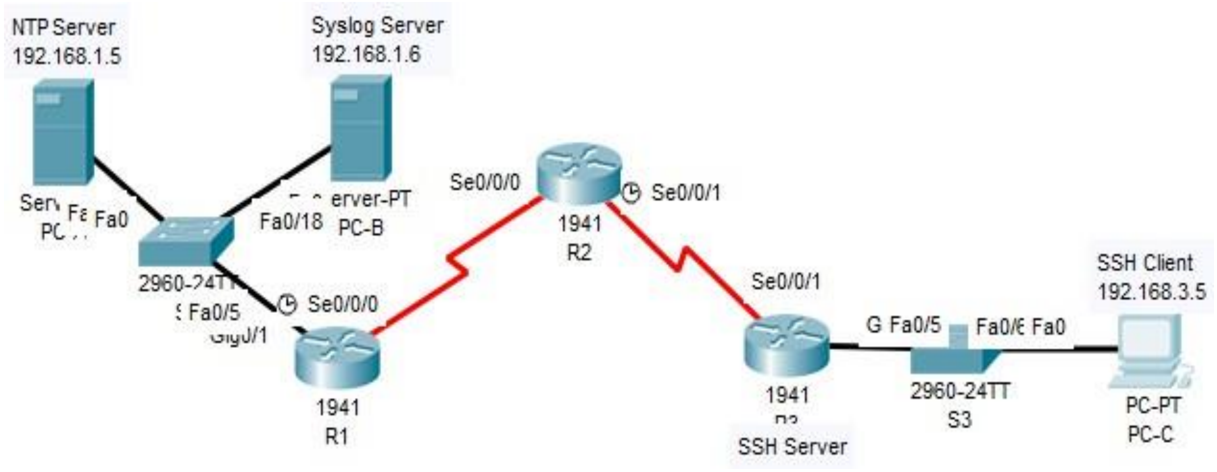
Practical No. 1

Aim: Configure Cisco Routers for OSPF MD5 authentication, SSH, NTP and Syslog.



Enable password: **ciscoenpa55**

A. Configure OSPF MD5 Authentication

Step 1: Configure OSPF MD5 authentication for all the routers in area 0

```
R1>en
Password:
R1#conf t
Enter configuration commands, one per line. End with CNTL/Z.
R1(config)#router ospf 1
R1(config-router)#area 0 authentication message-digest
R1(config-router)#

R2>en
Password:
R2#conf t
Enter configuration commands, one per line. End with CNTL/Z.
R2(config)#router ospf 1
R2(config-router)#area 0 authentication message-digest
R2(config-router)#

R3>en
Password:
R3#conf t
Enter configuration commands, one per line. End with CNTL/Z.
R3(config)#router ospf 1
R3(config-router)#area 0 authentication message-digest
R3(config-router)#
```

Step 2: Configure the MD5 key for all the routers in area 0.

```
R1(config)#int s0/0/0
R1(config-if)#ip ospf message-digest-key 1 md5 MD5pa55
OSPF: Key 1 already exists
R1(config-if)#

R2(config)#int s0/0/0
R2(config-if)#ip ospf message-digest-key 1 md5 MD5pa55
R2(config-if)#int s0/0/1
R2(config-if)#ip ospf message-digest-key 1 md5 MD5pa55
R2(config-if)#

R3(config)#int s0/0/1
R3(config-if)#ip ospf message-digest-key 1 md5 MD5pa55
R3(config-if)#
00:45:50: %OSPF-5-ADJCHG: Process 1, Nbr 10.2.2.2 on Serial0/0/1 from LOADING to FULL, Loading Done
```

Step 4: Verify configurations.

```
R1#sh ip ospf int (Take snapshot)
R2#sh ip ospf int (Take snapshot)
```

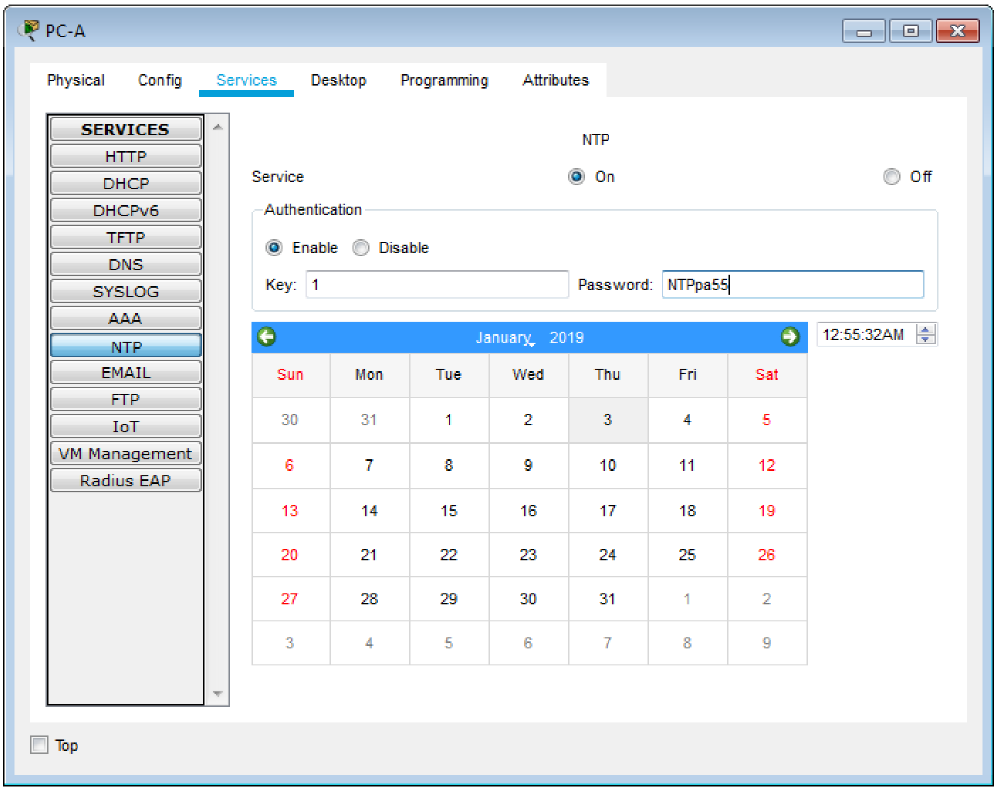
R3#sh ip ospf int (Take snapshot)

B. Configure NTP

Step 1: Enable NTP authentication on PC-A.

a. On **PC-A**, click **NTP** under the Services tab to verify NTP service is enabled.

b. To configure NTP authentication, click **Enable** under Authentication. Use key **1** and password **NTPpa55** for authentication.



NTP

Service ● On ○ Off

Authentication

● Enable ○ Disable

Key: 1 Password: NTPpa55

January 2019 12:55:32AM

Sun	Mon	Tue	Wed	Thu	Fri	Sat
30	31	1	2	3	4	5
6	7	8	9	10	11	12
13	14	15	16	17	18	19
20	21	22	23	24	25	26
27	28	29	30	31	1	2
3	4	5	6	7	8	9

Step 2: Configure R1, R2, and R3 as NTP clients.

```
R1#conf t
R1(config)#ntp server 192.168.1.5
R1(config)#
```

```
R2#conf t
R2(config)#ntp server 192.168.1.5
R2(config)#
```

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

Verify clients

R1#sh ntp status (Take snapshot)
R2#sh ntp status (Take snapshot)
R3#sh ntp status (Take snapshot)

Step 3: Configure routers to update hardware clock.

```
R1#conf t
R1(config)#ntp update-calendar
R1(config)#^Z
R1#
```

Verify hardware clock is updated

```
R1#sh clock
1:1:33.516 UTC Thu Jan 3 2019
R1#
```

```
R2#conf t
R2(config)#ntp update-calendar
R2(config)#^Z
R2#
```

Verify hardware clock is updated

```
R2#sh clock
1:2:59.396 UTC Thu Jan 3 2019
R2#
```

```
R3#conf t
R3(config)#ntp update-calendar
R3(config)#^Z
R3#
```

Verify hardware clock is updated

```
R3#sh clock
1:3:45.914 UTC Thu Jan 3 2019
R3#
```

Step 4: Configure NTP authentication on the routers.

```
R1#conf t
R1(config)#ntp authenticate
R1(config)#ntp trusted-key 1
R1(config)#ntp authentication-key 1 md5 NTPpa55
R1(config)#
```

```
R2#conf t
R2(config)#ntp authenticate
R2(config)#ntp trusted-key 1
R2(config)#ntp authentication-key 1 md5 NTPpa55
R2(config)#
```

```
R3#conf t
R3(config)#ntp authenticate
R3(config)#ntp trusted-key 1
R3(config)#ntp authentication-key 1 md5 NTPpa55
R3(config)#
```

Step 5: Configure routers to timestamp log messages.

```
R1(config)#service timestamps log datetime msec
R2(config)#service timestamps log datetime msec
R3(config)#service timestamps log datetime msec
```

C. Configure syslog server

Step 1: Configure the routers to identify the remote host (Syslog Server) that will receive logging messages.

```
R1#conf t
Enter configuration commands, one per line. End with CNTL/Z.
R1(config)#logging host 192.168.1.6
R1(config)#^Z
R1#
*Jan 03, 01:18:42.1818: %SYS-5-CONFIG_I: Configured from console by console
*Jan 03, 01:18:42.1818: *Jan 03, 01:18:42.1818: %SYS-6-LOGGINGHOST_STARTSTOP:
Logging to host 192.168.1.6 port 514 started - CLI initiated
R1#
```

```
R2#conf t
Enter configuration commands, one per line. End with CNTL/Z.
R2(config)#logging host 192.168.1.6
R2(config)#^Z
R2#
*Jan 03, 01:19:21.1919: %SYS-5-CONFIG_I: Configured from console by console
*Jan 03, 01:19:21.1919: *Jan 03, 01:19:21.1919: %SYS-6-LOGGINGHOST_STARTSTOP:
Logging to host 192.168.1.6 port 514 started - CLI initiated
R2#
```

