP command:

```
# config t
(config)# dhcpd ?
(config)# dhcpd address 192.168.0.20-192.168.0.40 inside
(config)# exit
# show dhcpd
# show running
```

which operates a DHCP daemon on the inside network. To add a DNS link:

45. To show the details of the DHCP command:

```
# config t
(config)# dhcpd dns 192.168.0.100
# show running
```

## CPU Usage

46. To copy an image from a TFTP server into the Flash memory:

```
# copy tftp flash
```

## Example configuration 1: ACLs

The following is a configuration which blocks WWW access from inside the network, and permits access to a WWW server in the DMZ (Figure 2).

```
> enable
# config t
(config)# nameif ?
(config)# nameif e2 dmz security50
(config)# ip address inside 10.0.0.1 255.255.255.0
(config)# ip address outside 192.168.0.1 255.255.255.0
(config)# ip address dmz 172.16.0.1 255.255.255.0
(config)# interface e0 auto
(config)# interface e1 auto
(config)# interface e2 auto
(config)# access-list ?
(config)# access-list acl_out1 permit tcp 10.0.0.0 255.0.0.0 host
      172.16.0.2 eq www
(config)# access-list acl_out1 deny tcp any any eq www
(config)# access-list acl_out1 permit ip any any
(config)# access-group ?
(config)# access-group acl_out1 in interface inside
(config)# exit
# show running
```

To allow ICMP: access-list acl\_out1 permit icmp any any

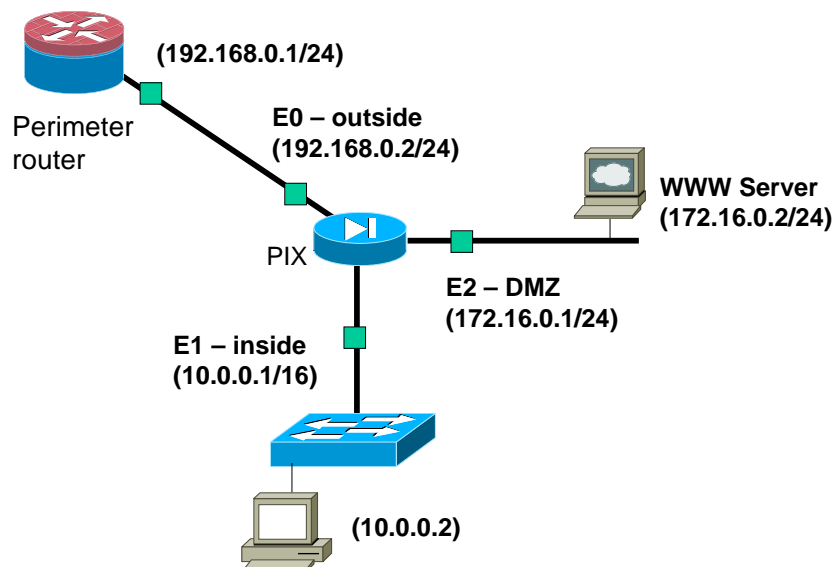


Figure 2 PIX firewall

## Example configuration 2: NAT

The following is a configuration which allows NAT to be setup on the inside and DMZ networks, and assigns global addresses to the DMZ and to the outside network.

```

> enable
# config t
(config)# nameif ?
(config)# nameif e2 dmz security50

(config)# ip address inside 10.0.0.1 255.255.255.0
(config)# ip address outside 192.168.0.1 255.255.255.0
(config)# ip address dmz 172.16.0.1 255.255.255.0

(config)# nat (inside) 1 10.0.0.0 255.255.255.0
(config)# nat (dmz) 1 172.16.0.0 255.255.255.0
(config)# global (outside) 1 192.168.0.20-192.168.0.254 netmask
                255.255.255.0
(config)# global (dmz) 1 172.16.0.2-172.16.0.254 netmask 255.255.255.0

(config)# interface e0 auto
(config)# interface e1 auto
(config)# interface e2 auto
(config)# exit
# show running

