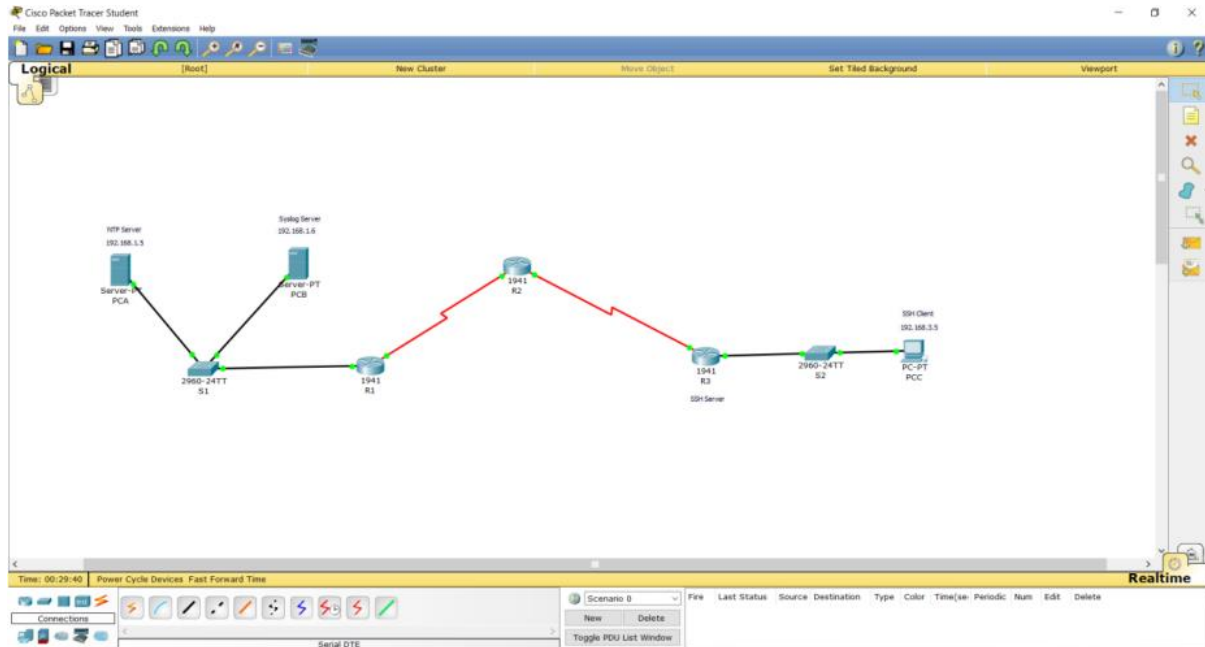


## Practical 1

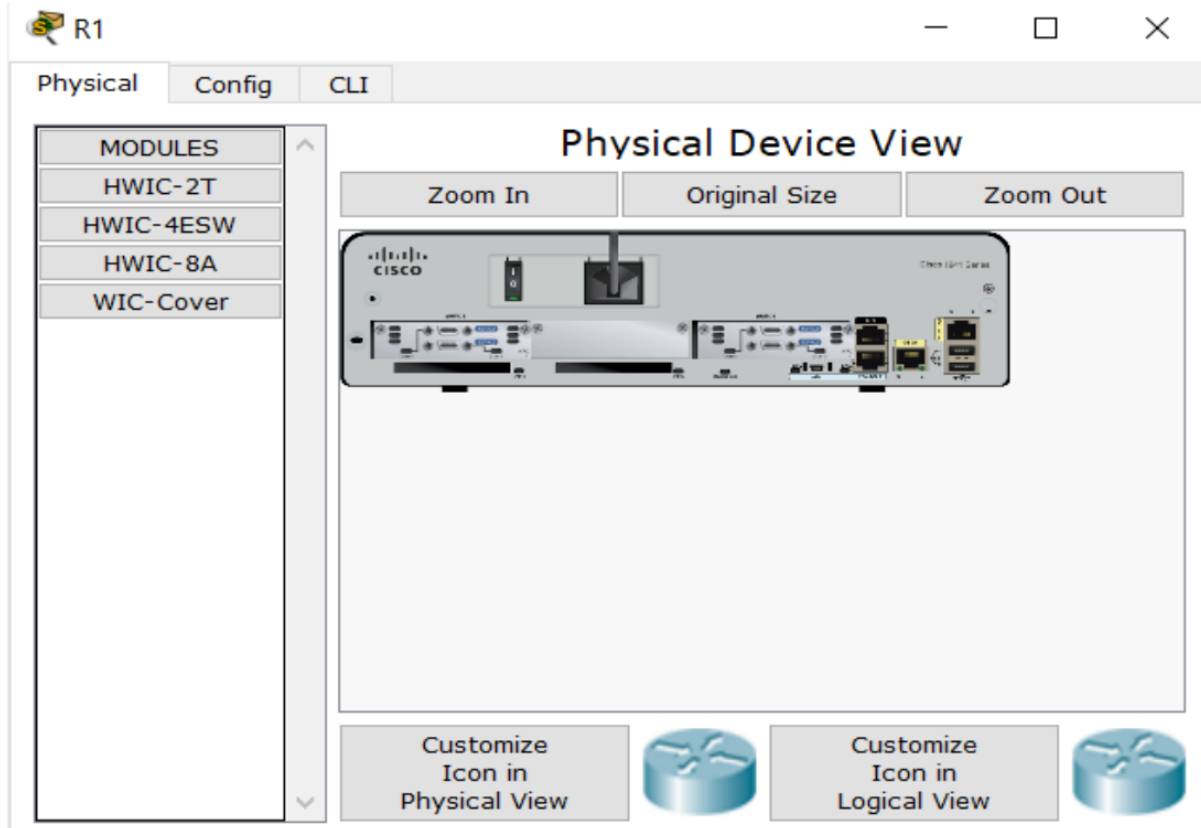
**Aim:** Configure Cisco Routers for Syslog, NTP, and SSH Operations

**Topology:**



Go to Each Router

Physical Section > Turn off router and move HWIC-2T to right and turn on the router.



Address Table:

Device	Interface	IP Address	Subnet Mask	Default Gateway	Switch Port
R1	G0/1	192.168.1.1	255.255.255.0	NIL	S1 F0/5
	S0/0/0	10.1.1.1	255.255.255.252	NIL	NIL
R2	S0/0/0	10.1.1.2	255.255.255.252	NIL	NIL
	S0/0/1	10.2.2.2	255.255.255.252	NIL	NIL
R3	G0/1	192.168.3.1	255.255.255.0	NIL	S3 F0/5
	S0/0/1	10.2.2.1	255.255.255.252	NIL	NIL
PCA	NIL	192.168.1.5	255.255.255.0	192.168.1.1	S1 F0/6
PCB	NIL	192.168.1.6	255.255.255.0	192.168.1.1	S2 F0/18
PCC	NIL	192.168.3.5	255.255.255.0	192.168.3.1	S3 F0/18

#### A. With OSPF MD5 Authentication

1. From PCA, PCB, PCC ping other Ip addresses of PCs in Command Prompt
2. Next we need to go to R1, R2, R3 CLI and type the respective commands as mentioned:

R1>en

R1#conf t R1(config)#router

ospf 1

R1(config-router)#area 0 authentication message-digest

R1(config-router)#exit

R1(config)#interface S0/0/0

R1(config-if)#ip ospf message-digest-key 1 md5 MD5pa55

R2>en

R2#conf t

R2(config)#router ospf 1

R2(config-router)#area 0 authentication message-digest

R2(config-router)#interface S0/0/0

R2(config-if)#ip ospf message-digest-key 1 md5 MD5pa55

R2(config-if)#interface S0/0/1

R2(config-if)#ip ospf message-digest-key 1 md5 MD5pa55

R3>en

R3#conf t

R3(config)#router ospf 1

R3(config-router)#area 0 authentication message-digest

R3(config-router)#interface S0/0/1

R3(config-if)#ip ospf message-digest-key 1 md5 MD5pa55

1. Now you need to verify MD5 authentication by typing this command in R1, R2, R3 CLI:

show ip ospf interface

To verify end to end connectivity again ping PCA, PCB, PCC address from each of their

```

Packet Tracer SERVER Command Line 1.0
SERVER> ping 192.168.3.5

Pinging 192.168.3.5 with 32 bytes of data:

Reply from 192.168.3.5: bytes=32 time=13ms TTL=125
Reply from 192.168.3.5: bytes=32 time=10ms TTL=125
Reply from 192.168.3.5: bytes=32 time=4ms TTL=125
Reply from 192.168.3.5: bytes=32 time=2ms TTL=125

Ping statistics for 192.168.3.5:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
    Approximate round trip times in milli-seconds:
        Minimum = 2ms, Maximum = 13ms, Average = 7ms

SERVER>

```

## B. Using NTP

1. Enable NTP authentication on PC-A. Server > NTP Key: 1 Password: NTPpa55

The screenshot shows the configuration interface for NTP on PC-A. The 'Config' tab is selected, and the 'Services' section is expanded. The 'NTP' service is checked 'On'. The 'Authentication' section shows 'Enable' selected, with 'Key' set to 0 and 'Password' set to NTPpa55. A calendar for February 2023 is displayed below the authentication settings, showing the current date as the 22nd.

2. Now type following code in CLI of R1, R2, R3:

```

R1(config)#ntp server 192.168.1.5 R1(config)#ntp
R1(config)#ntp authenticate
R1(config)#ntp trusted-key 1
R1(config)#ntp authentication-key 1 md5 NTPpa55
R1(config)#service timestamps log datetime msec
R2(config)#ntp server 192.168.1.5
R2(config)#ntp update-calendar
R2(config)#ntp authenticate
R2(config)#ntp trusted-key 1

```

```
R2(config)#ntp authentication-key 1 md5 NTPpa55
```

```
R2(config)#service timestamps log datetime msec
```

```
R3(config)#ntp server 192.168.1.5
```

```
R3(config)#ntp update-calendar
```

```
R3(config)#ntp authenticate
```

```
R3(config)#ntp trusted-key 1
```

```
R3(config)#ntp authentication-key 1 md5 NTPpa55
```

```
R3(config)#service timestamps log datetime msec
```

### C. To Log Messages to the Syslog Server

1. Type this command for router to identify the remote host:

```
R1(config)# logging host 192.168.1.6
```

```
R2(config)# logging host 192.168.1.6
```

```
R3(config)# logging host 192.168.1.6
```

Use the command to verify logging has been enabled: show logging

```
CLI initiated
Router#show logging
Syslog logging: enabled (0 messages dropped, 0 messages rate-limited,
                  0 flushes, 0 overruns, xml disabled, filtering disabled)

No Active Message Discriminator.

No Inactive Message Discriminator.

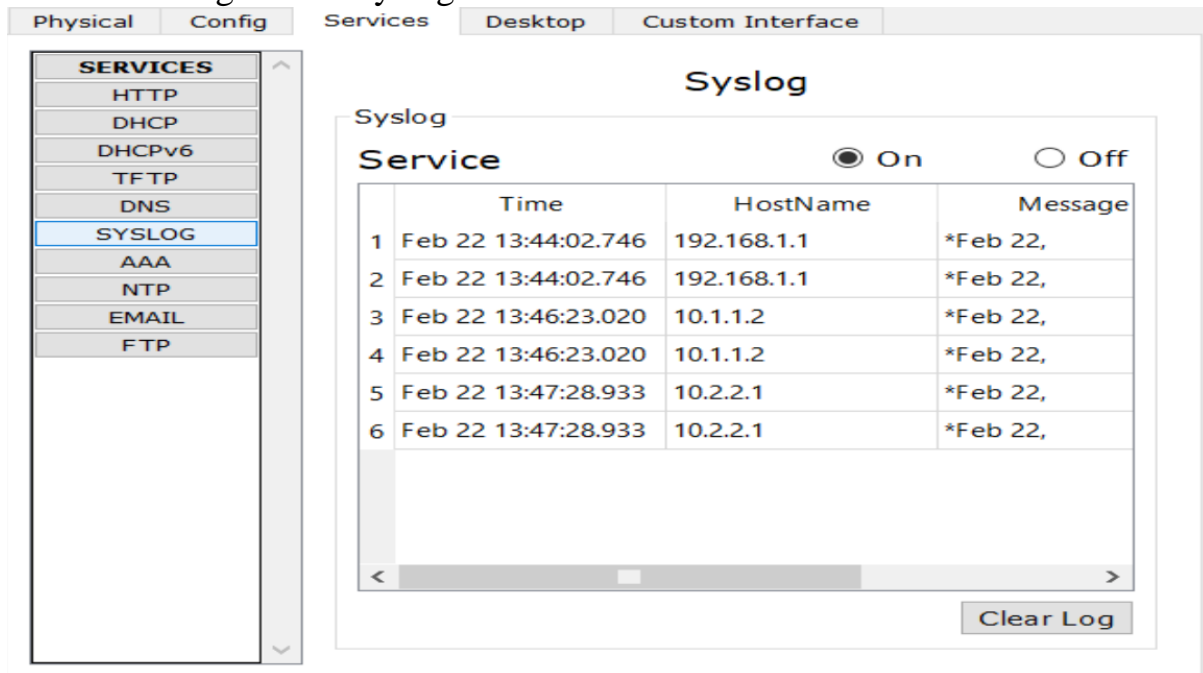
Console logging: level debugging, 12 messages logged, xml disabled,
                  filtering disabled
Monitor logging: level debugging, 0 messages logged, xml disabled,
                  filtering disabled
Buffer logging:   disabled, xml disabled,
                  filtering disabled

Logging Exception size (4096 bytes)
Count and timestamp logging messages: disabled
Persistent logging: disabled

No active filter modules.

--More--
```

### 1. Examine logs of the Syslog Server.



### D. To Support SSH Connections

1. Configure a domain name of ccnasecurity.com on R3 and users for login to the ssh server.

```
R3(config)#ip domain-name ccnasecurity.com
```

```
R3(config)#username SSHadmin privilege 15 secret ciscosshpa55
```

2. Configure the incoming vty lines on R3 and Erase existing key pairs on R3.

```
R3(config)#line
```

```
vty 0 4
```

```
R3(config-line)#login local R3(config-line)#transport input ssh
```

```
R3(config-line)#crypto key zeroize rsa
```

3. Generate the RSA encryption key pair for R3 and Verify the SSH configuration. Configure SSH timeouts and authentication parameters.

```
R1(config)#crypto key generate rsa
```

```
How many bits in the modulus [512]: 1024 R3(config)#show ip ssh
```

```
R3(config)#ip ssh time-out 90 R3(config)#ip ssh authentication-retries 2
```

```
R3(config)#ip ssh
```

```
version 2
```

4. Attempt to connect to R3 via Telnet from PC-C and Connect to R3 using SSH on PC-C.

```
PCC>telnet 192.168.3.1
```

PC>ssh -l SSHadmin 192.168.3.1

```
PC>telnet 192.168.3.1
Trying 192.168.3.1 ...Open

[Connection to 192.168.3.1 closed by foreign host]
PC>ssh -l SSHadmin 192.168.3.1
Open
Password:

R3#exit
```

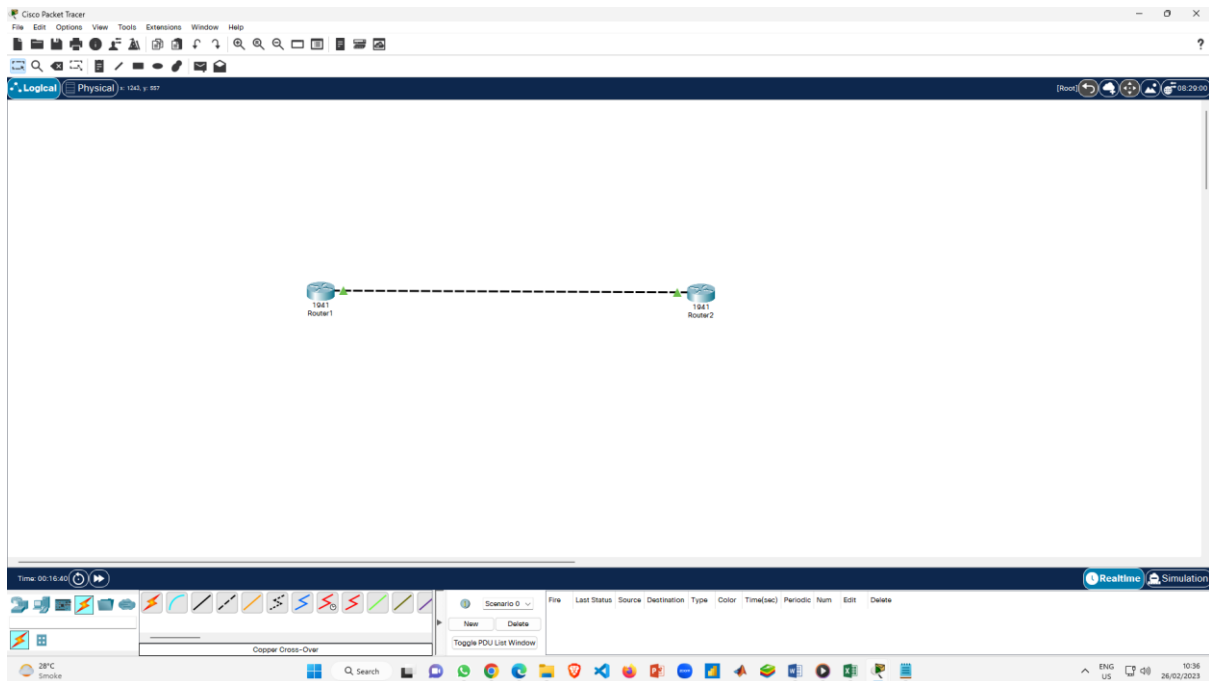
5. Connect to R3 using SSH on R2 and Check results.

```
R2#ssh -v 2 -l SSHadmin 10.2.2.1
R2#ssh -v 2 -l SSHadmin 10.2.2.1
Open
Password:
```

```
R3#|
```

## Practical-2

Aim: Configure AAA Authentication Topology:



Steps:

AAA configuration:

Now, in this example, we are configuring AAA Authentication on Router. It includes

following steps:

1. Enable AAA on router

R1>enable

R1#configure

terminal

Enter configuration commands, one per line. End with CNTL/Z.