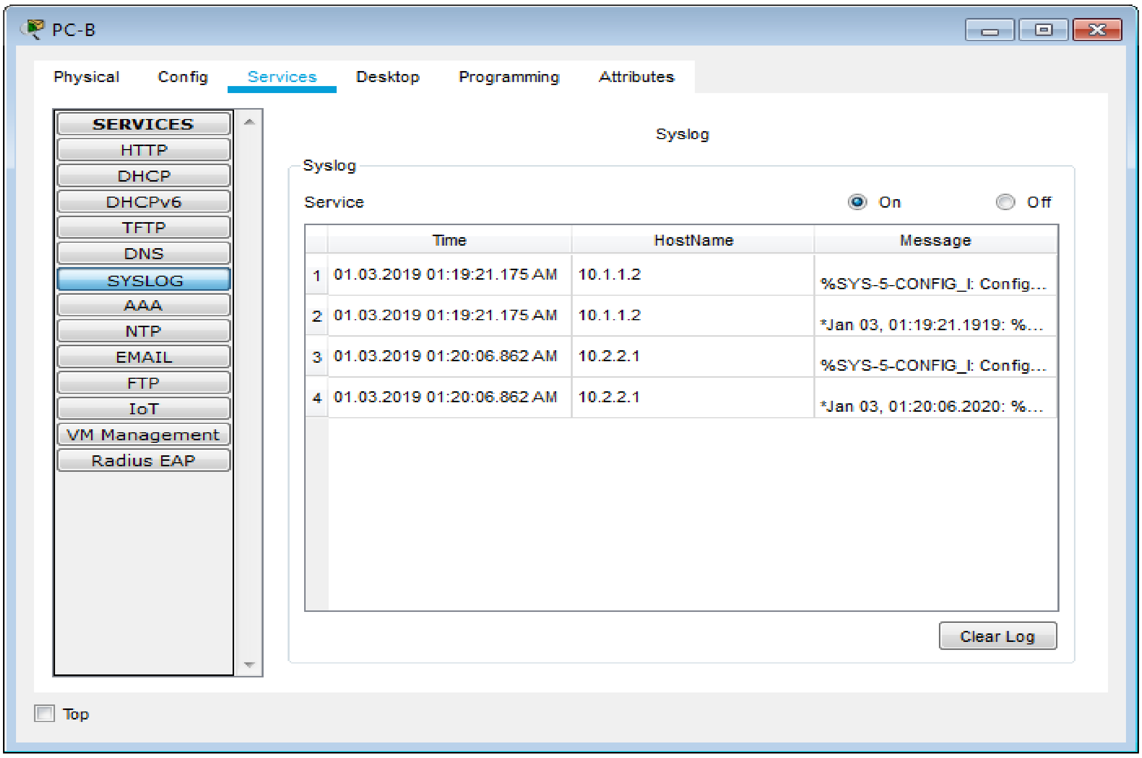
```
R3#conf t
Enter configuration commands, one per line. End with CNTL/Z.
R3(config)#logging host 192.168.1.6
R3(config)#^Z
R3#
*Jan 03, 01:20:06.2020: %SYS-5-CONFIG_I: Configured from console by console
*Jan 03, 01:20:06.2020: *Jan 03, 01:20:06.2020: %SYS-6-LOGGINGHOST_STARTSTOP:
Logging to host 192.168.1.6 port 514 started - CLI initiated
R3#
```

Step 2: Verify logging configuration.

```
R1#sh logging (Take snapshot)
R2#sh logging (Take snapshot)
R2#sh logging (Take snapshot)
```

Step 3: Examine logs of the Syslog Server.

From the **Services** tab of the **Syslog Server**'s dialogue box, select the **Syslog** services button. Observe the logging messages received from the routers.



Syslog

Syslog

Service ● On ○ Off

	Time	HostName	Message
1	01.03.2019 01:19:21.175 AM	10.1.1.2	%SYS-5-CONFIG_I: Config...
2	01.03.2019 01:19:21.175 AM	10.1.1.2	*Jan 03, 01:19:21.1919: %...
3	01.03.2019 01:20:06.862 AM	10.2.2.1	%SYS-5-CONFIG_I: Config...
4	01.03.2019 01:20:06.862 AM	10.2.2.1	*Jan 03, 01:20:06.2020: %...

Practical No. 2

Aim: Configure AAA Authentication on Cisco Routers.

Background / Scenario

The network topology shows routers R1, R2 and R3. Currently, all administrative security is based on knowledge of the enable secret password. Your task is to configure and test local and server-based AAA solutions.

You will create a local user account and configure local AAA on router R1 to test the console and vty logins.

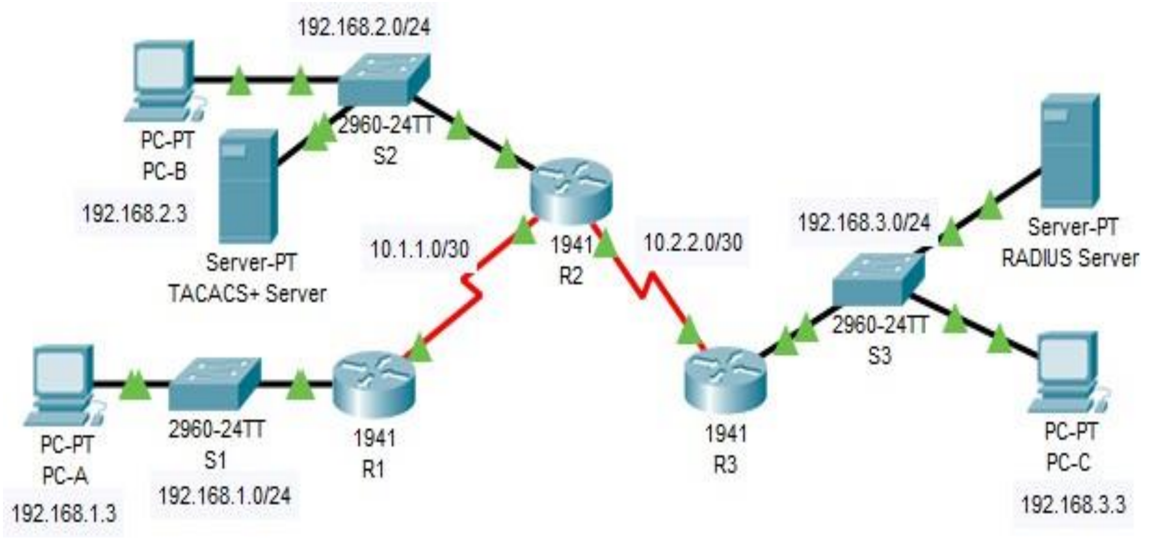
- User account: **Admin1** and password **admin1pa55**

The routers have also been pre-configured with the following:

- o Enable secret password: **ciscoenpa55**
- o OSPF routing protocol with MD5 authentication using password: **MD5pa55**

Note: The console and vty lines have not been pre-configured.

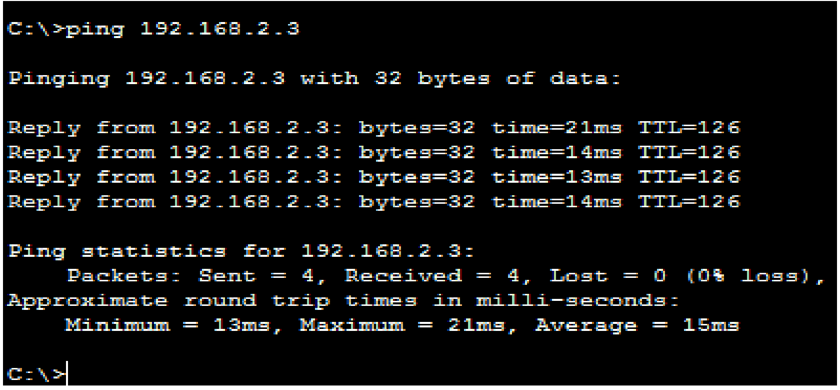
Note: IOS version 15.3 uses SCRYPT as a secure encryption hashing algorithm; however, the IOS version that is currently supported in Packet Tracer uses MD5. Always use the most secure option available on your equipment.



A. Configure Local AAA Authentication for Console Access on R1

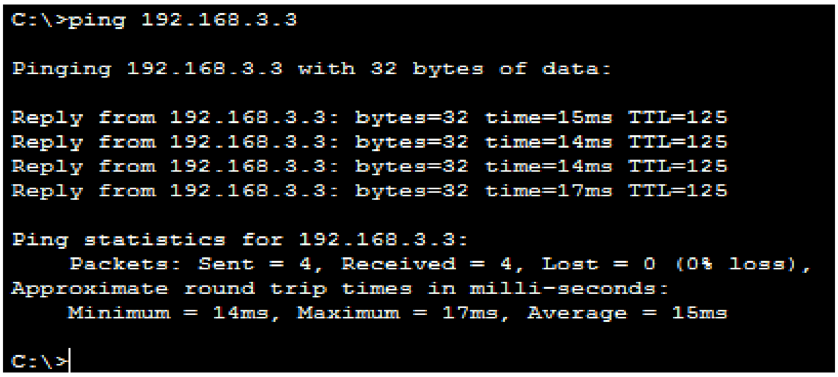
Step 1: Test connectivity.

- Ping from PC-A to PC-B.



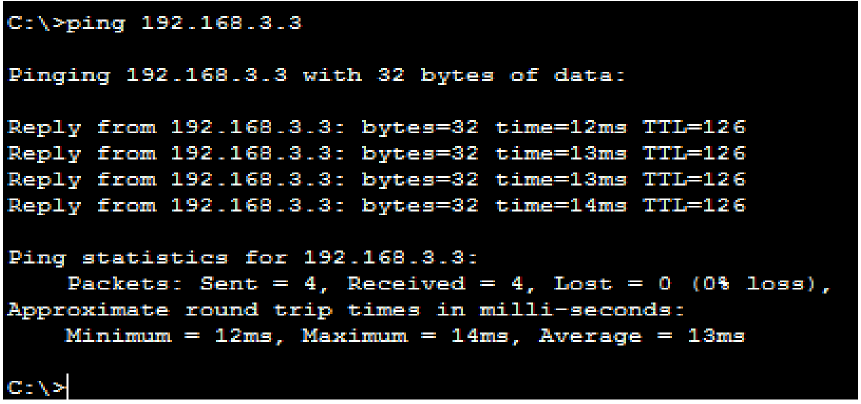
```
C:\>ping 192.168.2.3

Pinging 192.168.2.3 with 32 bytes of data:

Reply from 192.168.2.3: bytes=32 time=21ms TTL=126
Reply from 192.168.2.3: bytes=32 time=14ms TTL=126
Reply from 192.168.2.3: bytes=32 time=13ms TTL=126
Reply from 192.168.2.3: bytes=32 time=14ms TTL=126

Ping statistics for 192.168.2.3:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 13ms, Maximum = 21ms, Average = 15ms

C:\>
```

- Ping from PC-A to PC-C.



```
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=15ms TTL=125
Reply from 192.168.3.3: bytes=32 time=14ms TTL=125
Reply from 192.168.3.3: bytes=32 time=14ms TTL=125
Reply from 192.168.3.3: bytes=32 time=17ms 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 = 14ms, Maximum = 17ms, Average = 15ms

C:\>
```

- Ping from PC-B to PC-C.



```
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=12ms TTL=126
Reply from 192.168.3.3: bytes=32 time=13ms TTL=126
Reply from 192.168.3.3: bytes=32 time=13ms TTL=126
Reply from 192.168.3.3: bytes=32 time=14ms TTL=126

Ping statistics for 192.168.3.3:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 12ms, Maximum = 14ms, Average = 13ms

C:\>
```

Step 2: Configure a local username on R1.

```
R1>en
Password:
R1#conf t
R1(config)#username Admin1 secret admin1pa55
R1(config)#
```

Step 3: Configure local AAA authentication for console access on R1.

```
R1(config)#aaa new-model
R1(config)#aaa authentication login default local
```

Step 4: Configure the line console to use the defined AAA authentication method.

```
R1#conf t
R1(config)#line console 0
R1(config-line)#login authentication default
```

Step 5: Verify the AAA authentication method.

```
R1(config-line)#end
R1#
%SYS-5-CONFIG_I: Configured from console by console
R1#exit

R1 con0 is now available
Press RETURN to get started.

************ AUTHORIZED ACCESS ONLY *************
UNAUTHORIZED ACCESS TO THIS DEVICE IS PROHIBITED.

User Access Verification

Username: Admin1
Password:
R1>
```

B. Configure Local AAA Authentication for vty Lines on R1

Step 1: Configure domain name and crypto key for use with SSH.

a. Use ccnasecurity.com as the domain name on R1.

```
R1>en
Password:
R1#conf t
R1(config)#ip domain-name ccnasecurity.com
```

b. Create an RSA crypto key using 1024 bits.

```
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]
```

Step 2: Configure a named list AAA authentication method for the vty lines on R1.

```
R1(config)#aaa authentication login SSH-LOGIN local
*Mar 1 0:24:50.230: %SSH-5-ENABLED: SSH 1.99 has been enabled
```

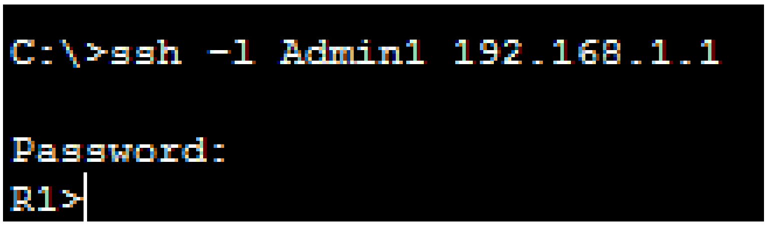
Step 3: Configure the vty lines to use the defined AAA authentication method.