```

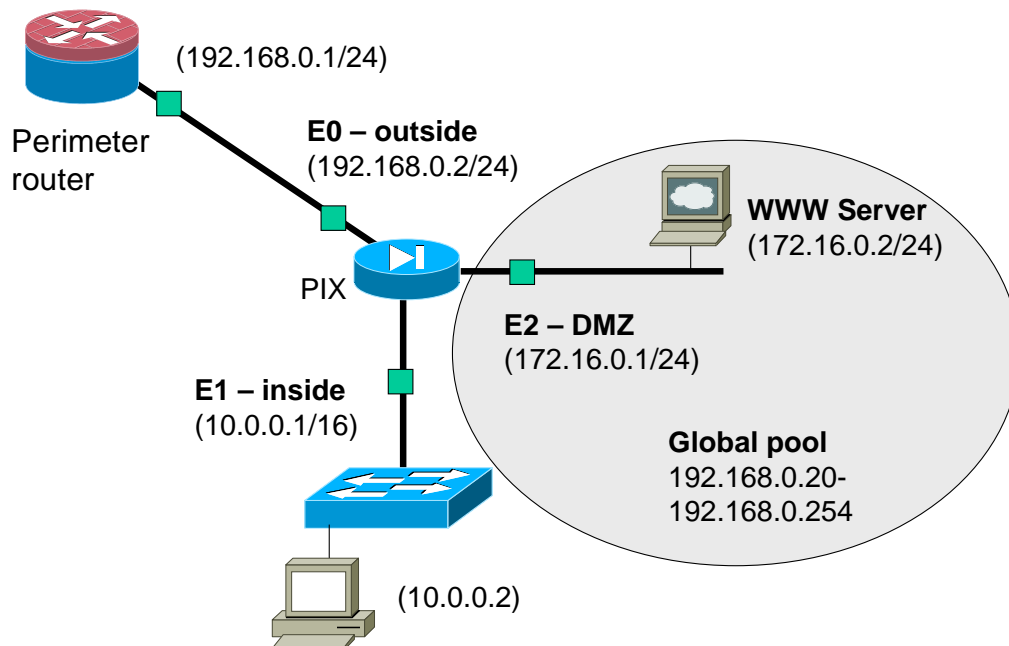


Figure 3 PIX firewall

## Appendix

The following are some sample labs which can be implemented on a real device.

### Simple setup

The following is basic setup of a PIX device with routers.

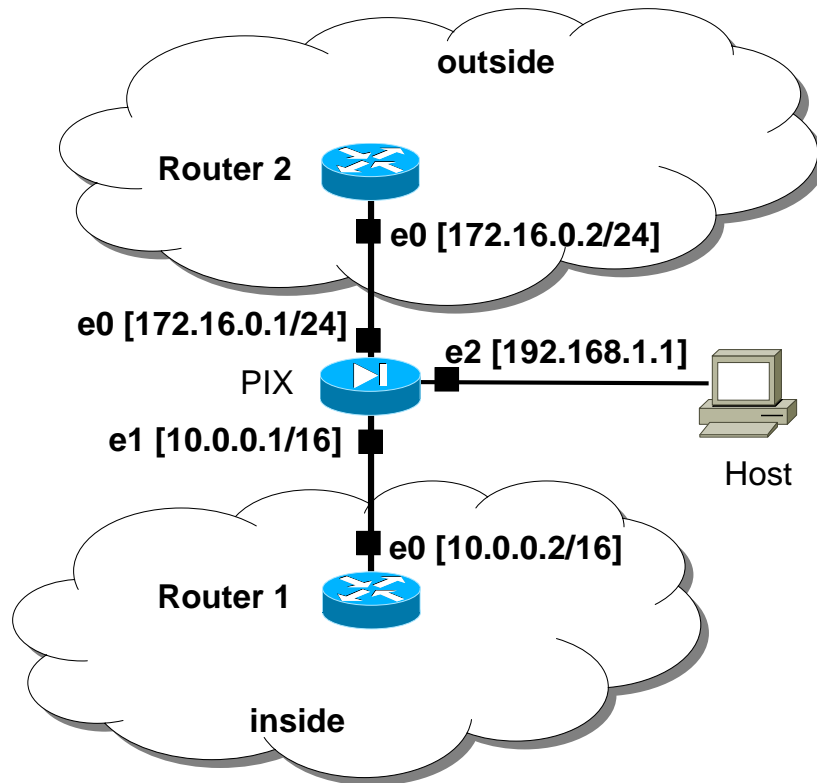


Figure 1: PIX pod example

The outline configuration of Router 2 is:

```
enable
config t
hostname outsider
int e0
ip address 172.16.0.2 255.255.255.0
no shut
exit
router rip
network 172.16.0.0
exit
ip default-gateway 172.16.0.1
line vty 0 4
password fred
login
exit
```