R1(config)#aaa new-model

AAA is enabled by the command aaa new-model.

R1(config)#aaa authentication login default local

It enabled by the command aaa authentication login default local.

In this command, default means we will use the default method list and local means we

will use the local database.



## 2. Apply the list to vty lines

```
R1(config)#line vty 0 4
```

```
R1(config-line)#login authentication
```

```
defaultR1(config-line)#exit
```

After creating the default method list, we have to apply it to the vty lines so that whenever some user try to access the router through SSH or telnet the user has to provide credential which are configured.

## 3. Creating local user on the router

```
R1(config)#username cisco privilege 15 password cisco
```

This is the most important step as we have to create a local database in which we provide the username (as cisco), privilege level 15 and password (as cisco).

## 4. Debugging aaa authentication

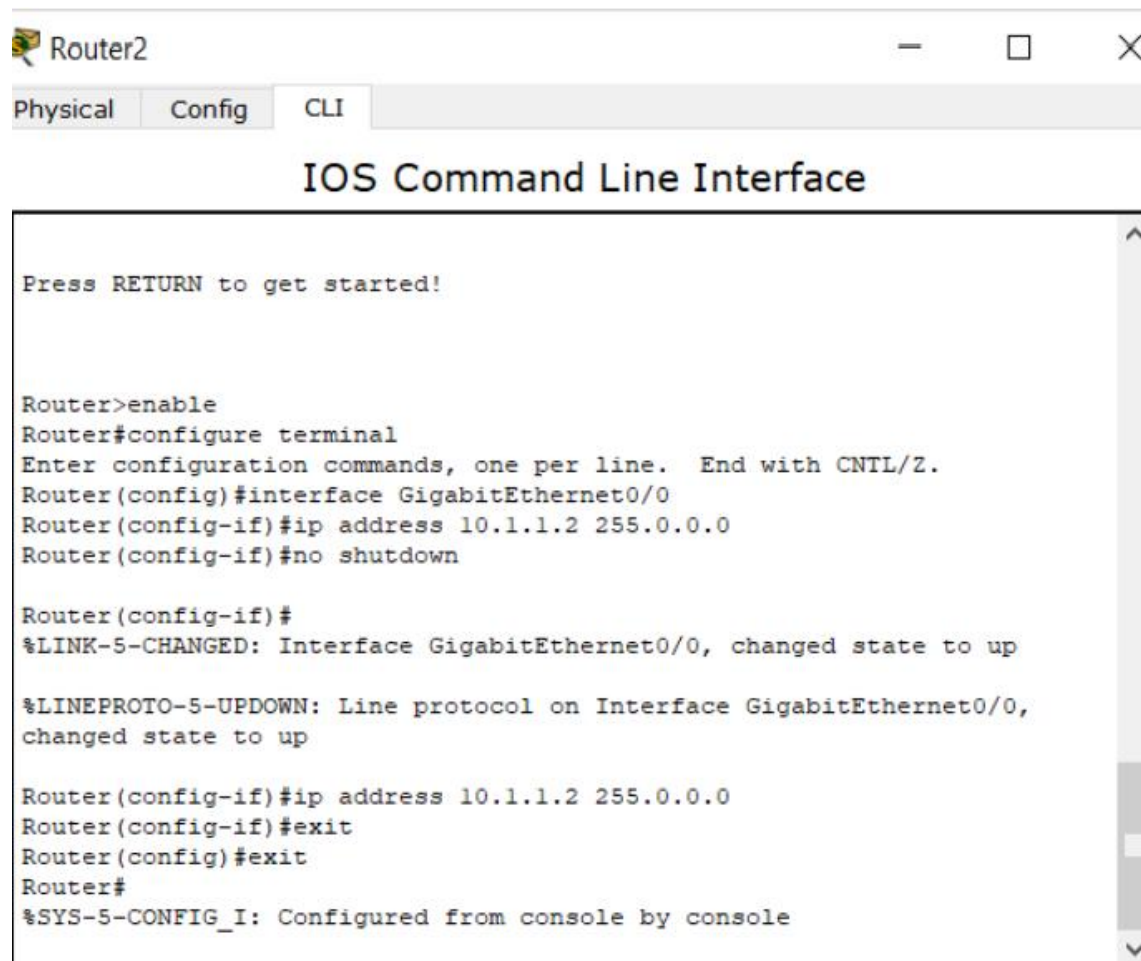
We can see the AAA authentication messages through the command

```
“debug aaaauthentication”
```

```
R1#debug aaa authentication
```

AAA Authentication debugging is on

Now we will telnet the router1 (ip address – 10.1.1.1/24) from router2 (ip address-10.1.1.2/24) and it will ask for the credential as shown in the figure.



The screenshot shows a window titled "Router2" with tabs for "Physical", "Config", and "CLI". The "CLI" tab is active, displaying the "IOS Command Line Interface". The text in the window is as follows:

```
Press RETURN to get started!

Router>enable
Router#configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
Router(config)#interface GigabitEthernet0/0
Router(config-if)#ip address 10.1.1.2 255.0.0.0
Router(config-if)#no shutdown

Router(config-if)#
%LINK-5-CHANGED: Interface GigabitEthernet0/0, changed state to up

%LINEPROTO-5-UPDOWN: Line protocol on Interface GigabitEthernet0/0,
changed state to up

Router(config-if)#ip address 10.1.1.2 255.0.0.0
Router(config-if)#exit
Router(config)#exit
Router#
%SYS-5-CONFIG_I: Configured from console by console
```

R2#telnet 10.1.1.1

```
Router#en
Router#telnet 10.1.1.1
Trying 10.1.1.1 ...Open

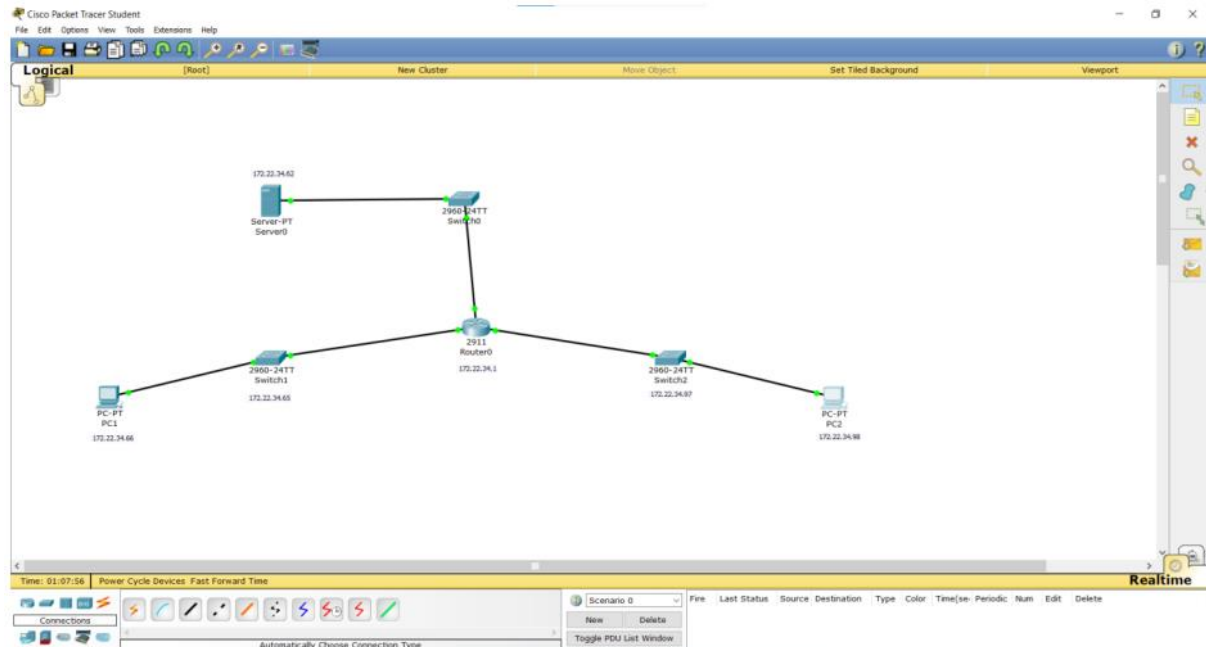
User Access Verification

Username: cisco
Password:
R1>
```

### Practical-3

Aim: Configuring Extended ACLs

Topology:



Address Table:

Device	Interface	IP Address	Subnet Mask	Default Gateway
R1	G0/0	172.22.34.65	255.255.255.224	NIL
	G0/1	172.22.34.97	255.255.255.240	NIL
	G0/2	172.22.34.1	255.255.255.192	NIL
Server	NIL	172.22.34.62	255.255.255.192	172.22.34.1
PC1	NIL	172.22.34.66	255.255.255.224	172.22.34.65
PC2	NIL	172.22.34.98	255.255.255.240	172.22.34.97

## A. Configure, Apply and Verify an Extended Numbered ACL

### 1. Configure an ACL to permit FTP and ICMP.

```
Router#en
```

```
Router#conf t
```

```
Router(config)#access-list ?
```

```
Router(config)#access-list 100 ?
```

```
Router(config)#access-list 100 permit ?
```

```
Router(config)#access-list 100 permit tcp ?
```

```
Router(config)#access-list 100 permit tcp 172.22.34.66 ?
```

Calculate the wildcard mask determining the binary opposite of a subnet mask.

11111111.11111111.11111111.11100000 = 255.255.255.224

00000000.00000000.00000000.00011111 = 0.0.0.31

```
Router(config)#access-list 100 permit tcp 172.22.34.66 0.0.0.31 ?
```

```
Router(config)#access-list 100 permit tcp 172.22.34.66 0.0.0.31 host  
172.22.34.62 ?
```

```
Router(config)#access-list 100 permit tcp 172.22.34.66 0.0.0.31 host  
172.22.34.62 eq ftp
```

```
Router(config)#access-list 100 permit tcp 172.22.34.64 0.0.0.31 host  
172.22.34.62
```

```
Router(config)#access-list 100 permit icmp 172.22.34.64 0.0.0.31 host  
172.22.34.62
```

### 2. Apply the ACL on the correct interface to filter traffic.

```
Router(config)#interface gigabitEthernet 0/0
```

```
Router(config-if)#ip access-group 100 in
```


### 3. Verify the ACL implementation.

Ping from PC1 to Server

Username & Password: cisco

```
ftp 172.22.34.62
```

```
ftp>quit
```

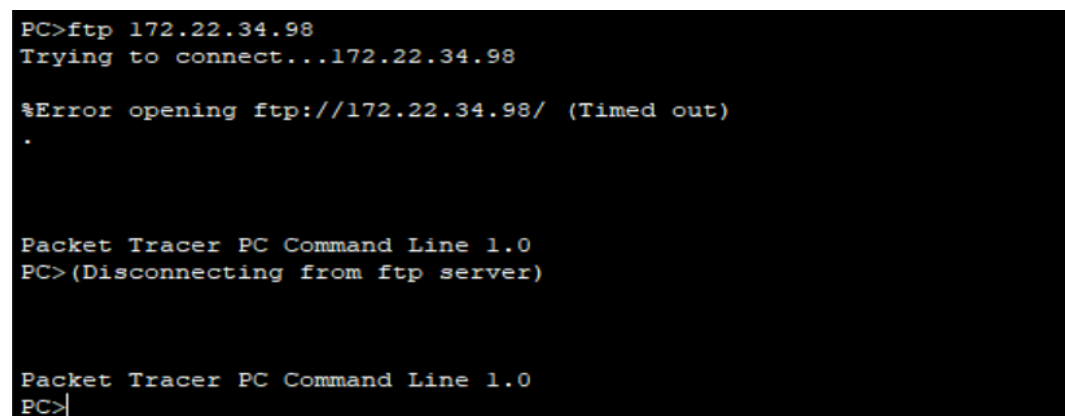


```
Command Prompt X

Packet Tracer PC Command Line 1.0
PC>ftp 172.22.34.62
Trying to connect...172.22.34.62
Connected to 172.22.34.62
220- Welcome to PT Ftp server
Username:cisco
331- Username ok, need password
Password:
230- Logged in
(passive mode On)
ftp>quit

Packet Tracer PC Command Line 1.0
PC>221- Service closing control connection.
PC>
```

Ping from PC1 to PC2. The destination host should be unreachable, because the traffic was not explicitly permitted.



```
PC>ftp 172.22.34.98
Trying to connect...172.22.34.98

%Error opening ftp://172.22.34.98/ (Timed out)
.

Packet Tracer PC Command Line 1.0
PC>(Disconnecting from ftp server)

Packet Tracer PC Command Line 1.0
PC>
```

## B. Configure, Apply and verify an extended Named ACL

### 1. Configure an ACL to permit HTTP access and ICMP.

```
Router>en
```

```
Router#conf t
```

```
Router(config)#ip access-list ?
```

```
Router(config)#ip access-list extended HTTP-ONLY
```

```
Router(config-ext-nacl)#permit tcp 172.22.34.98 ?
```

To calculate a wildcard is to subtract the subnet mask from

255.255.255.255. 255.255.255.255 - 255.255.255.240 ----- = 0. 0. 0. 15

```
Router(config-ext-nacl)#permit tcp 172.22.34.98 0.0.0.15 ?
```

```
Router(config-ext-nacl)#permit tcp 172.22.34.98 0.0.0.15 host 172.22.34.62 eq  
www
```

```
Router(config-ext-nacl)#permit icmp 172.22.34.98 0.0.0.15 host 172.22.34.62
```

```
Router(config-ext-nacl)#exit
```

```
Router(config)#interface gigabitEthernet 0/1
```

```
Router(config-if)#ip access-group HTTP_ONLY in
```