Configure the vty lines to use the named AAA method and only allow SSH for remote access.

```
R1(config)#line vty 0 4
R1(config-line)#login authentication SSH-LOGIN
R1(config-line)#transport input ssh
R1(config-line)#end
```

R1#
%SYS-5-CONFIG_I: Configured from console by console

Step 4: Verify the AAA authentication method.
Verify the SSH configuration SSH to **R1** from the command prompt of **PC-A**..



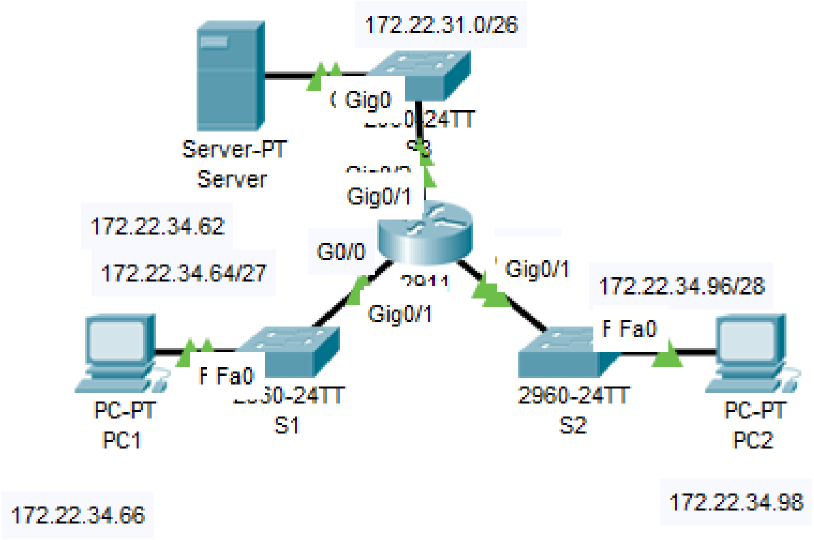
```
C:\>ssh -l Admin1 192.168.1.1

Password:
R1>
```

Practical No. 3

Aim: Configure Extended ACL's on Cisco Routers.

Background / Scenario

Two employees need access to services provided by the server. **PC1** needs only FTP access while **PC2** needs only web access. Both computers are able to ping the server, but not each other.



A. Configure, Apply and Verify an Extended Numbered ACL

Step 1: Configure an ACL to permit FTP and ICMP.

a. The statement would permit all TCP traffic. But we are only permitting FTP traffic; therefore, enter the **eq** keyword

```
R1>en
R1#conf t
R1(config)#access-list 100 permit tcp 172.22.34.64 0.0.0.31 host 172.22.34.62 eq ftp
R1(config)#
%SYS-5-CONFIG_I: Configured from console by console
```

b. Create a second access list statement to permit ICMP (ping, etc.) traffic from **PC1** to **Server**. Note that the access list number remains the same and no particular type of ICMP traffic needs to be specified.

```
R1#conf t
Enter configuration commands, one per line. End with CNTL/Z.
R1(config)#access-list 100 permit icmp 172.22.34.64 0.0.0.31 host 172.22.34.62
```

c. All other traffic is denied, by default.

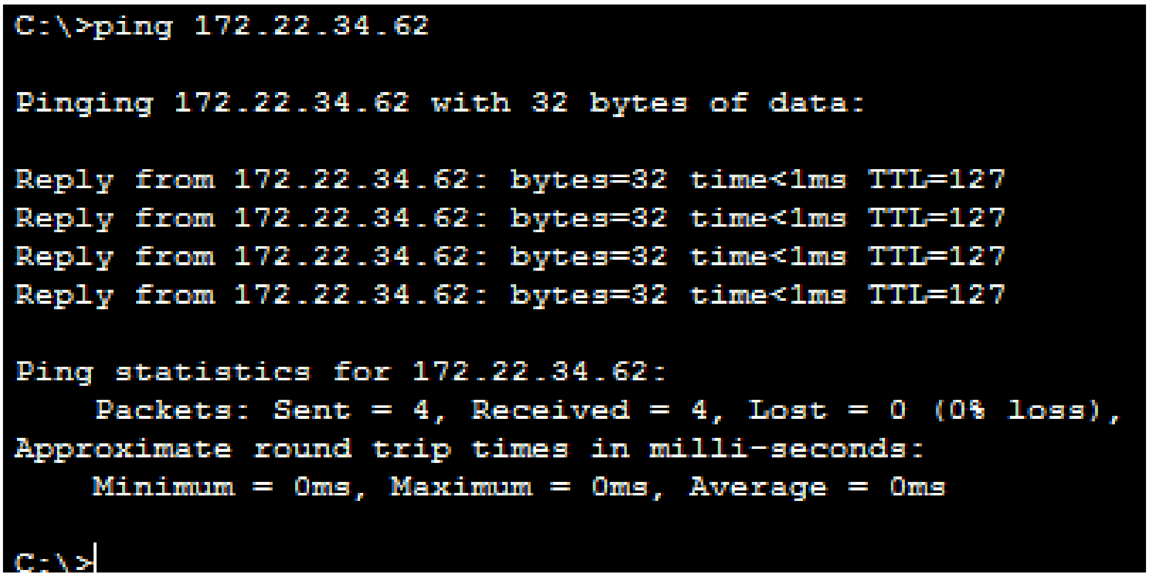
Step 2: Apply the ACL on the correct interface to filter traffic.

From **R1**'s perspective, the traffic that ACL 100 applies to is inbound from the network connected to Gigabit Ethernet 0/0 interface. Enter interface configuration mode and apply the ACL.

```
R1(config)#int gigabitEthernet 0/0
R1(config-if)#ip access-group 100 in
```

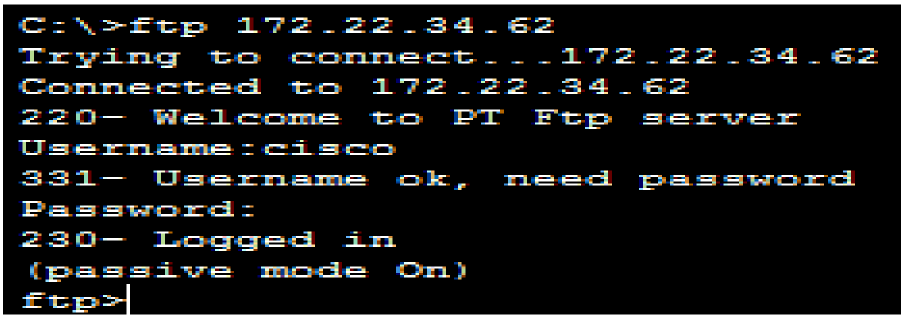
Step 3: Verify the ACL implementation.

a. Ping from **PC1** to **Server**. If the pings are unsuccessful, verify the IP addresses before continuing.



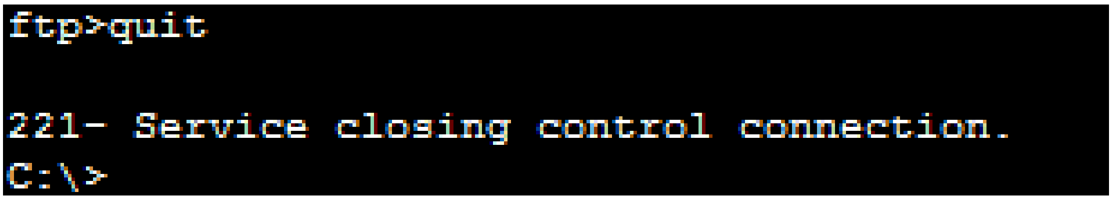
```
C:\>ping 172.22.34.62

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<1ms TTL=127
Reply from 172.22.34.62: bytes=32 time<1ms TTL=127
Reply from 172.22.34.62: bytes=32 time<1ms 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 = 0ms, Average = 0ms

C:\>
```

b. FTP from **PC1** to **Server**. The username and password are both **cisco**.



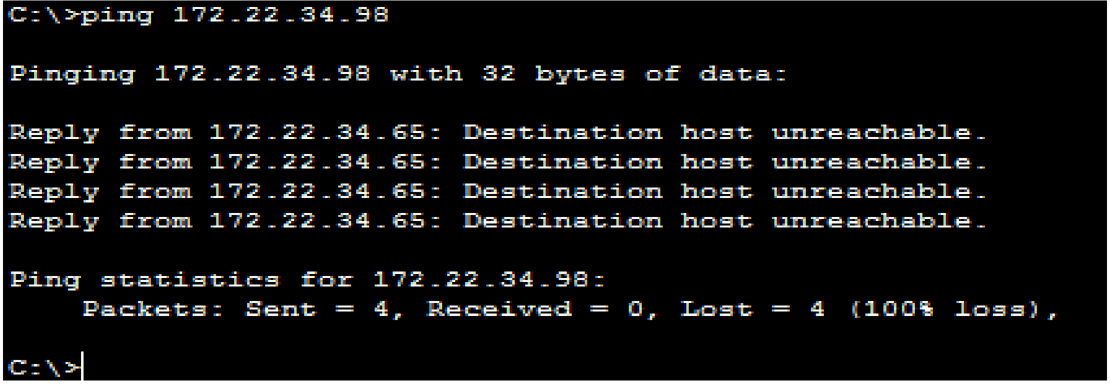
```
C:\>ftp 172.22.34.62
Trying to connect...172.22.34.62
Connected to 172.22.34.62
220- Welcome to PT Ftp server
Username:cisco
331- Username ok, need password
Password:
230- Logged in
(passive mode On)
ftp>
```

c. Exit the FTP service of the **Server**.



```
ftp>quit

221- Service closing control connection.
C:\>
```

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



```
C:\>ping 172.22.34.98

Pinging 172.22.34.98 with 32 bytes of data:

Reply from 172.22.34.65: Destination host unreachable.
Reply from 172.22.34.65: Destination host unreachable.
Reply from 172.22.34.65: Destination host unreachable.
Reply from 172.22.34.65: Destination host unreachable.

Ping statistics for 172.22.34.98:
    Packets: Sent = 4, Received = 0, Lost = 4 (100% loss),

C:\>
```

Practical No. 4

Aim: Configure IP ACL’s to Mitigate Attack.