The outline configuration of Router 1 is (146.176.165.230:2015):

```
enable
config t
hostname insider
int e0
ip address 10.0.0.1 255.255.0.0
no shut
exit
router rip
network 10.0.0.0
exit
ip default-gateway 10.0.0.1
ip route 0.0.0.0 0.0.0.0 10.0.0.2
line vty 0 4
password fred
login
exit
```

and the PIX is (146.176.165.230:2014):

```
enable
config t
hostname myPIX
interface ethernet0 auto
interface ethernet1 auto
interface ethernet2 auto
nameif ethernet0 outside security0
nameif ethernet1 inside security100
nameif ethernet2 dmz security50
hostname pixfirewall
ip address outside 172.16.0.1 255.255.255.0
ip address inside 10.0.0.2 255.255.0.0
ip address dmz 192.168.1.1 255.255.255.0
global (outside) 1 192.168.2.20-192.168.2.254
nat (inside) 1 10.0.0.0 255.255.0.0 0 0
```

2. Next verify the NAT translation by using the following command on the PIX device:

```
pixfirewall# show xlate
1 in use, 1 most used
Global 192.168.2.20 Local 10.0.0.1
```

2. Now go to Router 1 (the inside router), and telnet from there to Router 2:

```
insiderR#telnet 172.16.0.2
Trying 172.16.0.2 ... Open
```

User Access Verification

```
Password:
outsider>
```

3. Next go to Router 2 (the outside router), and try and telnet into Router 1:

```
outsiderR#telnet 10.0.0.2
Trying 10.0.0.2 ...
```

Which shows that the traffic from inside to outside is allowed, but outside to inside is barred.

4. Now enable the WWW server on Router 2:

```
outsideR#config t
Enter configuration commands, one per line.  End with CNTL/Z.
outsideR(config)#ip http server
outsideR(config)#exit
```

5. Next, to prove that traffic from inside the network can access the outside network, go to Router 1 (inside), and access the WWW server on Router 2:

```
insideR#telnet 172.16.0.2 www
Trying 172.16.0.2, 80 ... Open
get index.html
content-type: http/html

HTTP/1.0 400 Bad Request
Date: Sun, 07 Mar 1993 13:58:59 UTC
Content-type: text/html
Expires: Thu, 16 Feb 1989 00:00:00 GMT

<H1>400 Bad Request</H1>
```

6. Now, and this is not advisable from a security point-of-view, we shall allow everything from outside to access the inside network:

```
pixfirewall# config t
pixfirewall(config)# access-list a2 permit ip any any
pixfirewall(config)# access-group a2 in interface outside
pixfirewall(config)# exit
```

7. Now go back to Router 2 (inside) and try and telnet, and now it should be possible to telnet into Router 1:

```
outsideR#telnet 10.0.0.1
Trying 10.0.0.1 ...
outsideR#telnet 192.168.2.20
Trying 192.168.2.20 ... Open
```

User Access Verification

Password:

8. Explain why Router 2 is accessible using 192.168.2.20?

Finally erase the configuration on the PIX:

```
pixfirewall# write erase
```

```
Erase PIX configuration in flash memory? [confirm]
pixfirewall# reload
Proceed with reload? [confirm]
```

## Verifying PIX (Blocking TELNET)

This lab verifies the blocking of TELNET.