### 1. Verify connection between PC2 and Server using ping command

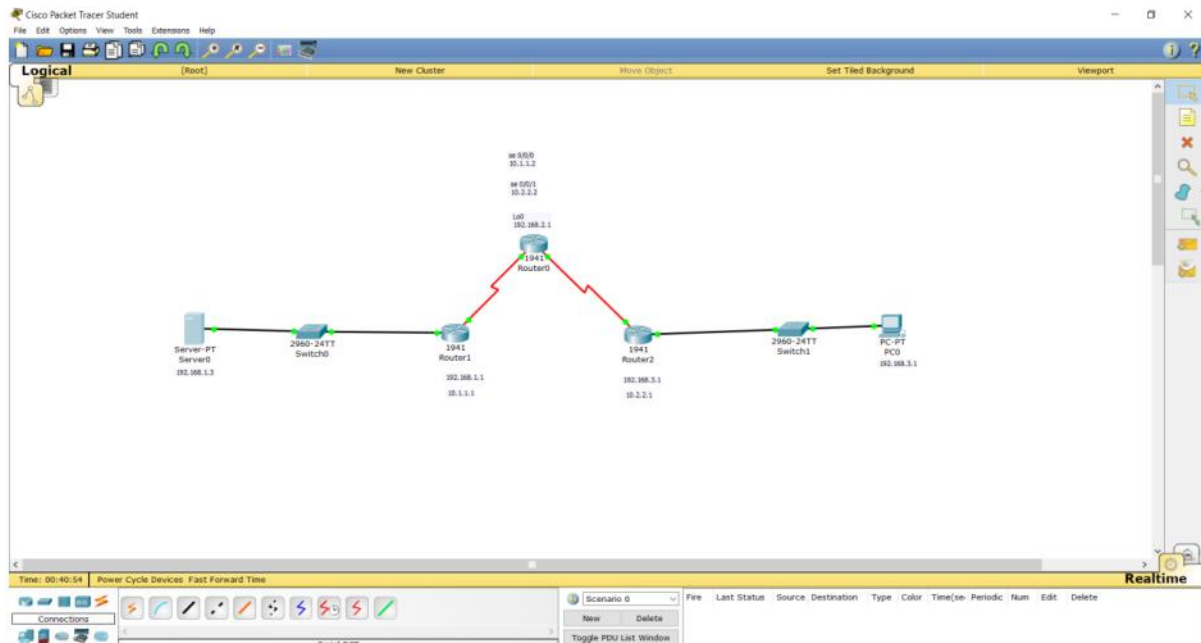
```
PC>ping 172.22.34.62  
  
Pinging 172.22.34.62 with 32 bytes of data:  
  
Reply from 172.22.34.62: bytes=32 time=1ms TTL=127  
Reply from 172.22.34.62: bytes=32 time=0ms TTL=127  
Reply from 172.22.34.62: bytes=32 time=0ms TTL=127  
Reply from 172.22.34.62: bytes=32 time=0ms TTL=127  
  
Ping statistics for 172.22.34.62:  
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),  
Approximate round trip times in milli-seconds:  
    Minimum = 0ms, Maximum = 1ms, Average = 0ms  
  
PC>|
```

2. Connect to server via PC2 browser.



## Practical-4

Aim: Configure IP ACLs to Mitigate Attacks Topology:



Addressing Table:

Device	Interface	IP Address	Subnet Mask	Default Gateway	Switch Port
R1	G0/1	192.168.1.1	255.255.255.0	N/A	S1 F0/5
	S0/0/0 (DCE)	10.1.1.1	255.255.255.252	N/A	N/A
R2	S0/0/0	10.1.1.2	255.255.255.252	N/A	N/A
	S0/0/1 (DCE)	10.2.2.2	255.255.255.252	N/A	N/A
	Lo0	192.168.2.1	255.255.255.0	N/A	N/A
R3	G0/1	192.168.3.1	255.255.255.0	N/A	S3 F0/5
	S0/0/1	10.2.2.1	255.255.255.252	N/A	N/A
PC-A	NIC	192.168.1.3	255.255.255.0	192.168.1.1	S1 F0/6
PC-C	NIC	192.168.3.3	255.255.255.0	192.168.3.1	S3 F0/18

Commands for SSh configuration

Perform all commands on R3 then on R1 and R2

Step 1:

Router>enable

Router#configure terminal

Enter configuration commands, one per line. End with CNTL/Z.

Router(config)#hostname R3



```
R3(config)#ip domain-name ccnasecurity.com
```

```
R3(config)#username SSHadmin privilege 15 secret ciscosshpa55
```

```
R3(config)#line vty 0 4
```

```
R3(config-line)#login local
```

```
R3(config-line)#transport input SSH
```

```
R3(config-line)#exit
```

```
R3(config)#crypto key generate RSA
```

The name for the keys will be: R3.ccnasecurity.com

Choose the size of the key modulus in the range of 360 to 2048 for your General Purpose Keys. Choosing a key modulus greater than 512 may take a few minutes.

How many bits in the modulus [512]: 1024

% Generating 1024 bit RSA keys, keys will be non-exportable...[OK]

```
R3(config)#ip ssh version 2
```

\*Mar 1 0:39:13.518: %SSH-5-ENABLED: SSH 1.99 has been enabled

```
R3(config)#ip ssh time-out 90
```

```
R3(config)#exit
```

Perform Loopback Interface Command on Router 2

```
Router>enable
```

```
Router#configure terminal
```

Enter configuration commands, one per line. End with CNTL/Z.

```
Router(config)#hostname R2
```

```
R2(config)#interface loopback 0
```

%LINK-5-CHANGED: Interface Loopback0, changed state to up

%LINEPROTO-5-UPDOWN: Line protocol on Interface Loopback0, changed state to up

```
R2(config-if)#ip address 192.168.2.1 255.255.255.0
```

```
R2(config-if)#no shut
```

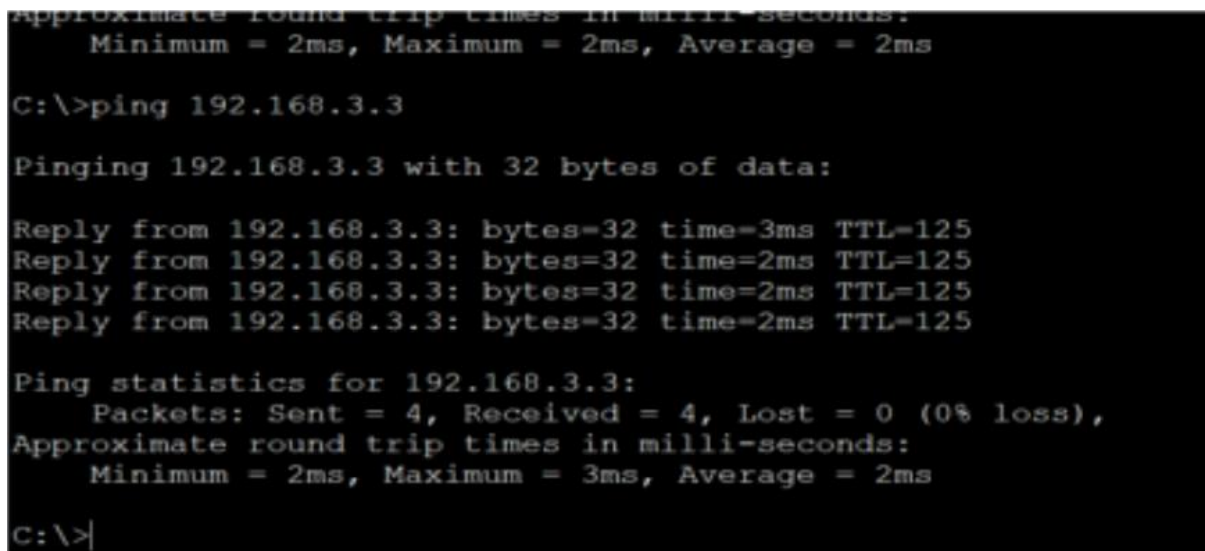
R2(config-if)#exit

### Part 1: Verify Basic Network Connectivity

Verify network connectivity prior to configuring the IP ACLs.

Step 1: From PC-A, verify connectivity to PC-C and R2.

a. From the command prompt, ping PC-C (192.168.3.3).



```
Approximate round trip times in milli-seconds:
  Minimum = 2ms, Maximum = 2ms, Average = 2ms

C:\>ping 192.168.3.3

Pinging 192.168.3.3 with 32 bytes of data:

Reply from 192.168.3.3: bytes=32 time=3ms TTL=125
Reply from 192.168.3.3: bytes=32 time=2ms TTL=125
Reply from 192.168.3.3: bytes=32 time=2ms TTL=125
Reply from 192.168.3.3: bytes=32 time=2ms TTL=125

Ping statistics for 192.168.3.3:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
    Approximate round trip times in milli-seconds:
        Minimum = 2ms, Maximum = 3ms, Average = 2ms

C:\>|
```

b. From the command prompt, establish an SSH session to R2 Lo0 interface (192.168.2.1) using username SSHadmin and password ciscosshpa55. When finished, exit the SSH session.

SERVER> ssh -l SSHadmin 192.168.2.1

```
Approximate round trip times in milli-seconds:
    Minimum = 2ms, Maximum = 3ms, Average = 2ms

C:\>ssh -l SSHadmin 192.168.2.1

% Connection timed out; remote host not responding
C:\>ssh -l SSHadmin 192.168.2.1

Password:

R2#exit

[Connection to 192.168.2.1 closed by foreign host]
C:\>
```

Step 2: From PC-C, verify connectivity to PC-A and R2. a. From the command prompt, ping PC-A (192.168.1.3).

```
Cisco Packet Tracer PC Command Line 1.0
C:\>ping 192.168.1.3

Pinging 192.168.1.3 with 32 bytes of data:

Reply from 192.168.1.3: bytes=32 time=2ms TTL=125
Reply from 192.168.1.3: bytes=32 time=2ms TTL=125
Reply from 192.168.1.3: bytes=32 time=2ms TTL=125
Reply from 192.168.1.3: bytes=32 time=2ms TTL=125

Ping statistics for 192.168.1.3:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 2ms, Maximum = 2ms, Average = 2ms
```

b. From the command prompt, establish an SSH session to R2 Lo0 interface (192.168.2.1) using username SSHadmin and password ciscosshpa55. Close the SSH session when finished. PC> ssh -l SSHadmin 192.168.2.1

```
Ping statistics for 192.168.1.3:
    Packets: Sent = 4, Received = 4, Lost = 0 (0%
loss),
Approximate round trip times in milli-seconds:
    Minimum = 2ms, Maximum = 2ms, Average = 2ms

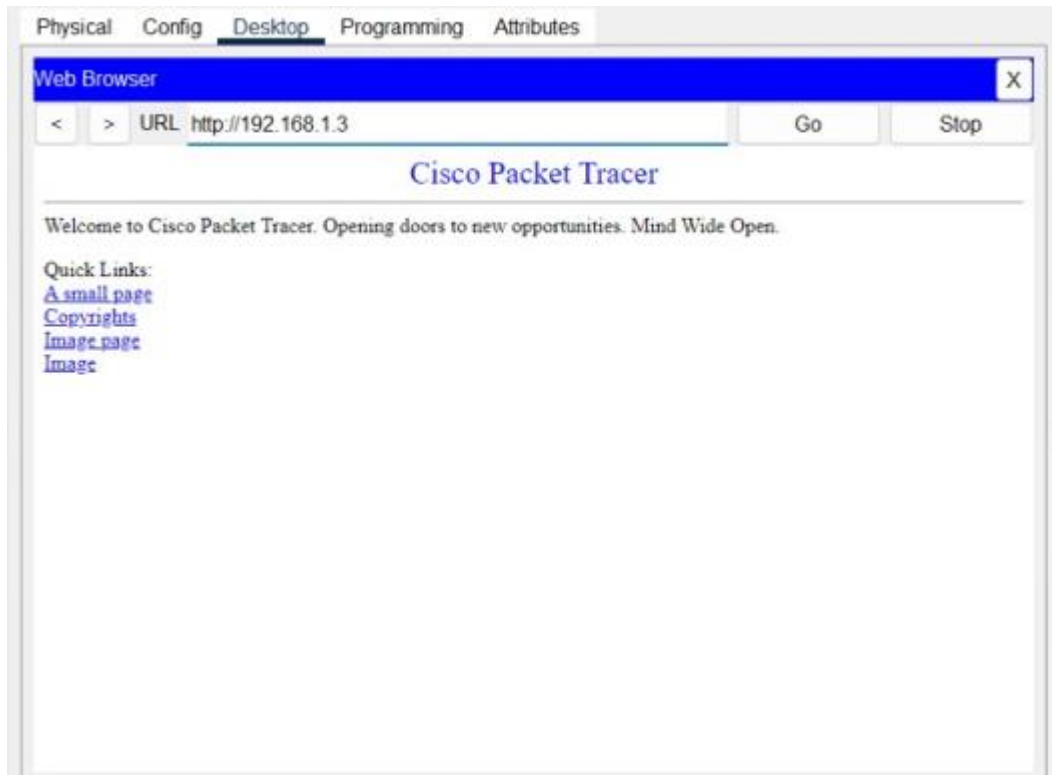
C:\>ssh -l SSHadmin 192.168.2.1

Password:

R2#exit

[Connection to 192.168.2.1 closed by foreign host]
C:\>|
```

c. Open a web browser to the PC-A server (192.168.1.3) to display the web page. Close the browser when done.



## Part 2: Secure Access to Routers

Step 1: Configure ACL 10 to block all remote access to the routers except from PC-C.

Use the access-list command to create a numbered IP ACL on R1, R2, and R3.

```
R1(config)# access-list 10 permit host 192.168.3.3
```

```
R2(config)# access-list 10 permit host 192.168.3.3
```

```
R3(config)# access-list 10 permit host 192.168.3.3
```

Step 2: Apply ACL 10 to ingress traffic on the VTY lines.

Use the access-class command to apply the access list to incoming traffic on the VTY lines.

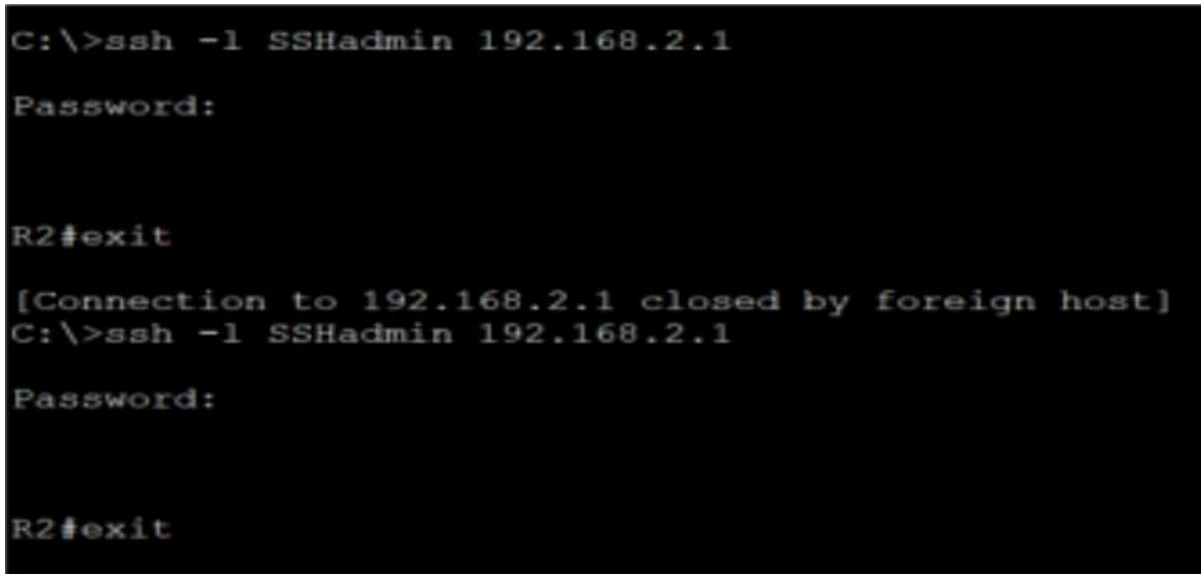
R1(config-line)# access-class 10 in

R2(config-line)# access-class 10 in

R3(config-line)# access-class 10 in

Step 3: Verify exclusive access from management station PC-C.(should be successful)

a. PC> ssh -l SSHAdmin 192.168.2.1



```
C:\>ssh -l SSHAdmin 192.168.2.1
Password:

R2#exit

[Connection to 192.168.2.1 closed by foreign host]
C:\>ssh -l SSHAdmin 192.168.2.1
Password:

R2#exit
```

b. Establish an SSH session to 192.168.2.1 from PC-A (should fail).