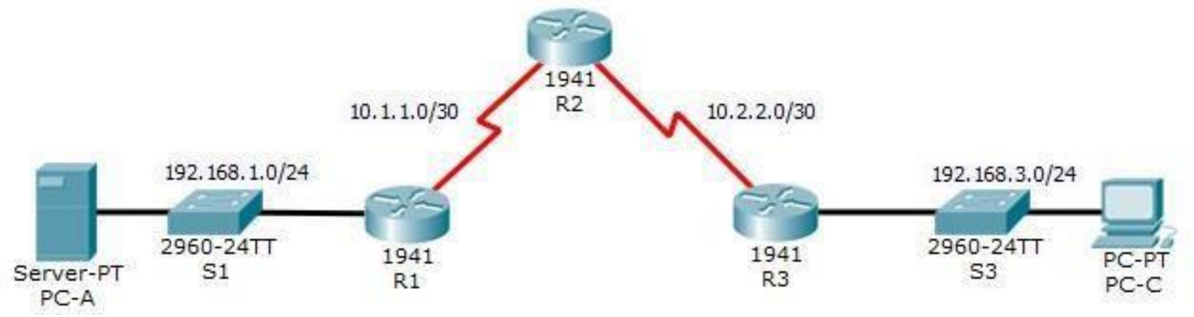
Background/Scenario

Access to routers R1, R2, and R3 should only be permitted from PC-C, the management station. PC-C is also used for connectivity testing to PC-A, which is a server providing DNS, SMTP, FTP, and HTTPS services.

Standard operating procedure is to apply ACLs on edge routers to mitigate common threats based on source and destination IP address. In this activity, you will create ACLs on edge routers R1 and R3 to achieve this goal. You will then verify ACL functionality from internal and external hosts.

The routers have been pre-configured with the following:

- o Enable password: **ciscoenpa55** o Password for console: **ciscoconpa55**
- o SSH logon username and password: **SSHadmin/ciscosshpa55**
- o IP addressing o Static routing

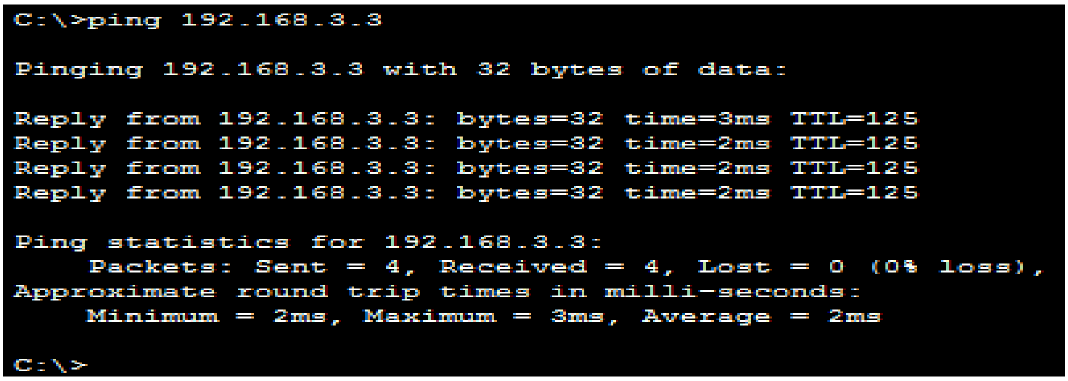


A. Verify Basic Network Connectivity

Verify network connectivity prior to configuring the IP ACLs.

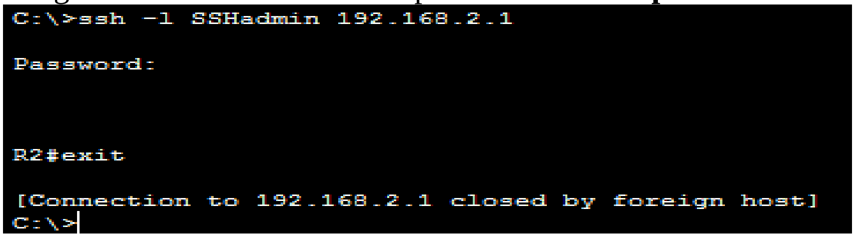
Step 1: From PC-A, verify connectivity to PC-C and R2.

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



```
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:\>
```

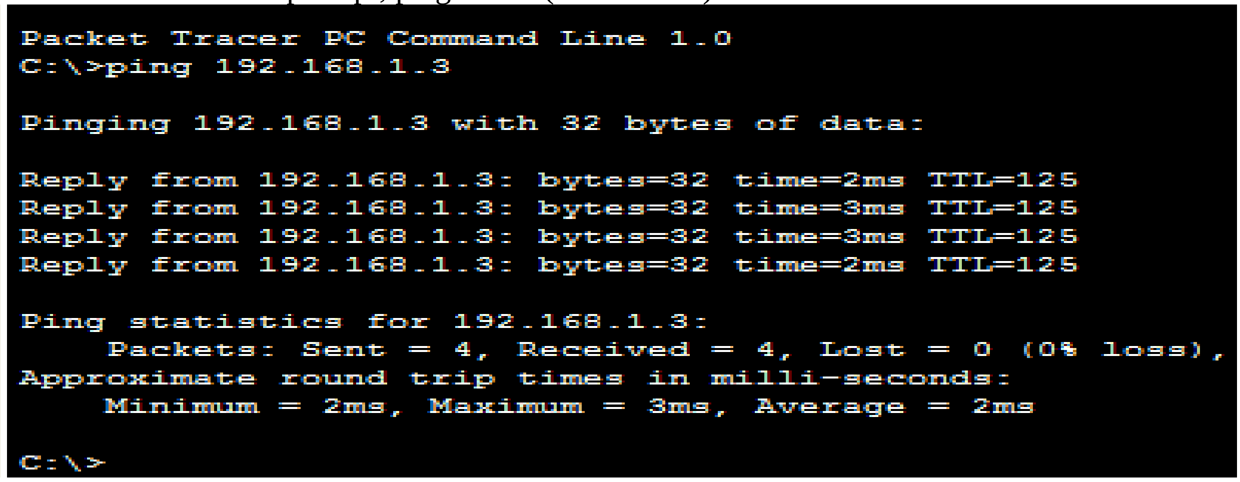
b. From the command prompt, establish an SSH session to **R2** Lo0 interface (192.168.2.1) using username **SSHadmin** and password **ciscosshpa55**. When finished, exit the SSH session.



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

Password:


R2#exit

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

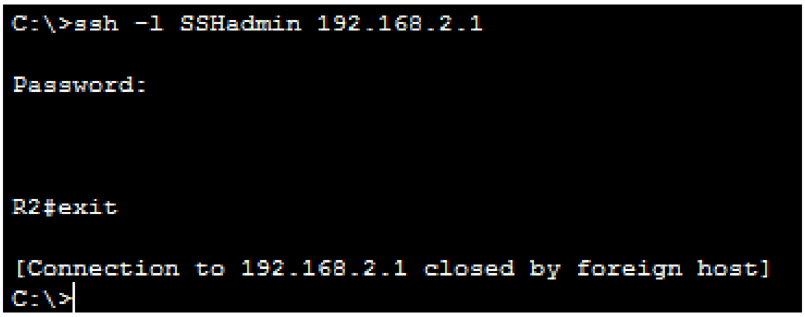
Step 2: From PC-C, verify connectivity to PC-A and R2.

a. From the command prompt, ping **PC-A** (192.168.1.3).



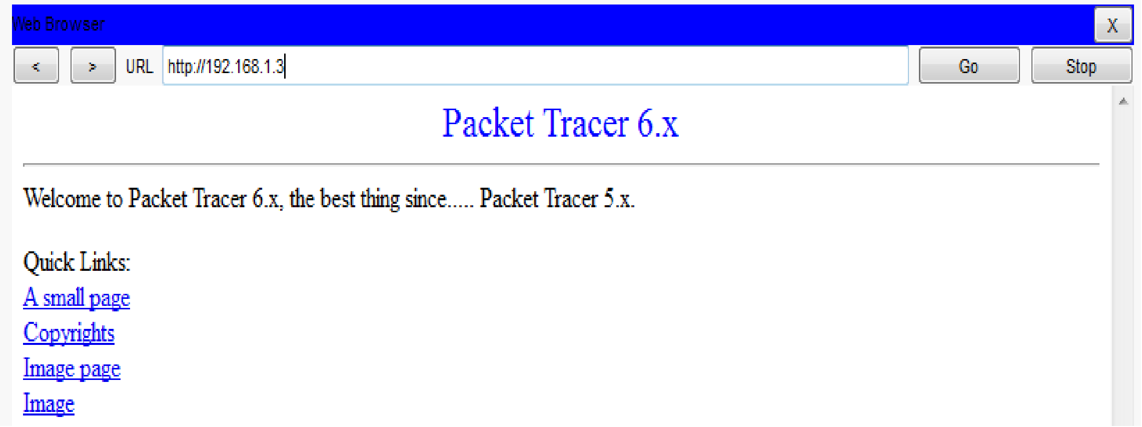
```
Packet Tracer PC Command Line 1.0
C:\>ping 192.168.1.3

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=3ms TTL=125
Reply from 192.168.1.3: bytes=32 time=3ms 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 = 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**. Close the SSH session when finished.



```
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.



Packet Tracer 6.x

Welcome to Packet Tracer 6.x, the best thing since..... Packet Tracer 5.x.

Quick Links:
A small page
Copyrights
Image page
Image

B. 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**. (pswd ciscoconpa55)

```
User Access Verification

Password:

R1>en
Password:
R1#conf t
R1(config)#access-list 10 permit host 192.168.3.3

User Access Verification
Password:

R2>en
Password:
R2#conf t
R2(config)#access-list 10 permit host 192.168.3.3

User Access Verification
Password:

R3>en
Password:
R3#conf t
Enter configuration commands, one per line. End with CNTL/Z.
R3(config)#access-list 10 permit host 192.168.3.3
```

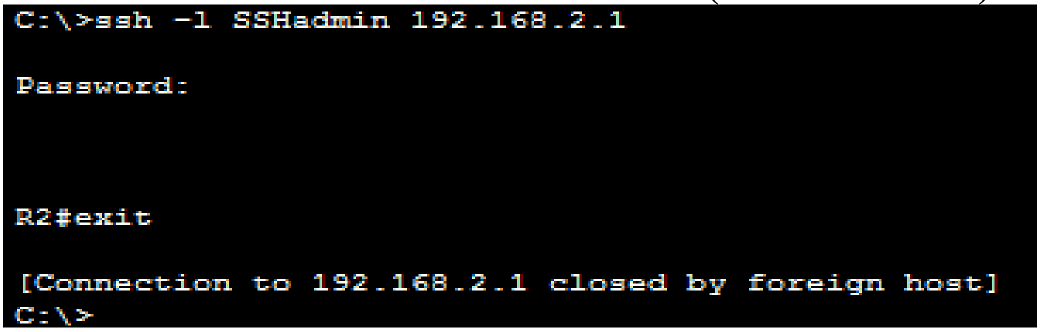
Step 2: Apply ACL 10 to ingress traffic on the VTY lines.
Use the **access-class** command to apply the access list to incoming traffic on the VTY lines.

```
R1(config)#line vty 0 4
R1(config-line)#access-class 10 in
```

R2(config)#line vty 0 4
R2(config-line)#access-class 10 in
R3(config)#line vty 0 4
R3(config-line)#access-class 10 in

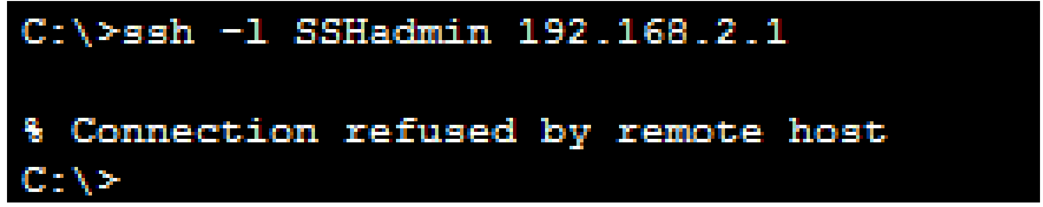
Step 3: Verify exclusive access from management station PC-C.