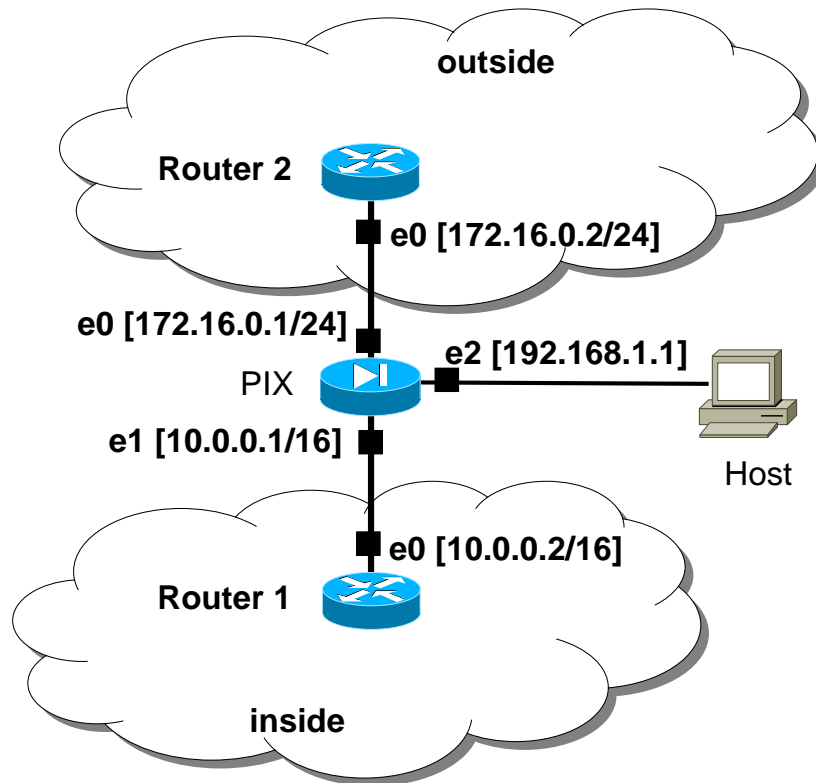


Figure 1: PIX pod example

1. Use the configuration from the previous lab (Page 12).
2. First make sure that you cannot access telnet on Router 1 from Router2:

```
outsider#telnet 192.168.2.20
Trying 192.168.2.20 ...
```

3. Next apply a rule which is applied on the outside port of the PIX device, so that it allows TELNET access to the port 192.168.2.20 (which is the 10.0.0.2 on the inside of the PIX device):

```
pixfirewall(config)# access-list myacl2 permit tcp any host 192.168.2.20
eq telnet
pixfirewall(config)# access-group myacl2 in interface outside
```

4. Now try and TELNET into the 192.168.2.20 port from Router 2, and it should be successful, such as:

```
outsider#telnet 192.168.2.20
Trying 192.168.2.20 ... Open
```

User Access Verification

Password:

5. Now, we will try and block TELNET access from the inside network to outside for every now in the inside network. First make sure you can telnet from Router 1 to Router 2:

```
insideR#telnet 172.16.0.2
Trying 172.16.0.2 ... Open
```

User Access Verification

Password:

6. Now, apply the rule which will block TELNET access to the external network:

```
pixfirewall(config)# access-list myacl3 deny tcp 10.0.0.0 255.255.255.0
                    host 10.0.0.0 255.255.255.0 host 172.16.0.2 eq telnet
pixfirewall(config)# access-group myacl3 in interface inside
```

7. Now, go to Router 1 (the inside router), and try and TELNET into Router 2 (outside), and now it should be blocked, such as:

```
insideR#telnet 172.16.0.2
Trying 172.16.0.2 ...
% Connection refused by remote host
```

8. Now, go to the PIX device, and get rid of the ACL which blocks TELNET, with

```
pixfirewall# config t
pixfirewall(config)# no access-group myacl3 in interface inside
```

8. Now, go back to Router 1 (the inside router) and verify that TELNET now works again, such as:

```
insideR#telnet 172.16.0.2
Trying 172.16.0.2 ... Open
```

User Access Verification

Password:

## Copyright information

© William Buchanan 2006

All rights reserved. No reproduction, copy or transmission of this publication may be made without written permission.

No paragraph of this publication may be reproduced, copied or transmitted save with written permission or in accordance with the provision of the Copyright, Designs and Patents Act 1988, or under the terms of any licence permitting limited copying issued by the Copyright Licensing Agency, 90 Tottenham Court Road, London W1T4LP.

Any person who does any unauthorised act in relation to this publication may be liable to criminal prosecution and civil claims for damages.

The author(s) have asserted their rights to be identified as the author(s) of this work in accordance with the Copyright, Designs and Patents Act 1988.

**Network-emulators.com -  
The best emulators on the Internet!**