Establish an SSH session to 192.168.2.1 from PC-C (should be successful).



```
[Connection to 192.168.2.1 closed by foreign host]
C:\>ssh -l SSHAdmin 192.168.2.1
% Connection refused by remote host
C:\>ssh -l SSHAdmin 192.168.2.1
% Connection refused by remote host
C:\>ssh -l SSHAdmin 192.168.2.1
% Connection refused by remote host
C:\>ssh -l SSHAdmin 192.168.2.1
% Connection refused by remote host
C:\>ssh -l SSHAdmin 192.168.2.1
% Connection refused by remote host
C:\>
```

```
C:\>ssh -l SSHadmin 192.168.2.1
Password:

R2#exit

[Connection to 192.168.2.1 closed by foreign host]
C:\>ssh -l SSHadmin 192.168.2.1
Password:

R2#exit
```

Part 3: Create a Numbered IP ACL 120 on R1

Create an IP ACL numbered 120 with the following rules:

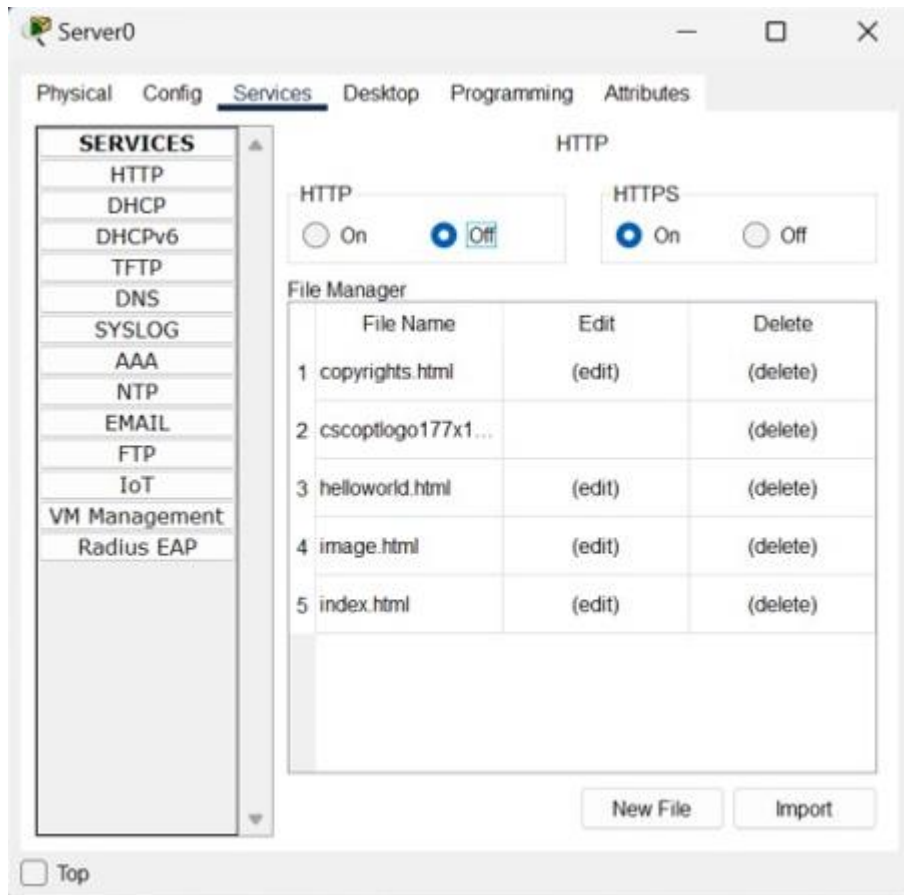
- o Permit any outside host to access DNS, SMTP, and FTP services on server PC-A.
- o Deny any outside host access to HTTPS services on PC-A.
- o Permit PC-C to access R1 via SSH.

Note: Check Results will not show a correct configuration for ACL 120 until you modify it in

Part 4.

Step 1: Verify that PC-C can access the PC-A via HTTPS using the web browser.

Be sure to disable HTTP and enable HTTPS on server PC-A.



Step 2: Configure ACL 120 to specifically permit and deny the specified traffic.

Use the access-list command to create a numbered IP ACL.

```
R1(config)#access-list 120 permit udp any host 192.168.1.3 eq domain
```

```
R1(config)#access-list 120 permit tcp any host 192.168.1.3 eq smtp
```

```
R1(config)#access-list 120 permit tcp any host 192.168.1.3 eq ftp
```

```
R1(config)#access-list 120 deny tcp any host 192.168.1.3 eq 443
```

```
R1(config)#access-list 120 permit tcp host 192.168.3.3 host 10.1.1.1 eq 22
```

Step 3: Apply the ACL to interface S0/0/0.

Use the ip access-group command to apply the access list to incoming traffic on interface

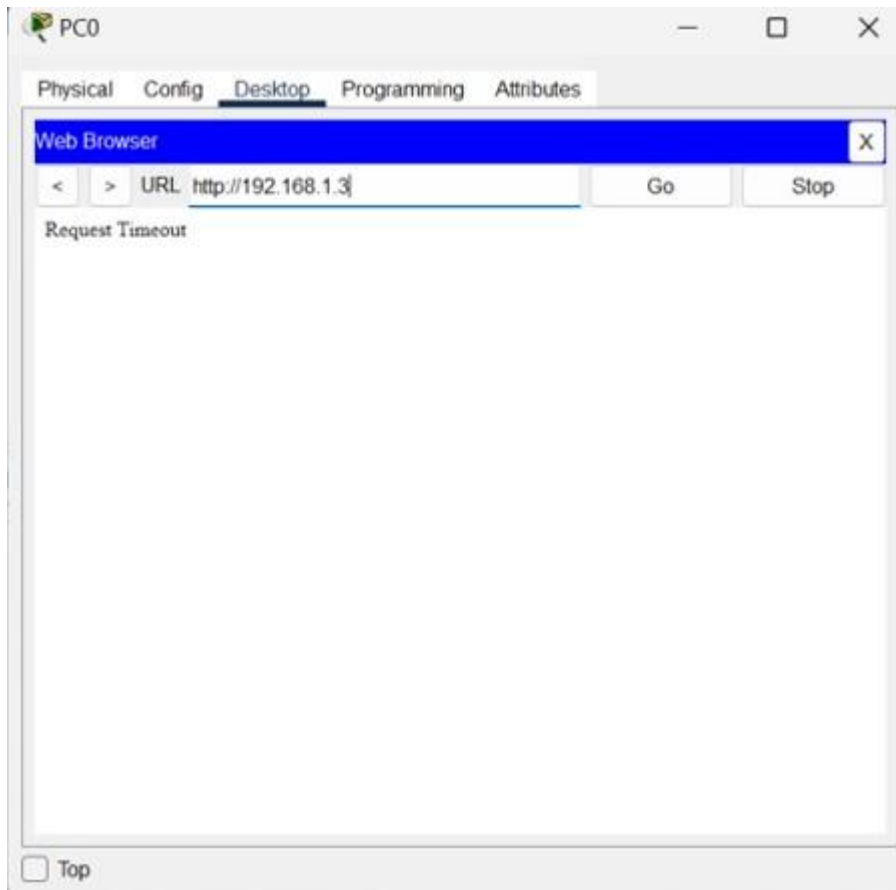
S0/0/0.

```
R1(config)#interface s0/0/0
```

```
R1(config-if)#ip access-group 120 in
```

Step 4: Verify that PC-C cannot access PC-A via HTTPS using the web browser.





#### Part 4: Modify an Existing ACL on R1

Permit ICMP echo replies and destination unreachable messages from the outside network

(relative to R1).

Deny all other incoming ICMP packets.

Step 1: Verify that PC-A cannot successfully ping the loopback interface on R2.

Step 2: Make any necessary changes to ACL 120 to permit and deny the specified traffic.

Use the access-list command to create a numbered IP ACL.

```
R1(config)#access-list 120 permit icmp any any echo-reply
```

```
R1(config)#access-list 120 permit icmp any any unreachable
```

```
R1(config)#access-list 120 deny icmp any any
```

```
R1(config)#access-list 120 permit ip any any
```

Step 3: Verify that PC-A can successfully ping the loopback interface on R2.

```
Command Prompt
C:\>ping 192.168.2.1

Pinging 192.168.2.1 with 32 bytes of data:

Reply from 192.168.2.1: bytes=32 time=1ms
TTL=254
Reply from 192.168.2.1: bytes=32 time=1ms
TTL=254
Reply from 192.168.2.1: bytes=32 time=1ms
TTL=254
Reply from 192.168.2.1: bytes=32 time=1ms
TTL=254

Ping statistics for 192.168.2.1:
    Packets: Sent = 4, Received = 4, Lost = 0
    (0% loss),
    Approximate round trip times in milli-seconds:
        Minimum = 1ms, Maximum = 1ms, Average = 1ms
```

Part 5: Create a Numbered IP ACL 110 on R3

Deny all outbound packets with source address outside the range of internal IP addresses on

R3.

Step 1: Configure ACL 110 to permit only traffic from the inside network.

Use the access-list command to create a numbered IP ACL.

```
R3(config)# access-list 110 permit ip 192.168.3.0 0.0.0.255 any
```

Step 2: Apply the ACL to interface G0/1.

Use the ip access-group command to apply the access list to incoming traffic on interface

G0/1.

```
R3(config)# interface g0/1
```

```
R3(config-if)# ip access-group 110 in
```

```
R3(config)#access-list 110 permit ip 192.168.3.0 0.0.0.255 any
```

```
R3(config)#interface g0/0
```

```
R3(config-if)#ip access-group 110 in
```

Part 6: Create a Numbered IP ACL 100 on R3

On R3, block all packets containing the source IP address from the following pool of

addresses: any RFC 1918 private addresses, 127.0.0.0/8, and any IP multicast address. Since

PC-C is being used for remote administration, permit SSH traffic from the 10.0.0.0/8 network

to return to the host PC-C.

Step 1: Configure ACL 100 to block all specified traffic from the outside network.

You should also block traffic sourced from your own internal address space if it is not an

RFC 1918 address. In this activity, your internal address space is part of the private address

space specified in RFC 1918.

Use the access-list command to create a numbered IP ACL.

```
R3(config)# access-list 100 permit tcp 10.0.0.0 0.255.255.255 eq 22 host 192.168.3.3
```

```
R3(config)# access-list 100 deny ip 10.0.0.0 0.255.255.255 any
```

```
R3(config)# access-list 100 deny ip 172.16.0.0 0.15.255.255 any
```

```
R3(config)# access-list 100 deny ip 192.168.0.0 0.0.255.255 any
```

```
R3(config)# access-list 100 deny ip 127.0.0.0 0.255.255.255 any
```

```
R3(config)# access-list 100 deny ip 224.0.0.0 15.255.255.255 any
```

```
R3(config)# access-list 100 permit ip any any
```

Step 2: Apply the ACL to interface Serial 0/0/1.

Use the ip access-group command to apply the access list to incoming traffic on interface

Serial 0/0/1.

```
R3(config)# interface s0/0/1
```

R3(config-if)# ip access-group 100 in

Step 3: Confirm that the specified traffic entering interface Serial 0/0/1 is handled correctly.

a. From the PC-C command prompt, ping the PC-A server. The ICMP echo replies are

blocked by the ACL since they are sourced from the 192.168.0.0/16 address space.

```
% Connection timed out; remote host not responding
C:\>ssh -l SSHadmin 192.168.2.1

% Connection timed out; remote host not responding
C:\>ping 192.168.1.3

Pinging 192.168.1.3 with 32 bytes of data:

Reply from 192.168.3.1: Destination host unreachable.
Reply from 192.168.3.1: Destination host unreachable.
Reply from 192.168.3.1: Destination host unreachable.
Reply from 192.168.3.1: Destination host unreachable.

Ping statistics for 192.168.1.3:
    Packets: Sent = 4, Received = 0, Lost = 4 (100%
loss),
```

b. Establish an SSH session to 192.168.2.1 from PC-C (should be successful)

```
Ping statistics for 192.168.1.3:
    Packets: Sent = 4, Received = 0, Lost = 4 (100%
loss),

C:\>ssh -l SSHadmin 192.168.2.1

% Connection timed out; remote host not responding
C:\>ssh -l SSHadmin 192.168.2.1

Password:

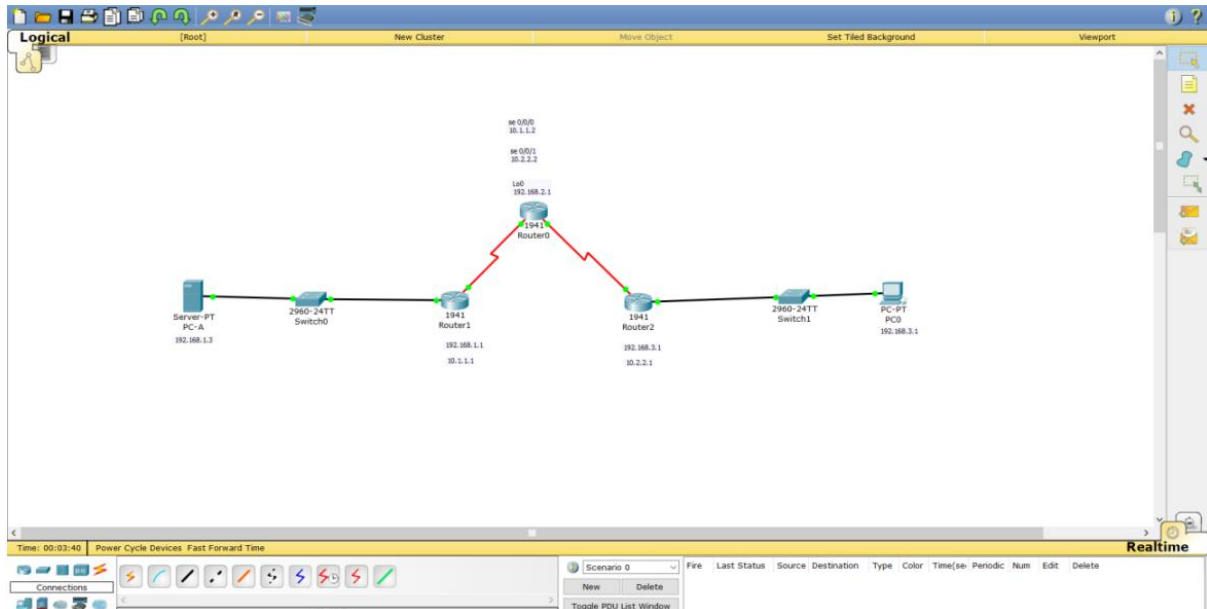
R2#exit

[Connection to 192.168.2.1 closed by foreign host]
C:\>
```

## Practical-5

Aim: Configuring a zone based policy firewall

Topology:



Addressing Table:

Device	Interface	IP Address	Subnet Mask	Default Gateway	Switch Port
R1	G0/1	192.168.1.1	255.255.255.0	N/A	S1 F0/5
	S0/0/0 (DCE)	10.1.1.1	255.255.255.252	N/A	N/A
R2	S0/0/0	10.1.1.2	255.255.255.252	N/A	N/A
	S0/0/1 (DCE)	10.2.2.2	255.255.255.252	N/A	N/A
R3	G0/1	192.168.3.1	255.255.255.0	N/A	S3 F0/5
	S0/0/1	10.2.2.1	255.255.255.252	N/A	N/A
PC-A	NIC	192.168.1.3	255.255.255.0	192.168.1.1	S1 F0/6
PC-C	NIC	192.168.3.3	255.255.255.0	192.168.3.1	S3 F0/18

### Part 1: Verify Basic Network Connectivity

Verify network connectivity prior to configuring the zone-based policy firewall.