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



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

Password:



R2#exit

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

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



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

% Connection refused by remote host
C:\>
```

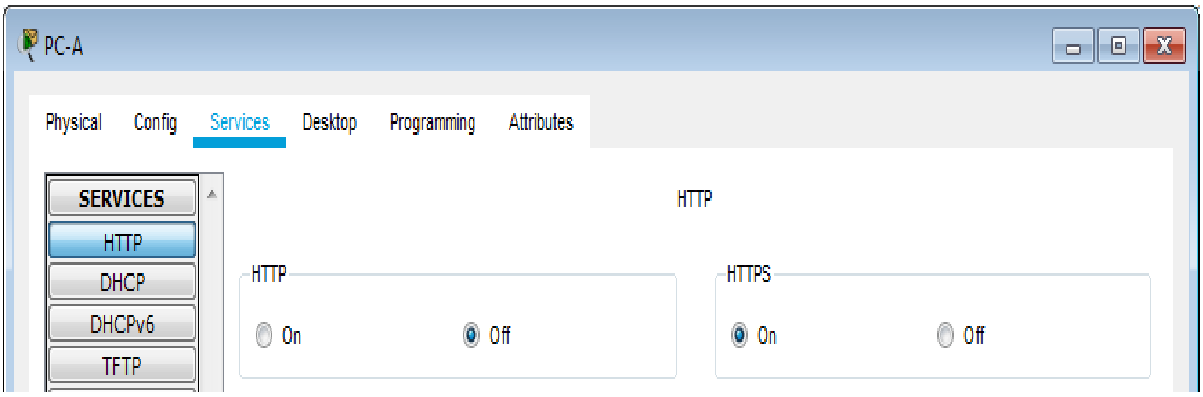
C. Create a Numbered IP ACL 120 on R1

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.

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.



HTTP

HTTP: ○ On ● Off

HTTPS: ● On ○ Off

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

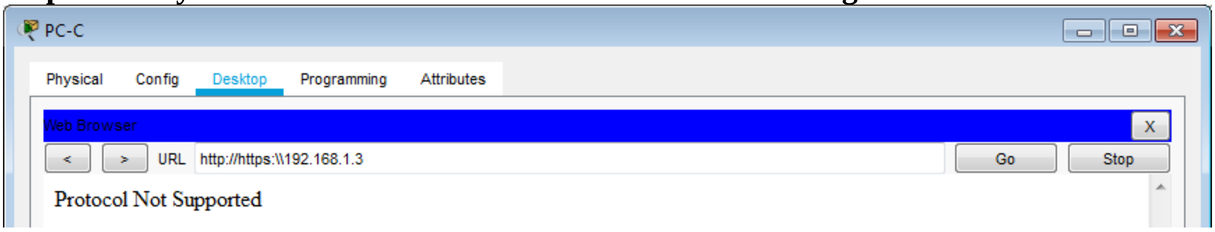
Step 3: Apply the ACL to interface S0/0/0.

Use the **ip access-group** command to apply the access list to incoming traffic on interface S0/0/0.

R1(config)#int 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.



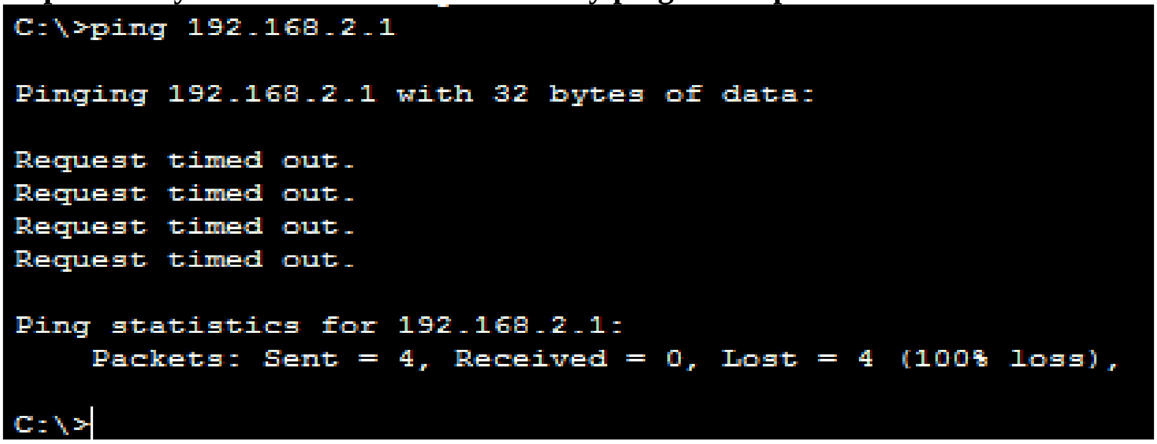
URL: http://https:\\192.168.1.3

Protocol Not Supported

D. 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.



```
C:\>ping 192.168.2.1

Pinging 192.168.2.1 with 32 bytes of data:

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

Ping statistics for 192.168.2.1:
    Packets: Sent = 4, Received = 0, Lost = 4 (100% loss),

C:\>
```

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.

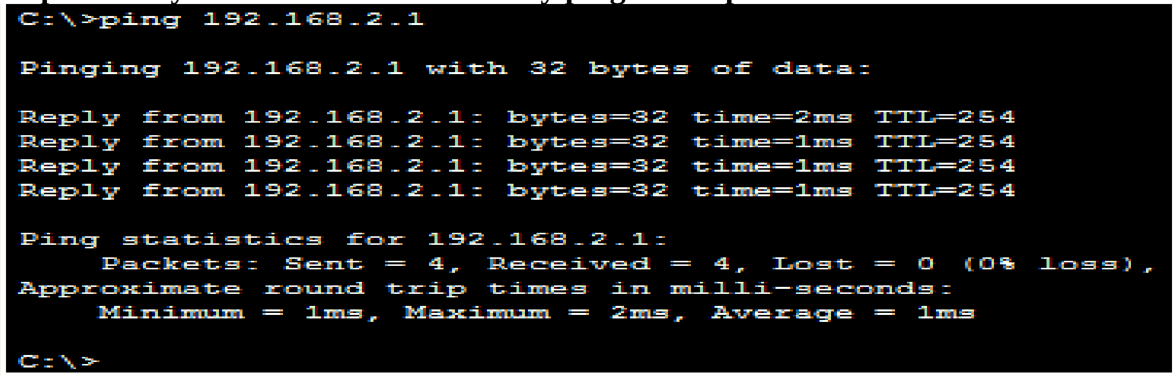
R1(config-if)#access-list 120 permit icmp any any echo-reply

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.



```
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=2ms 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 = 2ms, Average = 1ms

C:\>
```

E. 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.

```
User Access Verification
Password:

R3>en
Password:
R3#conf t
Enter configuration commands, one per line. End with CNTL/Z.
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)#int g0/1
R3(config-if)#ip access-group 110 in
```

F. 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
```

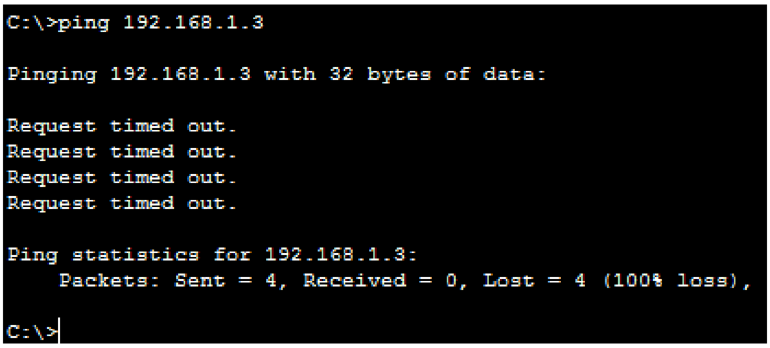
Step 2: Apply the ACL to interface Serial 0/0/1.

Use the **ip access-group** command to apply the access list to incoming traffic on interface Serial 0/0/1.

```
R3(config)#int 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.



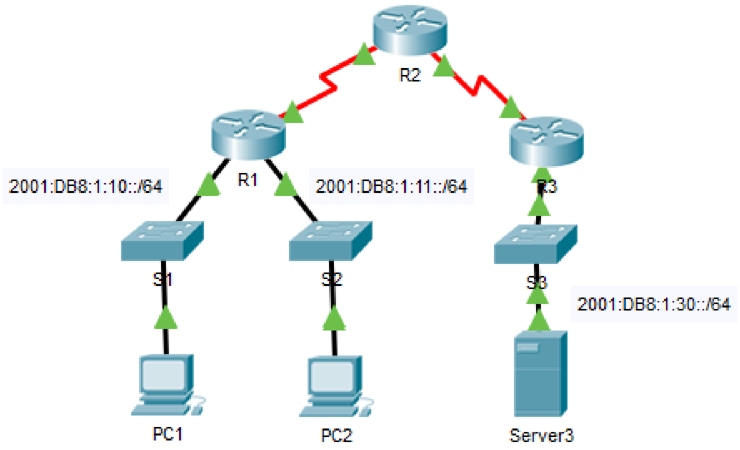
```
C:\>ping 192.168.1.3

Pinging 192.168.1.3 with 32 bytes of data:

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

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

C:\>
```

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

C:\>ssh -l SSHadmin 192.168.2.1

Practical No. 5

Aim: Configure IPv6 ACLs.



A. Configure, Apply, and Verify an IPv6 ACL

Logs indicate that a computer on the 2001:DB8:1:11::0/64 network is repeatedly refreshing a web page. This is causing a Denial-of-Service (DoS) attack against **Server3**. Until the client can be identified and cleaned, you must block HTTP and HTTPS access to that network with an access list.

Step 1: Configure an ACL that will block HTTP and HTTPS access.

Configure an ACL named **BLOCK_HTTP** on **R1** with the following statements.

a. Block HTTP and HTTPS traffic from reaching **Server3**.

```
R1>en
R1#conf t
R1(config)#ipv6 access-list BLOCK_HTTP
R1(config-ipv6-acl)#deny tcp any host 2001:DB8:1:30::30 eq www
R1(config-ipv6-acl)#deny tcp any host 2001:DB8:1:30::30 eq 443
R1(config-ipv6-acl)#
```

b. Allow all other IPv6 traffic to pass.

```
R1(config-ipv6-acl)#permit ipv6 any any
```

Step 2: Apply the ACL to the correct interface.

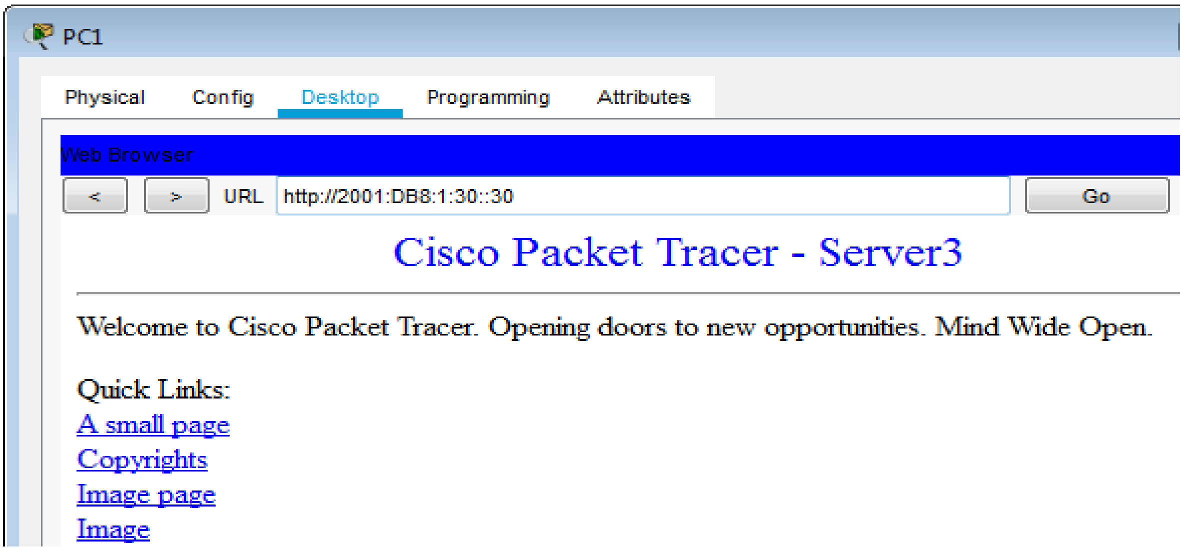
Apply the ACL on the interface closest to the source of the traffic to be blocked. (show link light and point on it interface will be g0/1)

```
R1(config)#interface GigabitEthernet0/1
R1(config-if)#ipv6 traffic-filter BLOCK_HTTP in
```

Step 3: Verify the ACL implementation.

Verify that the ACL is operating as intended by conducting the following tests:

- Open the **web browser** of **PC1** to http://2001:DB8:1:30::30 or https://2001:DB8:1:30::30. The website should appear.



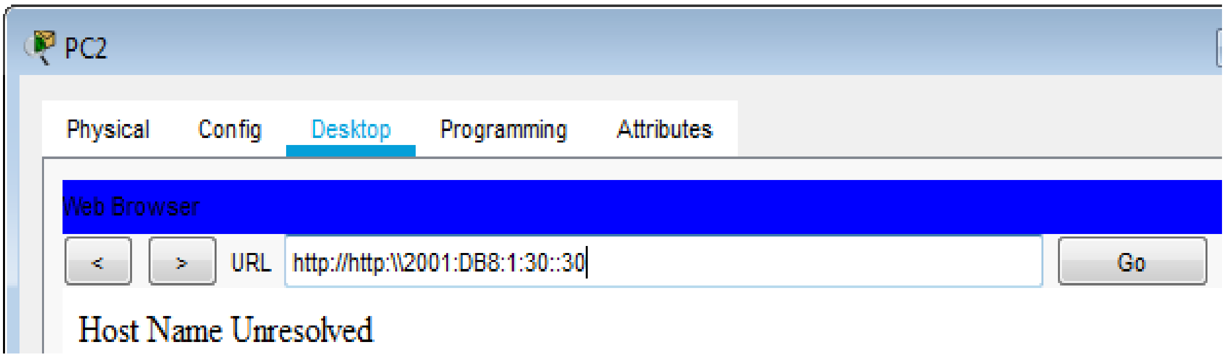
URL http://2001:DB8:1:30::30

Cisco Packet Tracer - Server3

Welcome to Cisco Packet Tracer. Opening doors to new opportunities. Mind Wide Open.

Quick Links:
A small page
Copyrights
Image page
Image

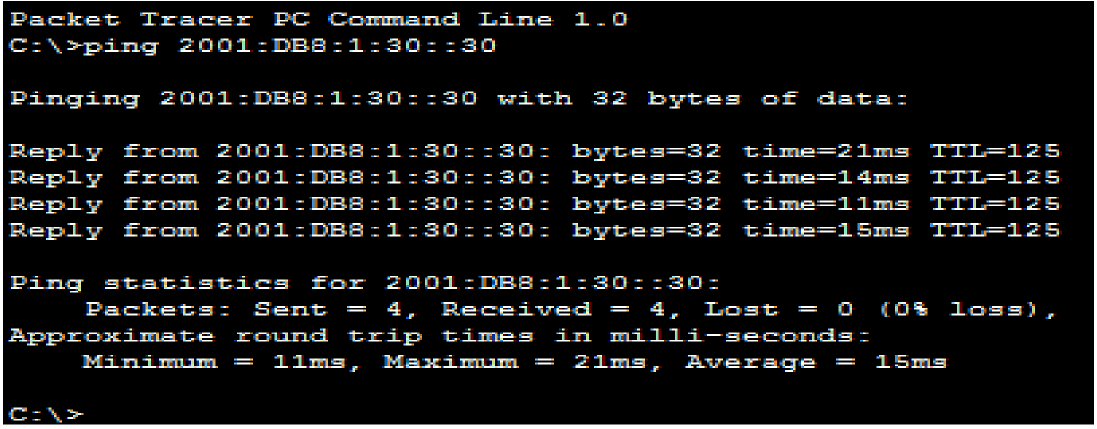
- Open the **web browser** of **PC2** to http://2001:DB8:1:30::30 or https://2001:DB8:1:30::30. The website should be blocked.



URL: http://http:\\2001:DB8:1:30::30

Host Name Unresolved

- Ping from **PC2** to 2001:DB8:1:30::30. The ping should be successful.



```
Packet Tracer PC Command Line 1.0
C:\>ping 2001:DB8:1:30::30

Pinging 2001:DB8:1:30::30 with 32 bytes of data:

Reply from 2001:DB8:1:30::30: bytes=32 time=21ms TTL=125
Reply from 2001:DB8:1:30::30: bytes=32 time=14ms TTL=125
Reply from 2001:DB8:1:30::30: bytes=32 time=11ms TTL=125
Reply from 2001:DB8:1:30::30: bytes=32 time=15ms TTL=125

Ping statistics for 2001:DB8:1:30::30:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 11ms, Maximum = 21ms, Average = 15ms

C:\>
```

B. Configure, Apply, and Verify a Second IPv6 ACL

The logs now indicate that your server is receiving pings from many different IPv6 addresses in a Distributed Denial of Service (DDoS) attack. You must filter ICMP ping requests to your server.

Step 1: Create an access list to block ICMP.

Configure an ACL named **BLOCK_ICMP** on **R3** with the following statements:

a. Block all ICMP traffic from any hosts to any destination.

```
R3>en
R3#conf t
Enter configuration commands, one per line. End with CNTL/Z.
R3(config)#ipv6 access-list BLOCK_ICMP
R3(config-ipv6-acl)#deny icmp any any
```

b. Allow all other IPv6 traffic to pass.

```
R3(config-ipv6-acl)#permit ipv6 any any
```

Step 2: Apply the ACL to the correct interface.

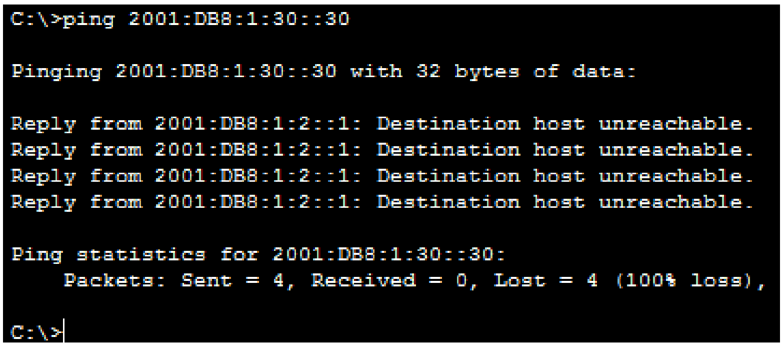
In this case, ICMP traffic can come from any source. To ensure that ICMP traffic is blocked, regardless of its source or any changes that occur to the network topology, apply the ACL closest to the destination.

```
R3(config)#interface GigabitEthernet0/0
```

R3(config-if)#ipv6 traffic-filter BLOCK_ICMP out

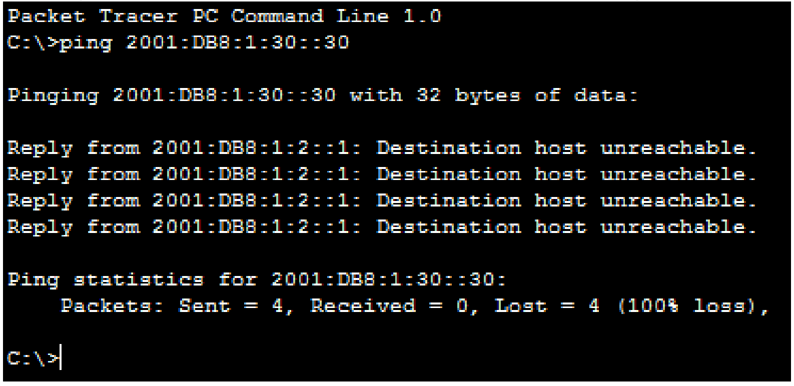
Step 3: Verify that the proper access list functions.

a. Ping from **PC2** to 2001:DB8:1:30::30. The ping should fail.



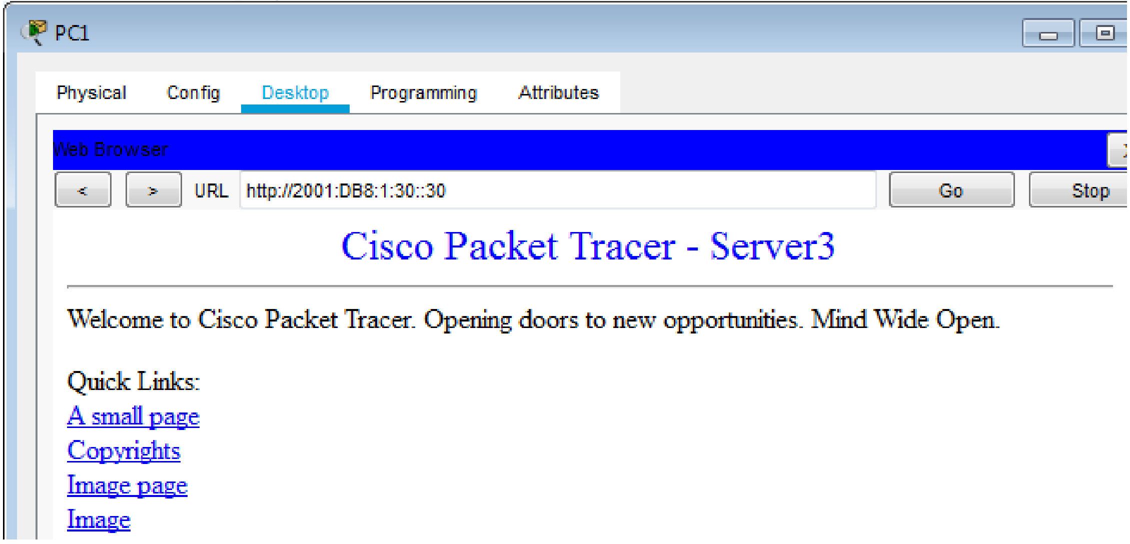
```
C:\>ping 2001:DB8:1:30::30

Pinging 2001:DB8:1:30::30 with 32 bytes of data:

Reply from 2001:DB8:1:2::1: Destination host unreachable.
Reply from 2001:DB8:1:2::1: Destination host unreachable.
Reply from 2001:DB8:1:2::1: Destination host unreachable.
Reply from 2001:DB8:1:2::1: Destination host unreachable.