Step 1: From the PC-A command prompt, ping PC-C at 192.168.3.3

```
Command Prompt
Approximate round trip times in milli-seconds:
    Minimum = 2ms, Maximum = 4ms, Average = 2ms

C:\>ping 192.168.3.3

Pinging 192.168.3.3 with 32 bytes of data:

Reply from 192.168.3.3: bytes=32 time=2ms TTL=125
Reply from 192.168.3.3: bytes=32 time=2ms TTL=125
Reply from 192.168.3.3: bytes=32 time=2ms TTL=125
Reply from 192.168.3.3: bytes=32 time=2ms TTL=125

Ping statistics for 192.168.3.3:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
    Approximate round trip times in milli-seconds:
        Minimum = 2ms, Maximum = 2ms, Average = 2ms

C:\>|
```

Step 2: Access R2 using SSH.

a. From the PC-C command prompt, SSH to the S0/0/1 interface on R2 at 10.2.2.2. Use the:

Password – ciscosshpa55

PC> ssh -l Admin 10.2.2.2

b. Exit the SSH session.

```
Cisco Packet Tracer PC Command Line 1.0
C:\>ssh -l SSHAdmin 10.2.2.2

Password:

R2#exit

[Connection to 10.2.2.2 closed by
foreign host]
C:\>|
```

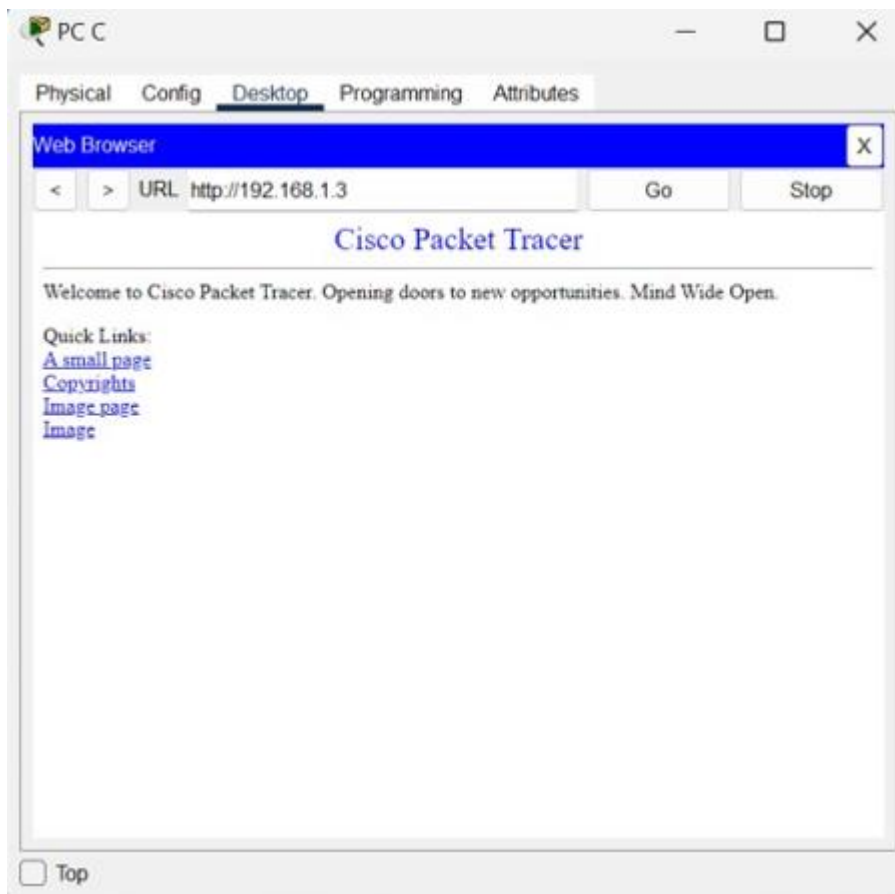
Step 3: From PC-C, open a web browser to the PC-A server.

a. Click the Desktop tab and then click the Web Browser application. Enter the PC-A IP

address 192.168.1.3 as the URL. The Packet Tracer welcome page from the web server

should be displayed.

b. Close the browser on PC-C



Part 2: Create the Firewall Zones on R3

Note: For all configuration tasks, be sure to use the exact names as specified.

Step 1: Enable the Security Technology package.

a. On R3, issue the show version command to view the Technology Package license

information.

b. If the Security Technology package has not been enabled, use the following command to

enable the package.

R3(config)# license boot module c1900 technology-package securityk9

c. Accept the end-user license agreement.

d. Save the running-config and reload the router to enable the security license.

e. Verify that the Security Technology package has been enabled by using the show version

command.

```

255K bytes of non-volatile configuration memory.
249856K bytes of ATA System CompactFlash 0 (Read/Write)

License Info:
License UDI:

-----
Device#    PID          SN
-----
*0         CISCO1941/K9    FTX15241R05-

Technology Package License Information for Module:'c1900'
-----
Technology    Technology-package    Technology-package
Current       Type                 Next reboot
-----
ipbase        ipbasek9             Permanent          ipbasek9
security      securityk9            Evaluation         securityk9
data          disable              None               None

Configuration register is 0x2102

R3#

```

Step 2: Create an internal zone.

Use the zone security command to create a zone named IN-ZONE.

R3(config)# zone security IN-ZONE

R3(config-sec-zone) exit

Step 3: Create an external zone.

Use the zone security command to create a zone named OUT-ZONE.

R3(config-sec-zone)# zone security OUT-ZONE

R3(config-sec-zone)# exit

Part 3: Identify Traffic Using a Class-Map

Step 1: Create an ACL that defines internal traffic.

Use the access-list command to create extended ACL 101 to permit all IP protocols from the

192.168.3.0/24 source network to any destination.



```
R3(config)# access-list 101 permit ip 192.168.3.0 0.0.0.255 any
```

Step 2: Create a class map referencing the internal traffic ACL.

Use the class-map type inspect command with the match-all option to create a class map

named IN-NET- CLASS-MAP. Use the match access-group command to match ACL 101.

```
R3(config)# class-map type inspect match-all IN-NET-CLASS-MAP
```

```
R3(config-cmap)# match access-group 101
```

```
R3(config-cmap)# exit
```

#### Part 4: Specify Firewall Policies

Step 1: Create a policy map to determine what to do with matched traffic.

Use the policy-map type inspect command and create a policy map named IN-2-OUT-PMAP.

```
R3(config)# policy-map type inspect IN-2-OUT-PMAP
```

Step 2: Specify a class type of inspect and reference class map IN-NET-CLASS-MAP.

```
R3(config-pmap)# class type inspect IN-NET-CLASS-MAP
```

Step 3: Specify the action of inspect for this policy map.

The use of the inspect command invokes context-based access control (other options include

pass and drop).

```
R3(config-pmap-c)# inspect
```

%No specific protocol configured in class IN-NET-CLASS-MAP for inspection. All

protocols will be inspected.

Issue the exit command twice to leave config-pmap-c mode and return to config mode.

```
R3(config-pmap-c)# exit
```

```
R3(config-pmap)# exit
```

#### Part 5: Apply Firewall Policies

Step 1: Create a pair of zones.

Using the zone-pair security command, create a zone pair named IN-2-OUT-ZPAIR. Specify

the source and destination zones that were created in Task 1.

```
R3(config)# zone-pair security IN-2-OUT-ZPAIR source IN-ZONE destination OUTZONE
```

Step 2: Specify the policy map for handling the traffic between the two zones.

Attach a policy-map and its associated actions to the zone pair using the service-policy type

inspect command and reference the policy map previously created, IN-2-OUT-PMAP.

```
R3(config-sec-zone-pair)# service-policy type inspect IN-2-OUT-PMAP
```

```
R3(config-sec-zone-pair)# exit
```

Step 3: Assign interfaces to the appropriate security zones.

Use the zone-member security command in interface configuration mode to assign G0/1 to

IN-ZONE and S0/0/1 to OUT-ZONE.

```
R3(config)# interface g0/1
```

```
R3(config-if)# zone-member security IN-ZONE
```

```
R3(config-if)# exit
```

```
R3(config)# interface s0/0/1
```

```
R3(config-if)# zone-member security OUT-ZONE
```

```
R3(config-if)# exit
```

Part 6: Test Firewall Functionality from IN-ZONE to OUT-ZONE

Verify that internal hosts can still access external resources after configuring the ZPF.

Step 1: From internal PC-C, ping the external PC-A server.

From the PC-C command prompt, ping PC-A at 192.168.1.3. The ping should succeed.

```
C:\>ping 192.168.1.3

Pinging 192.168.1.3 with 32 bytes of data:

Reply from 192.168.1.3: bytes=32 time=2ms TTL=125
Reply from 192.168.1.3: bytes=32 time=2ms TTL=125
Reply from 192.168.1.3: bytes=32 time=2ms TTL=125
Reply from 192.168.1.3: bytes=32 time=2ms TTL=125

Ping statistics for 192.168.1.3:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
    Approximate round trip times in milli-seconds:
        Minimum = 2ms, Maximum = 2ms, Average = 2ms

C:\>
```

Step 2: From internal PC-C, SSH to the R2 S0/0/1 interface.

a. From the PC-C command prompt, SSH to R2 at 10.2.2.2. Use the username Admin and the

password ciscosshpa55 to access R2. The SSH session should succeed.

```
Reply from 192.168.1.3: bytes=32 time=2ms TTL=125
Reply from 192.168.1.3: bytes=32 time=2ms TTL=125
Reply from 192.168.1.3: bytes=32 time=2ms TTL=125
Reply from 192.168.1.3: bytes=32 time=2ms TTL=125

Ping statistics for 192.168.1.3:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
    Approximate round trip times in milli-seconds:
        Minimum = 2ms, Maximum = 2ms, Average = 2ms

C:\>ssh -l SSHadmin 10.2.2.2

Password:

R2#|
```

b. While the SSH session is active, issue the command `show policy-map type inspect zonepair`

sessions on R3 to view established sessions.

R3# `show policy-map type inspect zone-pair sessions`

```
R3#show policy-map type inspect zone-pair sessions
policy exists on zp IN-2-OUT-ZPAIR
Zone-pair: IN-2-OUT-ZPAIR

Service-policy inspect : IN-2-OUT-PMAP

Class-map: IN-NET-CLASS-MAP (match-all)
  Match: access-group 101
  Inspect

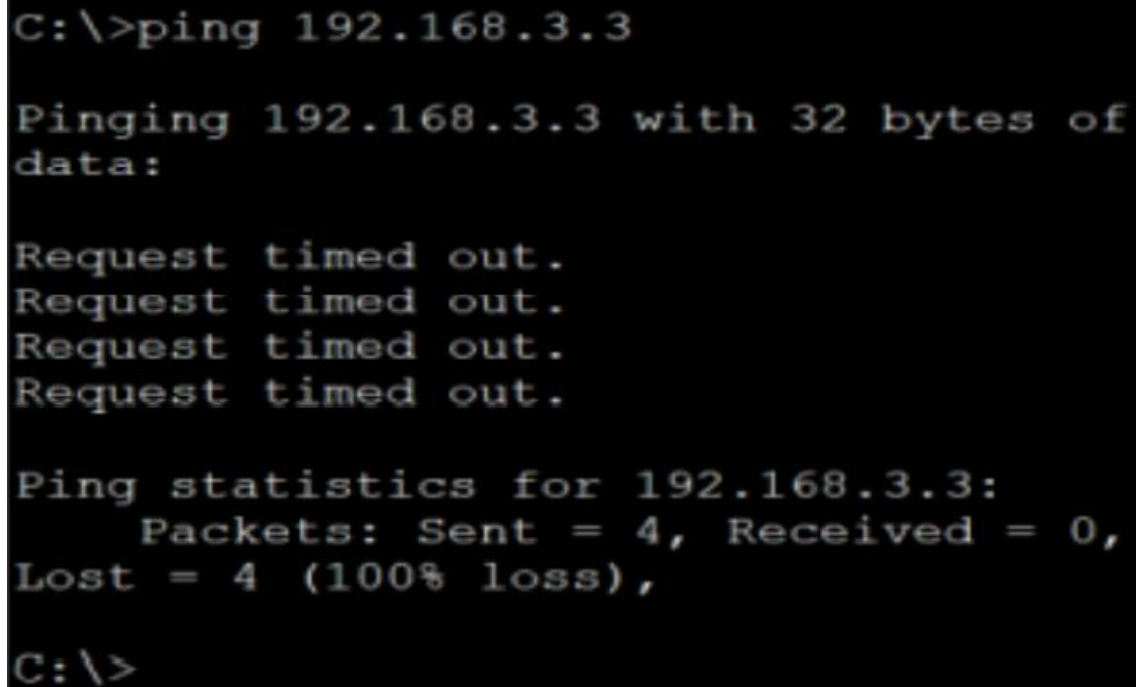
    Number of Established Sessions = 1
    Established Sessions
      Session 2985064112 (192.168.3.3:1027)=>(10.2.2.2:22) tcp SIS_OPEN/TCP_ESTAB
        Created 00:01:14, Last heard 00:01:11
        Bytes sent (initiator:responder) [1310:1018]
Class-map: class-default (match-any)
  Match: any
  Drop (default action)
    0 packets, 0 bytes
```

### Part 7: Test Firewall Functionality from OUT-ZONE to IN-ZONE

Verify that external hosts CANNOT access internal resources after configuring the ZPF.

Step 1: From the PC-A server command prompt, ping PC-C.

From the PC-A command prompt, ping PC-C at 192.168.3.3. The ping should fail.



```
C:\>ping 192.168.3.3

Pinging 192.168.3.3 with 32 bytes of data:

Request timed out.
Request timed out.
Request timed out.
Request timed out.

Ping statistics for 192.168.3.3:
    Packets: Sent = 4, Received = 0,
    Lost = 4 (100% loss),

C:\>
```

Step 2: From R2, ping PC-C.

From R2, ping PC-C at 192.168.3.3. The ping should fail.

```
Serial0/0/1, changed state to up

%LINEPROTO-5-UPDOWN: Line protocol on
Interface Serial0/0/1, changed state
to up

R2>ping 192.168.3.3

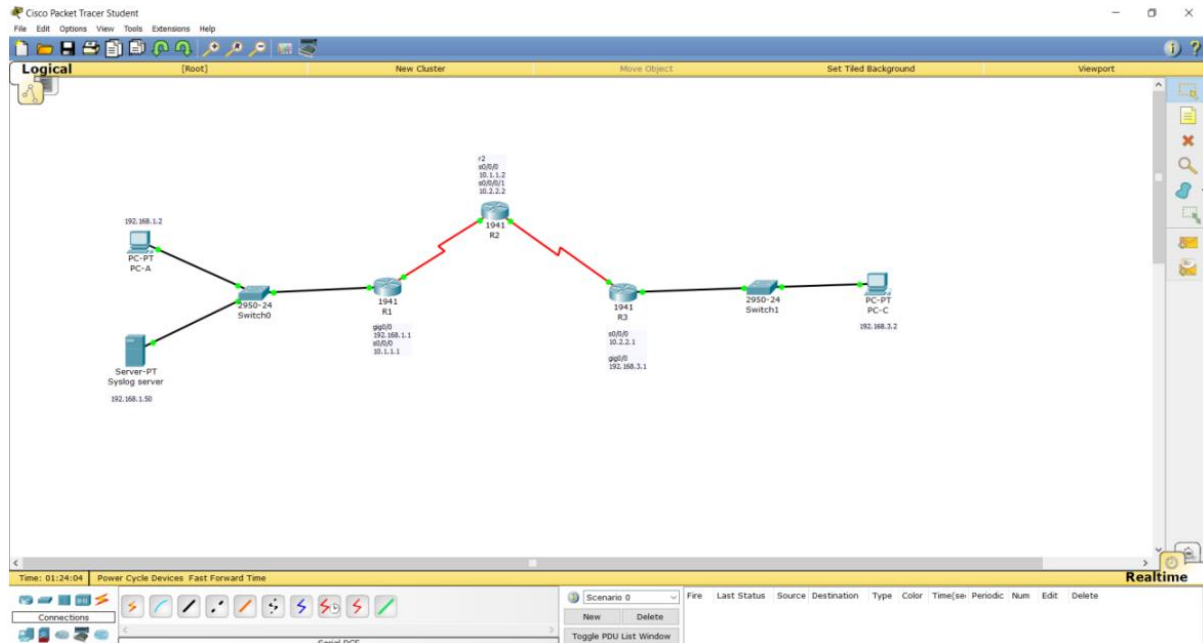
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to
192.168.3.3, timeout is 2 seconds:
.....
Success rate is 0 percent (0/5)

R2>|
```