Ping statistics for 2001:DB8:1:30::30:
    Packets: Sent = 4, Received = 0, Lost = 4 (100% loss),

C:\>
```

b. Ping from **PC1** to 2001:DB8:1:30::30. The ping should fail.



```
Packet Tracer PC Command Line 1.0
C:\>ping 2001:DB8:1:30::30

Pinging 2001:DB8:1:30::30 with 32 bytes of data:

Reply from 2001:DB8:1:2::1: Destination host unreachable.
Reply from 2001:DB8:1:2::1: Destination host unreachable.
Reply from 2001:DB8:1:2::1: Destination host unreachable.
Reply from 2001:DB8:1:2::1: Destination host unreachable.

Ping statistics for 2001:DB8:1:30::30:
    Packets: Sent = 4, Received = 0, Lost = 4 (100% loss),

C:\>
```

Open the **web browser** of **PC1** to http://2001:DB8:1:30::30 or https://2001:DB8:1:30::30. The website should display



URL: http://2001:DB8:1:30::30

Cisco Packet Tracer - Server3

Welcome to Cisco Packet Tracer. Opening doors to new opportunities. Mind Wide Open.

Quick Links:
A small page
Copyrights
Image page
Image

Practical No. 6

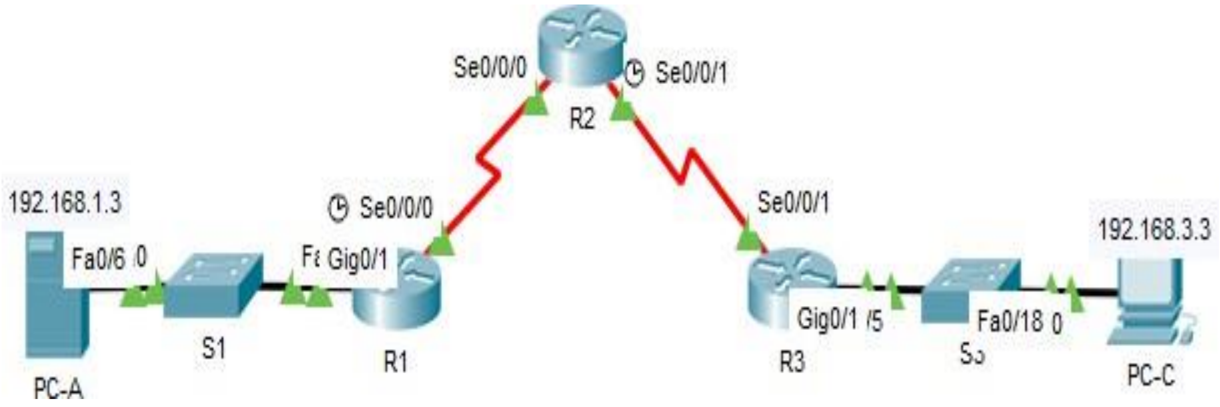
Aim: Configuring Zone Based Policy Firewall (ZPF).

Background/Scenario

ZPFs are the latest development in the evolution of Cisco firewall technologies. In this activity, you will configure a basic ZPF on an edge router R3 that allows internal hosts access to external resources and blocks external hosts from accessing internal resources. You will then verify firewall functionality from internal and external hosts.

The routers have been pre-configured with the following:

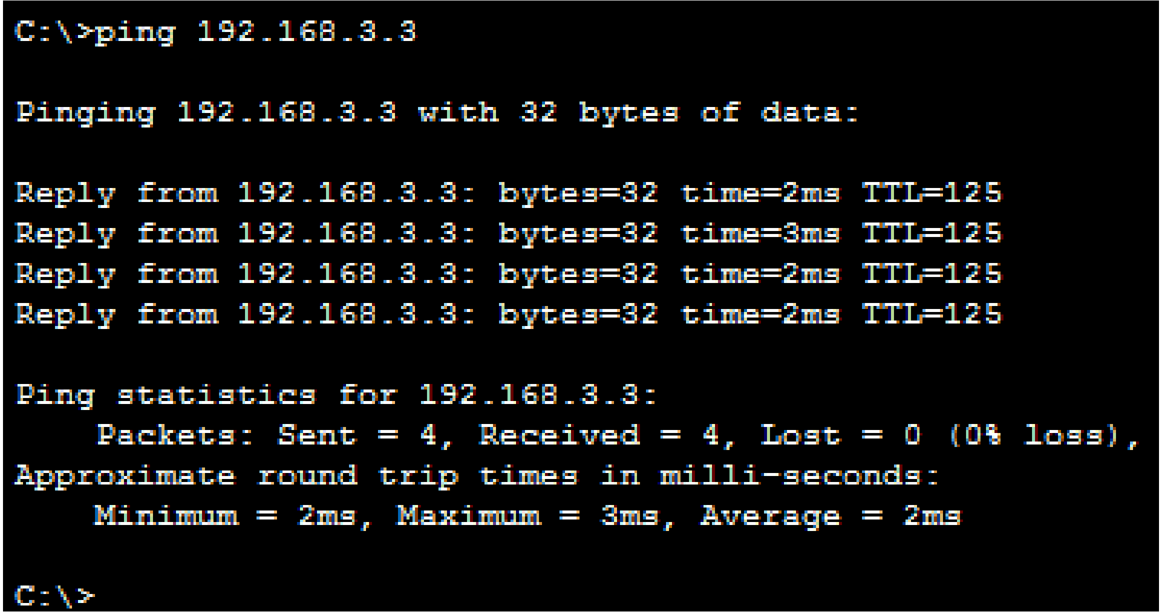
- o Console password: **ciscoconpa55**
- o Password for vty lines: **ciscovtypa55**
- o Enable password: **ciscoenpa55**
- o Host names and IP addressing
- o Local username and password: **Admin** / **Adminpa55**
- o Static routing



A. 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.

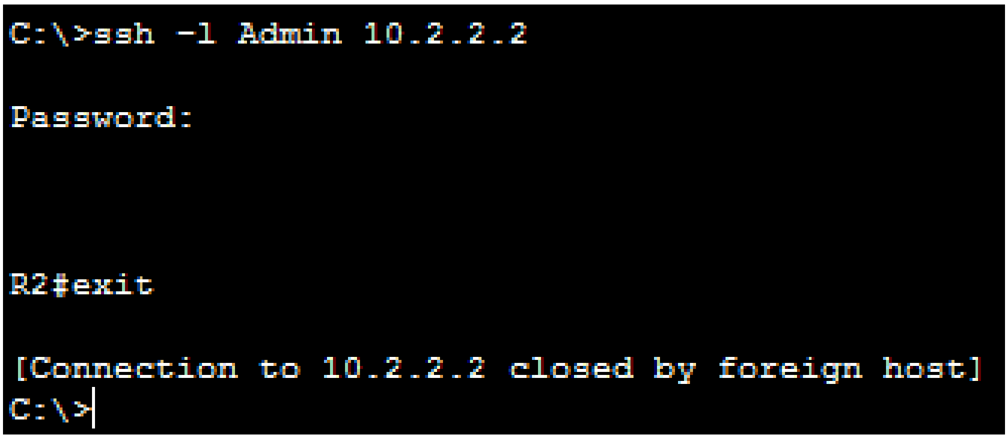


```
C:\>ping 192.168.3.3

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=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

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:\>
```

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 username **Admin** and password **Adminpa55** to log in.

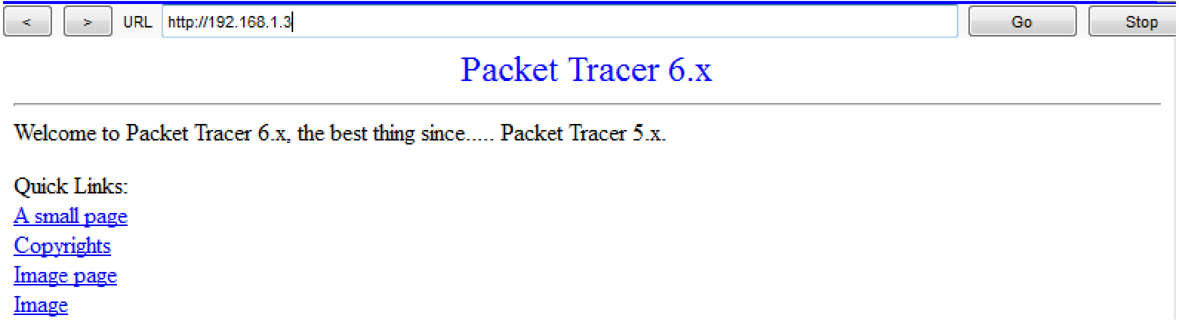
b. Exit the SSH session.



```
C:\>ssh -l Admin 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.



URL http://192.168.1.3

Packet Tracer 6.x

Welcome to Packet Tracer 6.x, the best thing since..... Packet Tracer 5.x.

Quick Links:
A small page
Copyrights
Image page
Image

b. Close the browser on **PC-C**.

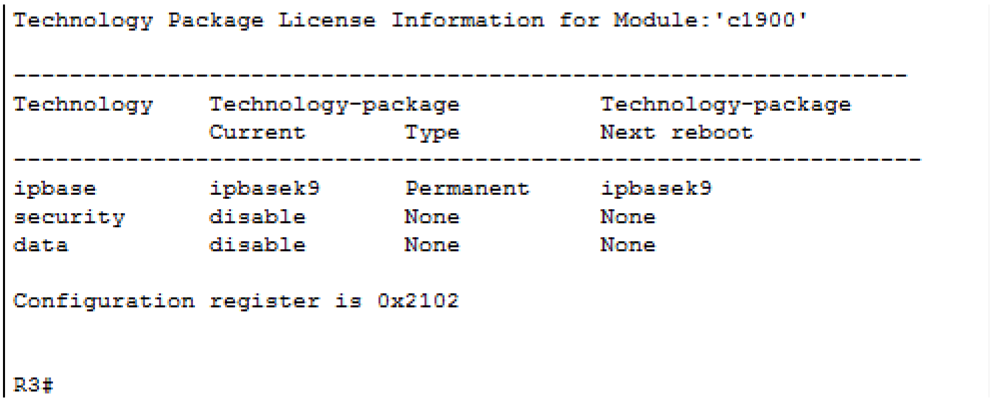
B. Create the Firewall Zones on R3

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.