## Practical-6

Aim: Configure IOS Intrusion Prevention System(IPS) Using the CLI

Topology:



Address Table:

Device	Interface	IP Address	Subnet Mask	Default Gateway	Switch Port
R1	G0/1	192.168.1.1	255.255.255.0	N/A	S1 F0/1
	S0/0/0	10.1.1.1	255.255.255.252	N/A	N/A
R2	S0/0/0 (DCE)	10.1.1.2	255.255.255.252	N/A	N/A
	S0/0/1 (DCE)	10.2.2.2	255.255.255.252	N/A	N/A
R3	G0/1	192.168.3.1	255.255.255.0	N/A	S3 F0/1
	S0/0/0	10.2.2.1	255.255.255.252	N/A	N/A
Syslog	NIC	192.168.1.50	255.255.255.0	192.168.1.1	S1 F0/2
PC-A	NIC	192.168.1.2	255.255.255.0	192.168.1.1	S1 F0/3
PC-C	NIC	192.168.3.2	255.255.255.0	192.168.3.1	S3 F0/2

Step 1: Implement SSH on R1 R2 and R3

R1

Router>

Router>enable

Router#configure terminal

Enter configuration commands, one per line. End with CNTL/Z.

```
Router(config)#hostname r1
r1(config)#ip domain-name ccnasecurity.com
r1(config)#username SSHadmin privilege 15 secret ciscosshpa55
r1(config)#line vty 0 4
r1(config-line)#login local
r1(config-line)#transport input SSH
r1(config-line)#exit
r1(config)#crypto key generate RSA
The name for the keys will be: r1.ccnasecurity.com
Choose the size of the key modulus in the range of 360 to 2048 for your
General Purpose Keys. Choosing a key modulus greater than 512 may take
a few minutes.
How many bits in the modulus [512]: 1024
% Generating 1024 bit RSA keys, keys will be non-exportable...[OK]
r1(config)#ip ssh version 2
*Mar 1 0:26:18.478: %SSH-5-ENABLED: SSH 1.99 has been enabled
r1(config)#ip ssh time-out 90
r1(config)#exit
%SYS-5-CONFIG_I: Configured from console by console
r1#show ip ssh
SSH Enabled - version 2.0
Authentication timeout: 90 secs; Authentication retries: 3
```



R2

Router>

Router>enable

Router#configure terminal

Enter configuration commands, one per line. End with CNTL/Z.

Router(config)#hostname r2

r2(config)#ip domain-name ccnasecurity.com

r2(config)#username SSHadmin privilege 15 secret ciscosshpa55

r2(config)#line vty 0 4

r2(config-line)#login local

r2(config-line)#transport input SSH

r2(config-line)#exit

r2(config)#crypto key generate RSA

The name for the keys will be: r2.ccnasecurity.com

Choose the size of the key modulus in the range of 360 to 2048 for your General Purpose Keys. Choosing a key modulus greater than 512 may take a few minutes.

How many bits in the modulus [512]: 1024

% Generating 1024 bit RSA keys, keys will be non-exportable...[OK]

r2(config)#ip ssh version 2

\*Mar 1 0:29:20.319: %SSH-5-ENABLED: SSH 1.99 has been enabled

r2(config)#ip ssh time-out 90

r2(config)#exit

%SYS-5-CONFIG\_I: Configured from console by console

r2#show ip ssh

SSH Enabled - version 2.0

Authentication timeout: 90 secs; Authentication retries:

R3

Router>

Router>enable

Router#configure terminal

Enter configuration commands, one per line. End with CNTL/Z.

Router(config)#hostname r3

r3(config)#ip domain-name ccnasecurity.com

r3(config)#username SSHadmin privilege 15 secret ciscosshpa55

r3(config)#line vty 0 4

r3(config-line)#login local

r3(config-line)#transport input SSH

r3(config-line)#exit

r3(config)#crypto key generate RSA

The name for the keys will be: r3.ccnasecurity.com

Choose the size of the key modulus in the range of 360 to 2048 for your General Purpose Keys. Choosing a key modulus greater than 512 may take a few minutes.

How many bits in the modulus [512]: 1024

% Generating 1024 bit RSA keys, keys will be non-exportable...[OK]

r3(config)#ip ssh version 2

\*Mar 1 0:33:2.850: %SSH-5-ENABLED: SSH 1.99 has been enabled

r3(config)#ip ssh time-out 90

r3(config)#exit

%SYS-5-CONFIG\_I: Configured from console by console

r3#show ip ssh

SSH Enabled - version 2.0

Authentication timeout: 90 secs; Authentication retries: 3

Step 2: Implement OSPF on R1 R2 R3

R1

r1>

r1>enable

r1#configure terminal

Enter configuration commands, one per line. End with CNTL/Z.

r1(config)#router ospf 101

r1(config-router)#network 192.168.1.0 0.0.0.255 area 0

r1(config-router)#network 10.1.1.0 0.0.0.3 area 0

r1(config-router)#exit

r1(config)#exit

%SYS-5-CONFIG\_I: Configured from console by console

r1#show ip ospf

Routing Process "ospf 101" with ID 192.168.1.1

Supports only single TOS(TOS0) routes

Supports opaque LSA

SPF schedule delay 5 secs, Hold time between two SPFs 10 secs

Minimum LSA interval 5 secs. Minimum LSA arrival 1 secs

Number of external LSA 0. Checksum Sum 0x000000

Number of opaque AS LSA 0. Checksum Sum 0x000000

Number of DCbitless external and opaque AS LSA 0

Number of DoNotAge external and opaque AS LSA 0

Number of areas in this router is 1. 1 normal 0 stub 0 nssa

External flood list length 0

Area BACKBONE(0)

Number of interfaces in this area is 2

Area has no authentication

SPF algorithm executed 2 times

Area ranges are

Number of LSA 1. Checksum Sum 0x003d7b

Number of opaque link LSA 0. Checksum Sum 0x000000

Number of DCbitless LSA 0

Number of indication LSA 0

Number of DoNotAge LSA 0

Flood list length 0

r1#

01:08:58: %OSPF-5-ADJCHG: Process 101, Nbr 10.2.2.2 on Serial0/0/0 from  
LOADING to

FULL, Loading Done

R2

r2>

r2>enable

r2#configure terminal

Enter configuration commands, one per line. End with CNTL/Z.

r2(config)#router ospf 101

r2(config-router)#network 10.2.2.0 0.0.0.3 area 0

r2(config-router)#network 10.1.1.0 0.0.0.3 area 0

r2(config-router)#exit

r2(config)#exit

01:08:50: %OSPF-5-ADJCHG: Process 101, Nbr 192.168.1.1 on Serial0/0/0  
from

LOADING to FULL, Loading Done

r2#

%SYS-5-CONFIG\_I: Configured from console by console

r2#show ip ospf

Routing Process "ospf 101" with ID 10.2.2.2

Supports only single TOS(TOS0) routes

Supports opaque LSA

SPF schedule delay 5 secs, Hold time between two SPFs 10 secs

Minimum LSA interval 5 secs. Minimum LSA arrival 1 secs

Number of external LSA 0. Checksum Sum 0x000000

Number of opaque AS LSA 0. Checksum Sum 0x000000

Number of DCbitless external and opaque AS LSA 0

Number of DoNotAge external and opaque AS LSA 0

Number of areas in this router is 1. 1 normal 0 stub 0 nssa

External flood list length 0

Area BACKBONE(0)

Number of interfaces in this area is 2

Area has no authentication

SPF algorithm executed 2 times

Area ranges are

Number of LSA 2. Checksum Sum 0x0134dd

Number of opaque link LSA 0. Checksum Sum 0x000000

Number of DCbitless LSA 0

Number of indication LSA 0

Number of DoNotAge LSA 0

Flood list length 0

r2#

01:09:50: %OSPF-5-ADJCHG: Process 101, Nbr 192.168.3.1 on Serial0/0/1  
from

LOADING to FULL, Loading Done

R3

r3>

r3>enable

r3#configure terminal

Enter configuration commands, one per line. End with CNTL/Z.

r3(config)#router ospf 101

r3(config-router)#network 10.2.2.0 0.0.0.3 area 0

01:09:37: %OSPF-5-ADJCHG: Process 101, Nbr 10.2.2.2 on Serial0/0/0 from  
LOADING to

FULL, Loading Done

r3(config-router)#network 192.168.3.0 0.0.0.255 area 0

r3(config-router)#exit

r3(config)#exit

%SYS-5-CONFIG\_I: Configured from console by console

r3#show ip ospf

Routing Process "ospf 101" with ID 192.168.3.1

Supports only single TOS(TOS0) routes

Supports opaque LSA

SPF schedule delay 5 secs, Hold time between two SPFs 10 secs

Minimum LSA interval 5 secs. Minimum LSA arrival 1 secs

Number of external LSA 0. Checksum Sum 0x000000

Number of opaque AS LSA 0. Checksum Sum 0x000000

Number of DCbitless external and opaque AS LSA 0

Number of DoNotAge external and opaque AS LSA 0

Number of areas in this router is 1. 1 normal 0 stub 0 nssa

External flood list length 0

Area BACKBONE(0)

Number of interfaces in this area is 2

Area has no authentication

SPF algorithm executed 3 times

Area ranges are

Number of LSA 3. Checksum Sum 0x020a7c

Number of opaque link LSA 0. Checksum Sum 0x000000

Number of DCbitless LSA 0

Number of indication LSA 0

Number of DoNotAge LSA 0

Flood list length 0

Step 3: Activate NTP protocol on server and configure R1 with NTP

r1>

r1>enable

r1#configure terminal

Enter configuration commands, one per line. End with CNTL/Z.

r1(config)#ntp server 192.168.1.50

r1(config)#ntp update-calendar

r1(config)#ntp authenticate

r1(config)#ntp trusted-key 1

r1(config)#ntp authentication-key 1 md5 NTPpa55

r1(config)#service timestamps log datetime msec

r1(config)#exit

\*Mar 01, 00:59:58.5959: SYS-5-CONFIG\_I: Configured from console by console

r1#show clock

17:21:45.949 UTC Fri Feb 10 2023

r1#

Step 4: Enable Syslog server to update logs

r1#configure terminal

Enter configuration commands, one per line. End with CNTL/Z.

```
r1(config)#logging host 192.168.1.50
```

```
r1(config)#exit
```

```
*Feb 10, 17:29:56.2929: SYS-5-CONFIG_I: Configured from console by console
```

```
*Feb 10, 17:29:56.2929: %SYS-6-LOGGINGHOST_STARTSTOP: Logging to host
```

```
192.168.1.50 port 514 started - CLI initiated
```

```
r1#show logging
```

```
Syslog logging: enabled (0 messages dropped, 0 messages rate-limited,  
0 flushes, 0 overruns, xml disabled, filtering disabled)
```

```
No Active Message Discriminator.
```

```
No Inactive Message Discriminator.
```

```
Console logging: level debugging, 10 messages logged, xml disabled,  
filtering disabled
```

```
Monitor logging: level debugging, 10 messages logged, xml disabled,  
filtering disabled
```

```
Buffer logging: disabled, xml disabled,  
filtering disabled
```

```
Logging Exception size (4096 bytes)
```

```
Count and timestamp logging messages: disabled
```

```
Persistent logging: disabled
```

```
No active filter modules.
```

```
ESM: 0 messages dropped
```

```
Trap logging: level informational, 10 message lines logged
```

```
Logging to 192.168.1.50 (udp port 514, audit disabled,  
authentication disabled, encryption disabled, link up),
```

```
2 message lines logged,
```



0 message lines rate-limited,  
0 message lines dropped-by-MD,  
xml disabled, sequence number disabled  
filtering disabled

Step 5: Enable IOS IPS

- a. r1(config)#license boot module c1900 technology-package securityk9
- b. verify the connectivity

Ping Pc-A to Pc-C

```
Cisco Packet Tracer PC Command Line 1.0
C:\>ping 192.168.3.2

Pinging 192.168.3.2 with 32 bytes of data:

Reply from 192.168.3.2: bytes=32 time=2ms TTL=125
Reply from 192.168.3.2: bytes=32 time=2ms TTL=125
Reply from 192.168.3.2: bytes=32 time=2ms TTL=125
Reply from 192.168.3.2: bytes=32 time=2ms TTL=125

Ping statistics for 192.168.3.2:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 2ms, Maximum = 2ms, Average = 2ms

C:\>
```

Ping Pc-C to Pc-A

```
Cisco Packet Tracer PC Command Line 1.0
C:\>ping 192.168.1.2

Pinging 192.168.1.2 with 32 bytes of data:

Reply from 192.168.1.2: bytes=32 time=2ms TTL=125
Reply from 192.168.1.2: bytes=32 time=2ms TTL=125
Reply from 192.168.1.2: bytes=32 time=2ms TTL=125
Reply from 192.168.1.2: bytes=32 time=2ms TTL=125

Ping statistics for 192.168.1.2:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
    Approximate round trip times in milli-seconds:
        Minimum = 2ms, Maximum = 2ms, Average = 2ms

C:\>
```

Step 6: Create an IOS IPS configuration directory in flash.

```
r1#mkdir ipsdir
```

Create directory filename [ipsdir]?

Created dir flash:ipsdir

Step 7: Reload the router

```
r1#reload
```

System configuration has been modified. Save? [yes/no]:yes

Building configuration...

[OK]

Proceed with reload? [confirm]

Step 8: Configure the IPS signature storage location.

```
r1(config)#ip ips config location flash:ipsdir
```

Step 9: Create an IPS rule.

```
r1(config)#ip ips name iosips
```

Step 10: enable logging

```
r1(config)#ip ips notify log
```

```
r1(config)#exit
```

\*Mar 01, 00:04:51.044: SYS-5-CONFIG\_I: Configured from console by console

r1#show clock

0:4:53.990 UTC Mon Mar 1 1993

r1#clock set

r1#clock set 10:20:00 10 january 2014

r1#configure terminal

Enter configuration commands, one per line. End with CNTL/Z.

r1(config)#service timestamps log datetime msec

r1(config)#logging host 192.168.1.50

Step 11: Configure IOS IPS to use the signature categories.

r1(config)#ip ips signature-category

r1(config-ips-category)#category all

r1(config-ips-category-action)#retired true

r1(config-ips-category-action)#exit

r1(config-ips-category)#category ios\_ips basic

r1(config-ips-category-action)#retired false

r1(config-ips-category-action)#exit

r1(config-ips-category)#exit

Do you want to accept these changes? [confirm]

Applying Category configuration to signatures ...

%IPS-6-ENGINE\_BUILDING: atomic-ip - 288 signatures - 6 of 13 engines

%IPS-6-ENGINE\_READY: atomic-ip - build time 30 ms - packets for this engine will be

scanned

Step 12: Apply the IPS rule to an interface.

r1(config)#interface gigabitEthernet 0/0

r1(config-if)#ip ips iosips out

Step 13: Modify the Signature

Change the event-action of signature

```
r1(config)#ip ips signature-definition
```

```
r1(config-sigdef)#signature 2004 0
```

```
r1(config-sigdef-sig)#status
```

```
r1(config-sigdef-sig-status)#retired false
```

```
r1(config-sigdef-sig-status)#enabled true
```

```
r1(config-sigdef-sig-status)#exit
```

```
r1(config-sigdef-sig)#engine
```

```
r1(config-sigdef-sig-engine)#event-action produce-alert
```

```
r1(config-sigdef-sig-engine)#event-action deny-packet-inline
```

```
r1(config-sigdef-sig-engine)#exit
```

```
r1(config-sigdef-sig)#exit
```

```
r1(config-sigdef)#exit
```

Do you want to accept these changes? [confirm]

%IPS-6-ENGINE\_BUILDS\_STARTED:

%IPS-6-ENGINE\_BUILDING: atomic-ip - 303 signatures - 3 of 13 engines

%IPS-6-ENGINE\_READY: atomic-ip - build time 480 ms - packets for this engine will be

scanned

%IPS-6-ALL\_ENGINE\_BUILDS\_COMPLETE: elapsed time 648 ms

```
r1(config)#exit
```

Step 14: Verify that IPS is working properly.

From PC-C, attempt to ping PC-A. Were the pings successful? Explain.

The pings should fail. This is because the IPS rule for event-action of an echo request

was set to “deny-packet-inline”.

```
Cisco Packet Tracer PC Command Line 1.0
C:\>ping 192.168.1.2

Pinging 192.168.1.2 with 32 bytes of data:

Reply from 192.168.1.2: bytes=32 time=2ms TTL=125
Reply from 192.168.1.2: bytes=32 time=2ms TTL=125
Reply from 192.168.1.2: bytes=32 time=2ms TTL=125
Reply from 192.168.1.2: bytes=32 time=2ms TTL=125

Ping statistics for 192.168.1.2:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
    Approximate round trip times in milli-seconds:
        Minimum = 2ms, Maximum = 2ms, Average = 2ms

C:\>ping 192.168.1.2

Pinging 192.168.1.2 with 32 bytes of data:

Request timed out.
Request timed out.
Request timed out.
Request timed out.

Ping statistics for 192.168.1.2:
    Packets: Sent = 4, Received = 0, Lost = 4 (100% loss),

C:\>
```

From PC-A, attempt to ping PC-C. Were the pings successful? Explain.

The ping should be successful. This is because the IPS rule does not cover echo reply.

When PC-A pings

PC-C, PC-C responds with an echo reply.

```
C:\>ping 192.168.3.2

Pinging 192.168.3.2 with 32 bytes of data:

Request timed out.
Reply from 192.168.3.2: bytes=32 time=2ms TTL=125
Reply from 192.168.3.2: bytes=32 time=2ms TTL=125
Reply from 192.168.3.2: bytes=32 time=2ms TTL=125

Ping statistics for 192.168.3.2:
    Packets: Sent = 4, Received = 3, Lost = 1 (25% loss),
    Approximate round trip times in milli-seconds:
        Minimum = 2ms, Maximum = 2ms, Average = 2ms

C:\>ping 192.168.3.2

Pinging 192.168.3.2 with 32 bytes of data:

Reply from 192.168.3.2: bytes=32 time=22ms TTL=125
Reply from 192.168.3.2: bytes=32 time=2ms TTL=125
Reply from 192.168.3.2: bytes=32 time=2ms TTL=125
Reply from 192.168.3.2: bytes=32 time=2ms TTL=125

Ping statistics for 192.168.3.2:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
    Approximate round trip times in milli-seconds:
        Minimum = 2ms, Maximum = 22ms, Average = 7ms
```

Step 15: View the syslog messages.

a. Click the Syslog server.

b. Select the Services tab.

c. In the left navigation menu, select SYSLOG to view the log file.

Syslog Server

Physical Config **Services** Desktop Programming Attributes

SERVICES

- HTTP
- DHCP
- DHCPv6
- TFTP
- DNS
- SYSLOG**
- AAA
- NTP
- EMAIL
- FTP
- IoT
- VM Management
- Radius EAP

Syslog

Service On Off

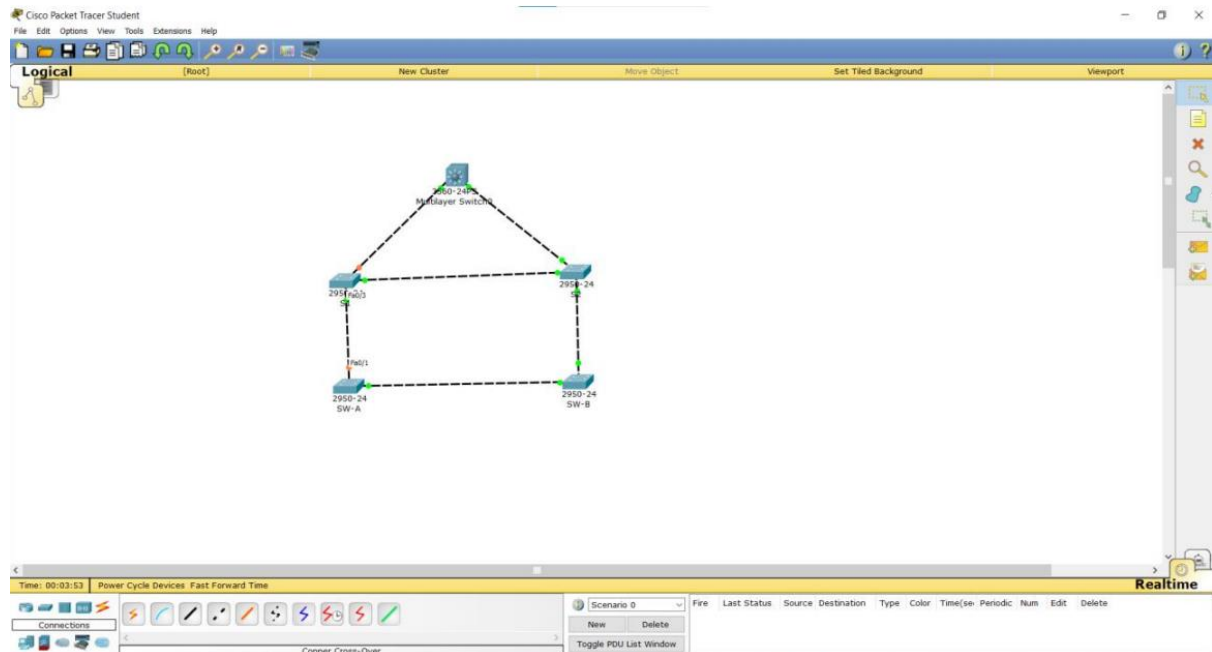
	Time	HostName	Message
1	02.10.2023 05:29:56.055 PM	192.168.1.1	%SYS-5-CONFIG_I: Configured from console by console
2	02.10.2023 05:29:56.055 PM	192.168.1.1	-%SYS-6-LOGGINGHOST_STARTSTOP: Logging to host 192.168.1.50 port 514 started - CLI initiated
3	02.10.2023 05:33:16.840 PM	192.168.1.1	-%WOS_LICENSE_IMAGE_APPLICATION-6-LICENSE_LEVEL: Module name = C1900 Next reboot level = security9 and License = security9
4	02.10.2023 05:38:19.832 PM	192.168.1.1	%SYS-5-CONFIG_I: Configured from console by console
5	02.10.2023 05:38:43.291 PM	192.168.1.1	%SYS-5-CONFIG_I: Configured from console by console
6	02.10.2023 05:38:52.799 PM	192.168.1.1	%SYS-5-CONFIG_I: Configured from console by console
7	02.10.2023 05:45:28.746 PM	192.168.1.1	%SYS-5-CONFIG_I: Configured from console by console
8	03.01.1993 12:04:51.071 AM	192.168.1.1	%SYS-5-CONFIG_I: Configured from console by console
9	01.10.2014 10:21:24.036 AM	192.168.1.1	%SYS-5-CONFIG_I: Configured from console by console
10	01.10.2014 10:22:16.139 AM	192.168.1.1	%IPS-6-ENGINE_BUILDS_STARTED: 10:22:16 UTC Jan 10 2014
11	01.10.2014 10:22:16.139 AM	192.168.1.1	%IPS-6-ENGINE_BUILDING: atomic-ip - 3 signatures - 1 of 13 engines
12	01.10.2014 10:22:16.139 AM	192.168.1.1	%IPS-6-ENGINE_READY: atomic-ip - build time 8 ms - packets for this ...
13	01.10.2014 10:22:16.139 AM	192.168.1.1	%IPS-6-ALL_ENGINE_BUILDS_COMPLETE: elapsed time 8 ms
14	01.10.2014 10:24:16.124 AM	192.168.1.1	%SYS-5-CONFIG_I: Configured from console by console
15	01.10.2014 10:27:12.869 AM	192.168.1.1	

Clear Log

## Practical-7

Aim: Layer 2 security

Topology:



Step 1: Go to Multi-Layer switch

Using the spanning-tree vlan 1 root primary command, and assign Central as the root bridge.

Switch>enable

Switch#configure terminal

Enter configuration commands, one per line. End with CNTL/Z.

Switch(config)#hostname Multilayer

Multilayer(config)#spanning-tree vlan 1 root primary

Multilayer(config)#end

%SYS-5-CONFIG\_I: Configured from console by console

Step 2: Go to switch SW-1 and Assign SW-1 as a secondary root bridge.

SW1#configure terminal

Enter configuration commands, one per line. End with CNTL/Z.

SW1(config)#spanning-tree vlan 1 root secondary

SW1(config)#interface range f0/23-24

SW1(config-if-range)#spanning-tree guard root

SW1(config-if-range)#end

%SYS-5-CONFIG\_I: Configured from console by console

Step 3: Verify the spanning-tree configuration.

Issue the show spanning-tree command to verify that Central is the root bridge.

```
Multilayer>enable
Multilayer#show spanning-tree
VLAN0001
  Spanning tree enabled protocol ieee
  Root ID    Priority    24577
             Address     0010.117A.3185
             This bridge is the root
             Hello Time 2 sec  Max Age 20 sec  Forward Delay 15 sec

  Bridge ID  Priority    24577 (priority 24576 sys-id-ext 1)
             Address     0010.117A.3185
             Hello Time 2 sec  Max Age 20 sec  Forward Delay 15 sec
             Aging Time 20

Interface Role Sts Cost Prio.Nbr Type
-----
Fa0/1 Desg FWD 19 128.1 P2p
Fa0/2 Desg FWD 19 128.2 P2p

Multilayer#
```

Step 4: Go to switch 2 and perform following commands

Switch>enable

Switch#configure terminal

Enter configuration commands, one per line. End with CNTL/Z.

Switch(config)#hostname SW2

SW2(config)#interface range f0/23-24

SW2(config-if-range)#spanning-tree guard root

SW2(config-if-range)#end

%SYS-5-CONFIG\_I: Configured from console by console

Part 2: Protect Against STP Attacks

Secure the STP parameters to prevent STP manipulation attacks.

Step 1: Enable PortFast on all access ports.

PortFast is configured on access ports that connect to a single workstation or server to enable



them to become active more quickly. On the connected access ports of the SW-A and SW-B,

use the spanning-tree portfast command.

```
SW-A(config)# interface range f0/1 - 4
```

```
SW-A(config-if-range)# spanning-tree portfast
```

```
SW-B(config)# interface range f0/1 - 4
```

```
SW-B(config-if-range)# spanning-tree portfast
```

Step 2: Enable BPDU guard on all access ports.

BPDU guard is a feature that can help prevent rogue switches and spoofing on access ports.

Enable BPDU guard on SW-A and SW-B access ports.

```
SW-A(config)# interface range f0/1 - 4
```

```
SW-A(config-if-range)# spanning-tree bpduguard enable
```

```
SW-B(config)# interface range f0/1 - 4
```

```
SW-B(config-if-range)# spanning-tree bpduguard enable
```

Step 3: Enable root guard.

Root guard can be enabled on all ports on a switch that are not root ports. It is best deployed

on ports that connect to other non-root switches. Use the show spanning-tree command to

determine the location of the root port on each switch.

On SW-1, enable root guard on ports F0/23 and F0/24. On SW-2, enable root guard on ports

F0/23 and F0/24.

```
SW1>enable
```

```
SW1#configure terminal
```

Enter configuration commands, one per line. End with CNTL/Z.

```
SW1(config)#interface range f0/23-24
```

```
SW1(config-if-range)#spanning-tree guard root
```

```
SW2>enable
```

```
SW2#configure termina
```

Enter configuration commands, one per line. End with CNTL/Z.

```
SW2(config)#interface range f0/23-24
```

```
SW2(config-if-range)#spanning-tree guard root
```

### Part 3: Configure Port Security and Disable Unused Ports

Step 1: Configure basic port security on all ports connected to host devices.

This procedure should be performed on all access ports on SW-A and SW-B. Set the

maximum number of learned MAC addresses to 2, allow the MAC address to be learned

dynamically, and set the violation to shutdown.

Note: A switch port must be configured as an access port to enable port security.

```
SWA(config)#interface range f0/1-22
```

```
SWA(config-if-range)#switchport mode access
```

```
SWA(config-if-range)#switchport port-security
```

```
SWA(config-if-range)#switchport port-security maximum 2
```

```
SWA(config-if-range)#switchport port-security violation shutdown
```

```
SWA(config-if-range)#switchport port-security mac-address sticky
```

```
SWB(config-if-range)#exit
```

```
SWB(config)#interface range f0/1-22
```

```
SWB(config-if-range)#switchport mode access
```

```
SWB(config-if-range)#switchport port-security
```

```
SWB(config-if-range)#switchport port-security maximum 2
```

```
SWB(config-if-range)#switchport port-security violation shutdown
```

```
SWB(config-if-range)#switchport port-security mac-address sticky
```

Step 2: Verify port security.

- a. On SW-A, issue the command show port-security interface f0/1 to verify that port security has been configured.

```
SWA>
SWA>enable
SWA#show port-security interface f0/1
Port Security           : Enabled
Port Status             : Secure-shutdown
Violation Mode          : Shutdown
Aging Time              : 0 mins
Aging Type              : Absolute
SecureStatic Address Aging : Disabled
Maximum MAC Addresses   : 2
Total MAC Addresses     : 0
Configured MAC Addresses : 0
Sticky MAC Addresses    : 0
Last Source Address:Vlan : 0000.0000.0000:0
Security Violation Count : 0

SWA#|
```

Step 3: Disable unused ports.

Disable all ports that are currently unused.

```
SWA>enable
```

```
SWA#configure terminal
```

Enter configuration commands, one per line. End with CNTL/Z.

```
SWA(config)#interface range f0/5-22
```

```
SWA(config-if-range)#shutdown
```

```
SWB>enable
```

```
SWB#configure terminal
```

```
SWB(config)#interface range f0/5-22
```

```
SWB(config-if-range)#shutdown
```



```
Switch(config)#hostname Switch1
```

```
Switch1(config)#vlan 10
```

```
Switch1(config-vlan)#name areaIToffice
```

```
Switch1(config-vlan)#exit
```

```
Switch1(config)#vlan 20
```

```
Switch1(config-vlan)#name areaITprod
```

```
Switch1(config-vlan)#exit
```

```
Switch1(config)#vlan 200
```

```
Switch1(config-vlan)#name areaITmgmt
```

```
Switch1(config-vlan)#exit
```

```
Switch1(config)#exit
```

```
Switch1#
```

```
%SYS-5-CONFIG_I: Configured from console by console
```

```
Switch1#show vlan
```

VLAN	Name	Status	Ports
1	default	active	Fa0/1, Fa0/2, Fa0/3, Fa0/4 Fa0/5, Fa0/6, Fa0/7, Fa0/8 Fa0/9, Fa0/10, Fa0/11, Fa0/12 Fa0/13, Fa0/14, Fa0/15, Fa0/16 Fa0/17, Fa0/18, Fa0/19, Fa0/20 Fa0/21, Fa0/22, Fa0/23, Fa0/24 Gig0/1, Gig0/2
10	areaIToffice	active	
20	areaITprod	active	
200	areaITmgmt	active	
1002	fddi-default	active	
1003	token-ring-default	active	
1004	fddinet-default	active	
1005	trnet-default	active	

VLAN	Type	SAID	MTU	Parent	RingNo	BridgeNo	Stp	BrdgMode	Trans1	Trans2
1	enet	100001	1500	-	-	-	-	-	0	0
10	enet	100010	1500	-	-	-	-	-	0	0
20	enet	100020	1500	-	-	-	-	-	0	0

Step 2:

Switch1>

Switch1>enable

Switch1#configure terminal

Enter configuration commands, one per line. End with CNTL/Z.

Switch1(config)#interface fastEthernet 0/2

Switch1(config-if)#switchport access vlan 30

% Access VLAN does not exist. Creating vlan 30

Switch1(config-if)#switchport mode access

Switch1(config-if)#exit

Switch1(config)#interface fastEthernet 0/1

Switch1(config-if)#switchport trunk native vlan 30

Switch1(config-if)#switchport mode trunk

Switch1(config-if)#

%LINEPROTO-5-UPDOWN: Line protocol on Interface FastEthernet0/1,  
changed state

to down

%LINEPROTO-5-UPDOWN: Line protocol on Interface FastEthernet0/1,  
changed state

to up

Switch1(config-if)#switchport nonegotiate

Step 3:

Router>

Router>enable

Router#configure terminal

Enter configuration commands, one per line. End with CNTL/Z.

Router(config)#hostname Router1

Router1(config)#int g0/0.10

```
Router1(config-subif)#
```

```
%LINK-5-CHANGED: Interface GigabitEthernet0/0.10, changed state to up
```

```
%LINEPROTO-5-UPDOWN: Line protocol on Interface GigabitEthernet0/0.10,  
changed
```

```
state to up
```

```
Router1(config-subif)#ip address 192.168.200.10 255.255.255.0
```

```
% Configuring IP routing on a LAN subinterface is only allowed if that  
subinterface is already configured as part of an IEEE 802.10, IEEE 802.1Q,  
or ISL vLAN.
```

```
Router1(config-subif)#exit
```

```
Router1(config)#int g0/0.20
```

```
Router1(config-subif)#
```

```
%LINK-5-CHANGED: Interface GigabitEthernet0/0.20, changed state to up
```

```
%LINEPROTO-5-UPDOWN: Line protocol on Interface GigabitEthernet0/0.20,  
changed
```

```
state to up
```

```
Router1(config-subif)#ip address 192.168.200.20 255.255.255.0
```

```
% Configuring IP routing on a LAN subinterface is only allowed if that  
subinterface is already configured as part of an IEEE 802.10, IEEE 802.1Q,  
or ISL vLAN.
```

```
Router1(config-subif)#exit
```

```
Router1(config)#int g0/0.200
```

```
Router1(config-subif)#
```

```
%LINK-5-CHANGED: Interface GigabitEthernet0/0.200, changed state to up
```

```
%LINEPROTO-5-UPDOWN:      Line      protocol      on      Interface  
GigabitEthernet0/0.200, changed
```

```
state to up
```

```
Router1(config-subif)#ip address 192.168.200.30 255.255.255.0
```

% Configuring IP routing on a LAN subinterface is only allowed if that subinterface is already configured as part of an IEEE 802.10, IEEE 802.1Q, or ISL vLAN.

```
Router1(config-subif)#exit
```

```
Router1(config)#int g0/0.200
```

```
Router1(config-subif)#description areaITmgmt
```

```
Router1(config-subif)#ip address 192.168.200.1 255.255.255.0
```

% Configuring IP routing on a LAN subinterface is only allowed if that subinterface is already configured as part of an IEEE 802.10, IEEE 802.1Q, or ISL vLAN.

```
Router1(config-subif)#exit
```

```
Router1(config)#access-list 101 deny ip 192.168.200.0 0.0.0.255 any
```

```
Router1(config)#access-list 101 permit ip any any
```

```
Router1(config)#access-list 102 permit ip host 192.168.200.10 any
```

```
Router1(config)#int g0/0.10
```

```
Router1(config-subif)#no ip address
```

% Configuring IP routing on a LAN subinterface is only allowed if that subinterface is already configured as part of an IEEE 802.10, IEEE 802.1Q, or ISL vLAN.

```
Router1(config-subif)#ip access-group 101 in
```

```
Router1(config-subif)#description areaIToffice
```

```
Router1(config-subif)#no ip address
```

% Configuring IP routing on a LAN subinterface is only allowed if that subinterface is already configured as part of an IEEE 802.10, IEEE 802.1Q, or ISL vLAN.

```
Router1(config-subif)#ip access-group 101 in
```

```
Router1(config-subif)#int g0/0.20
```



```
Router1(config-subif)#description areaITprod
```

```
Router1(config-subif)#no ip address
```

% Configuring IP routing on a LAN subinterface is only allowed if that subinterface is already configured as part of an IEEE 802.10, IEEE 802.1Q, or ISL vLAN.

```
Router1(config-subif)#ip access-group 101 in
```

```
Router1(config-subif)#int g0/0.200
```

```
Router1(config-subif)#description areaITmgmt
```

```
Router1(config-subif)#encapsulation dot1Q 200
```

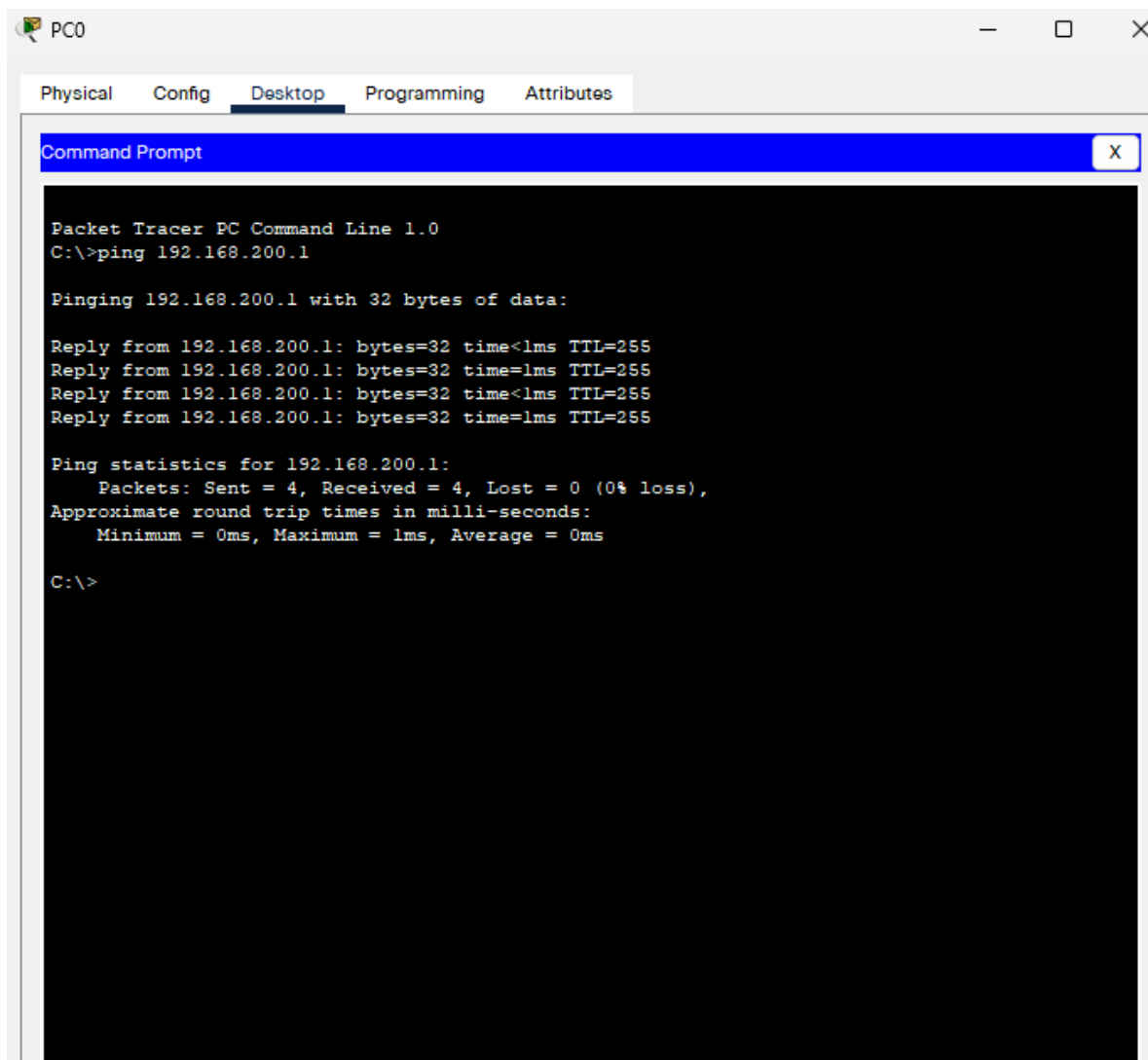
```
Router1(config-subif)#ip address 192.168.200.1 255.255.255.0
```

% 192.168.200.0 overlaps with GigabitEthernet0/0

Step 4:

Verifying the configuration

- Pc0 to Router1 – ping should be successful



The screenshot shows the Packet Tracer interface for PC0. The 'Desktop' tab is selected, displaying a 'Command Prompt' window. The command prompt shows the execution of a ping command to 192.168.200.1, which is successful. The output includes the number of bytes, time, and TTL for each of the four replies, as well as the overall ping statistics showing 0% loss.

```
Packet Tracer PC Command Line 1.0
C:\>ping 192.168.200.1

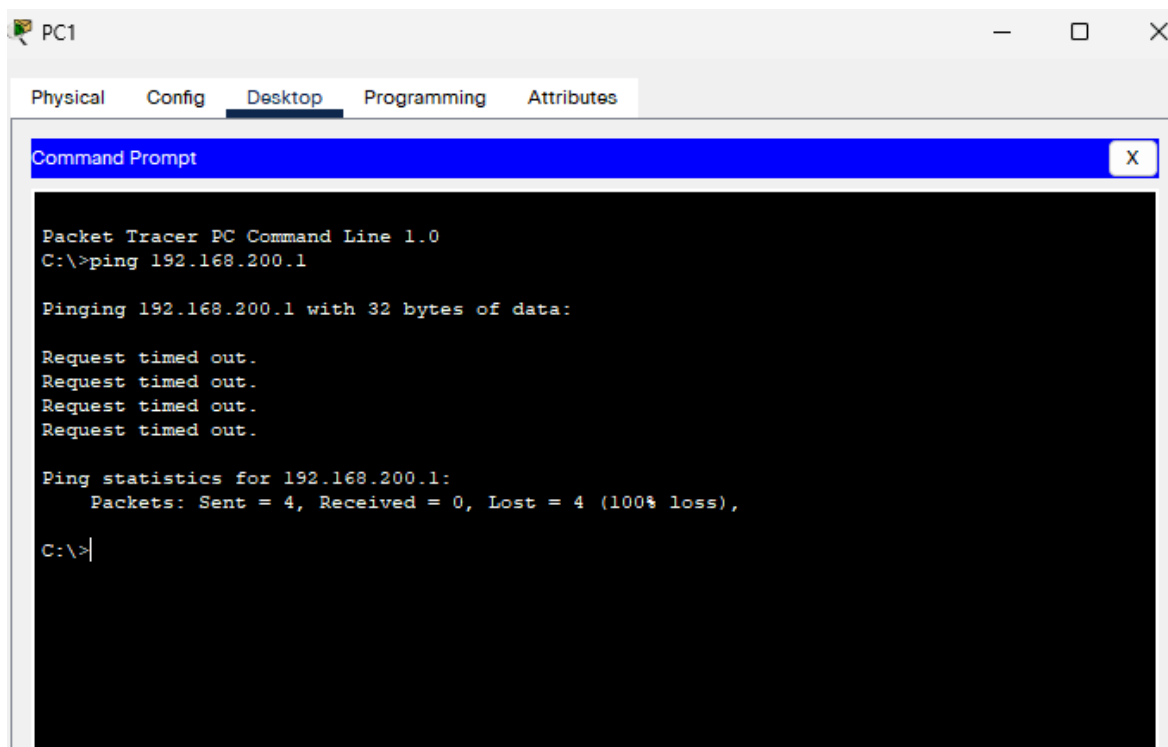
Pinging 192.168.200.1 with 32 bytes of data:

Reply from 192.168.200.1: bytes=32 time<1ms TTL=255
Reply from 192.168.200.1: bytes=32 time=1ms TTL=255
Reply from 192.168.200.1: bytes=32 time<1ms TTL=255
Reply from 192.168.200.1: bytes=32 time=1ms TTL=255

Ping statistics for 192.168.200.1:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
    Approximate round trip times in milli-seconds:
        Minimum = 0ms, Maximum = 1ms, Average = 0ms

C:\>
```

- PC1 to Router1 – should be unsuccessful i.e., blocked



The screenshot shows a Packet Tracer PC window for PC1. The 'Desktop' tab is active, displaying a Command Prompt. The command prompt shows the execution of a ping command to 192.168.200.1. The output indicates that all four requests timed out, resulting in a 100% loss of packets.

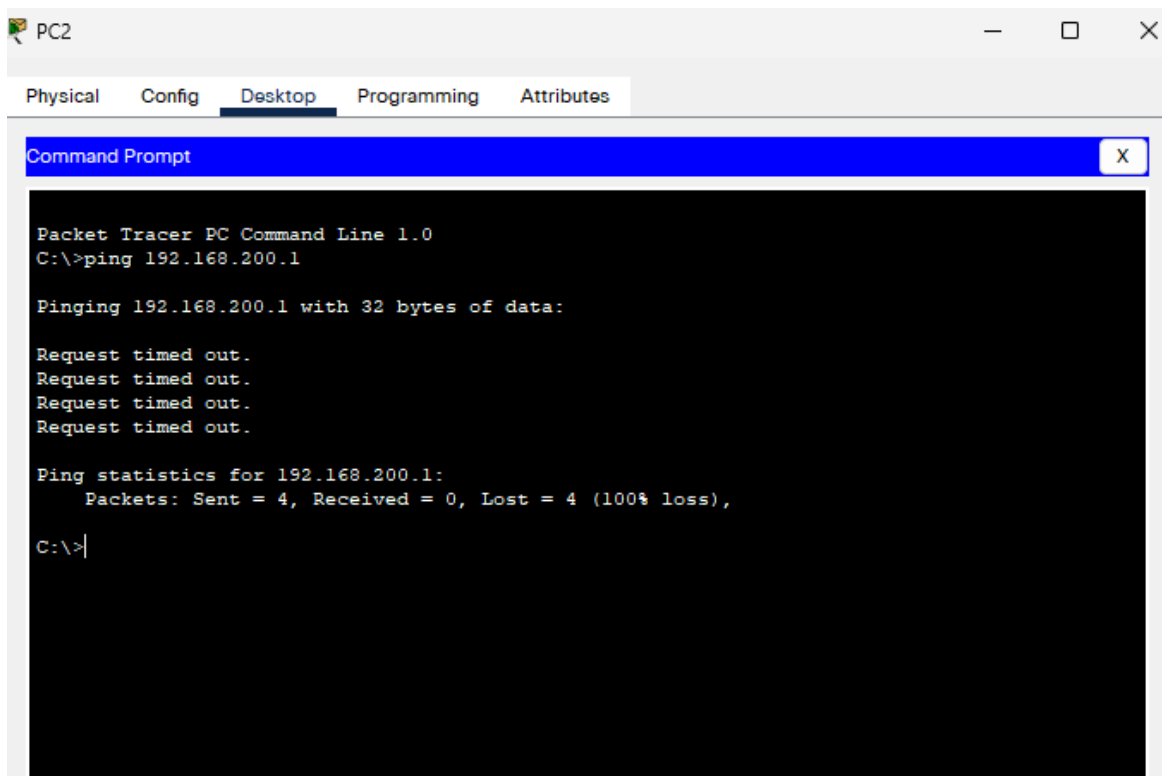
```
Packet Tracer PC Command Line 1.0
C:\>ping 192.168.200.1

Pinging 192.168.200.1 with 32 bytes of data:

Request timed out.
Request timed out.
Request timed out.
Request timed out.

Ping statistics for 192.168.200.1:
    Packets: Sent = 4, Received = 0, Lost = 4 (100% loss),
C:\>|
```

- Pc2 to Router1



The screenshot shows a Packet Tracer PC window for PC2. The 'Desktop' tab is active, displaying a Command Prompt. The command prompt shows the execution of a ping command to 192.168.200.1. The output indicates that all four requests timed out, resulting in a 100% loss of packets.

```
Packet Tracer PC Command Line 1.0
C:\>ping 192.168.200.1

Pinging 192.168.200.1 with 32 bytes of data:

Request timed out.
Request timed out.
Request timed out.
Request timed out.

Ping statistics for 192.168.200.1:
    Packets: Sent = 4, Received = 0, Lost = 4 (100% loss),
C:\>|
```