User Access Verification
Password:
R3>en
Password:
R3#sh version



```
Technology Package License Information for Module:'c1900'

-----------------------------------------------------------------
Technology    Technology-package           Technology-package
              Current       Type           Next reboot
------------------------------------------------------------------
ipbase        ipbasek9      Permanent      ipbasek9
security      disable       None           None
data          disable       None           None

Configuration register is 0x2102


R3#
```

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

c. Accept the end-user license agreement.

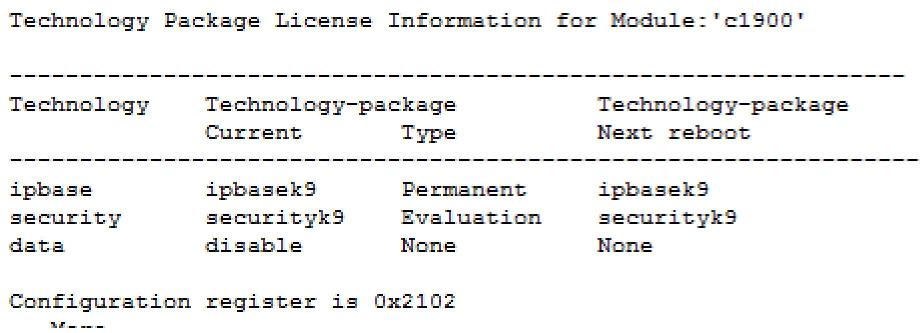
d. Save the running-config and reload the router to enable the security license.

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

R3#
R3#conf t
R3(config)# **license boot module c1900 technology-package securityk9**
ACCEPT? [yes/no]: yes
R3(config)#end
R3#
%SYS-5-CONFIG_I: Configured from console by console
R3#copy run start
Destination filename [startup-config]?
Building configuration...
[OK]
R3#reload
Proceed with reload? [confirm]

User Access Verification
Password:
R3>en
Password:

R3#sh version



```
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
   M---
```

Step 2: Create an internal zone.

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

```
R3#conf t
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)#zone security OUT-ZONE
R3(config-sec-zone)#exit
```

C. 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-NETCLASS-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
R3(config)#
```

D. 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)#policy-map type inspect IN-2-OUT-PMAP
```

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

E. 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 OUT-ZONE

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
R3(config)#

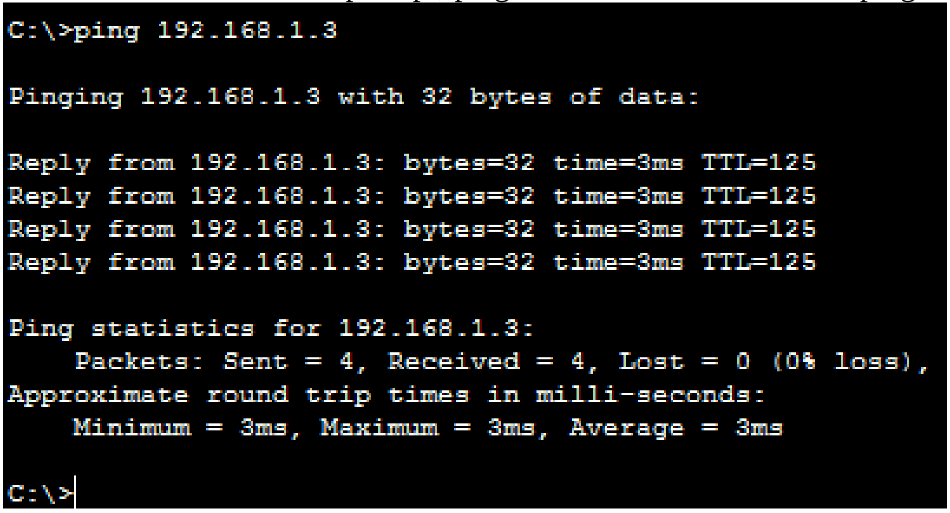
Step 4: Copy the running configuration to the startup configuration.

F. 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.

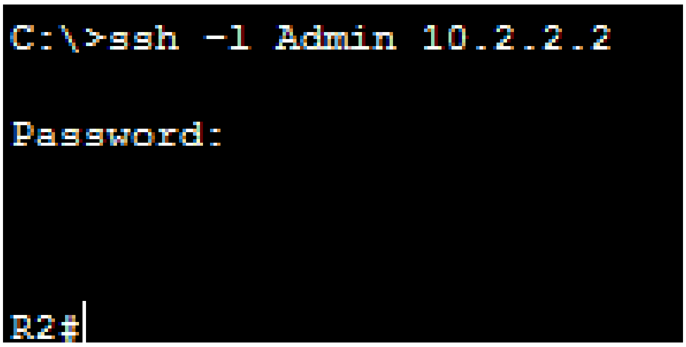
From the **PC-C** command prompt, ping **PC-A** at 192.168.1.3. The ping should succeed.



```
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=3ms TTL=125
Reply from 192.168.1.3: bytes=32 time=3ms TTL=125
Reply from 192.168.1.3: bytes=32 time=3ms TTL=125
Reply from 192.168.1.3: bytes=32 time=3ms 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 = 3ms, Maximum = 3ms, Average = 3ms

C:\>
```

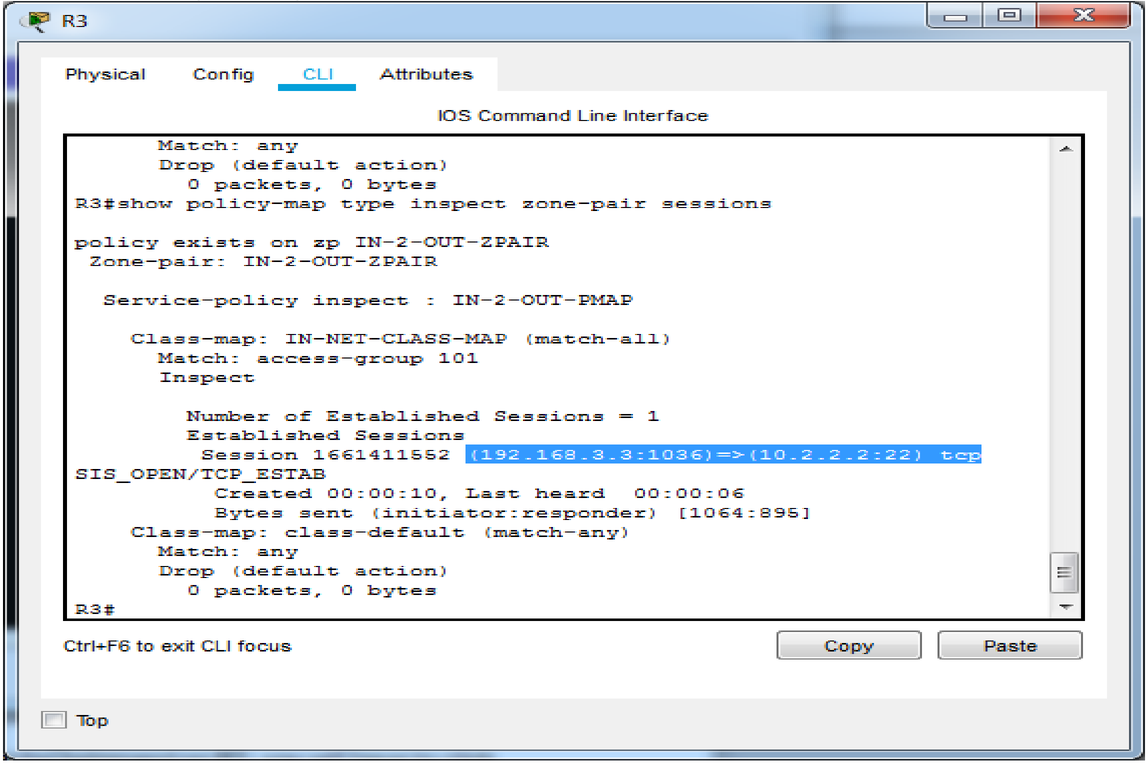
Step 2: From internal PC-C, SSH to the R2 S0/0/1 interface.

a. From the **PC-C** command prompt, SSH to **R2** at 10.2.2.2. Use the username **Admin** and the password **Adminpa55** to access R2. The SSH session should succeed.



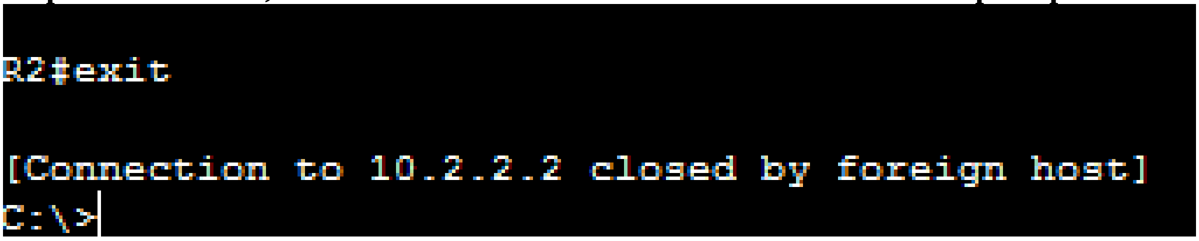
```
C:\>ssh -l Admin 10.2.2.2

Password:



R2#
```

b. While the SSH session is active, issue the command **show policy-map type inspect zone-pair sessions** on **R3** to view established sessions. (check IP add and port number)
R3#show policy-map type inspect zone-pair sessions



```
      Match: any
      Drop (default action)
        0 packets, 0 bytes
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 1661411552 (192.168.3.3:1036)=>(10.2.2.2:22) tcp
SIS_OPEN/TCP_ESTAB
          Created 00:00:10, Last heard  00:00:06
          Bytes sent (initiator:responder) [1064:895]
    Class-map: class-default (match-any)
      Match: any
      Drop (default action)
        0 packets, 0 bytes
R3#
```

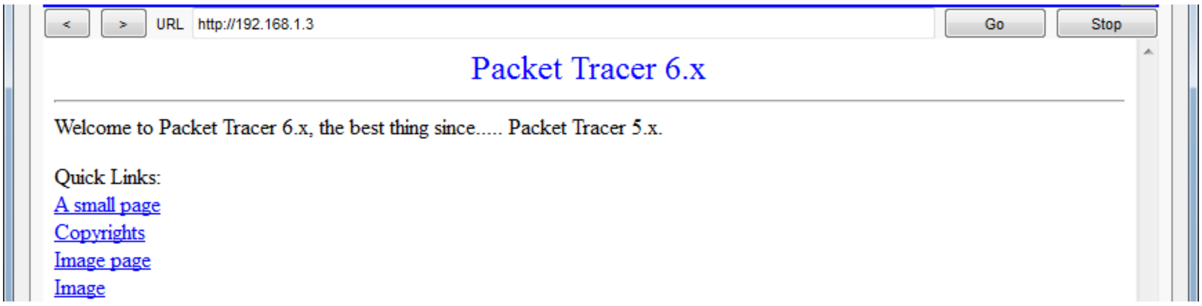
Step 3: From PC-C, exit the SSH session on R2 and close the command prompt window.



```
R2#exit

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

Step 4: From internal PC-C, open a web browser to the PC-A server web page.
Enter the server IP address **192.168.1.3** in the browser URL field, and click **Go**. The HTTP session should succeed. While the HTTP session is active, issue the command **show policy-map type inspect zone-pair sessions** on **R3** to view established sessions.

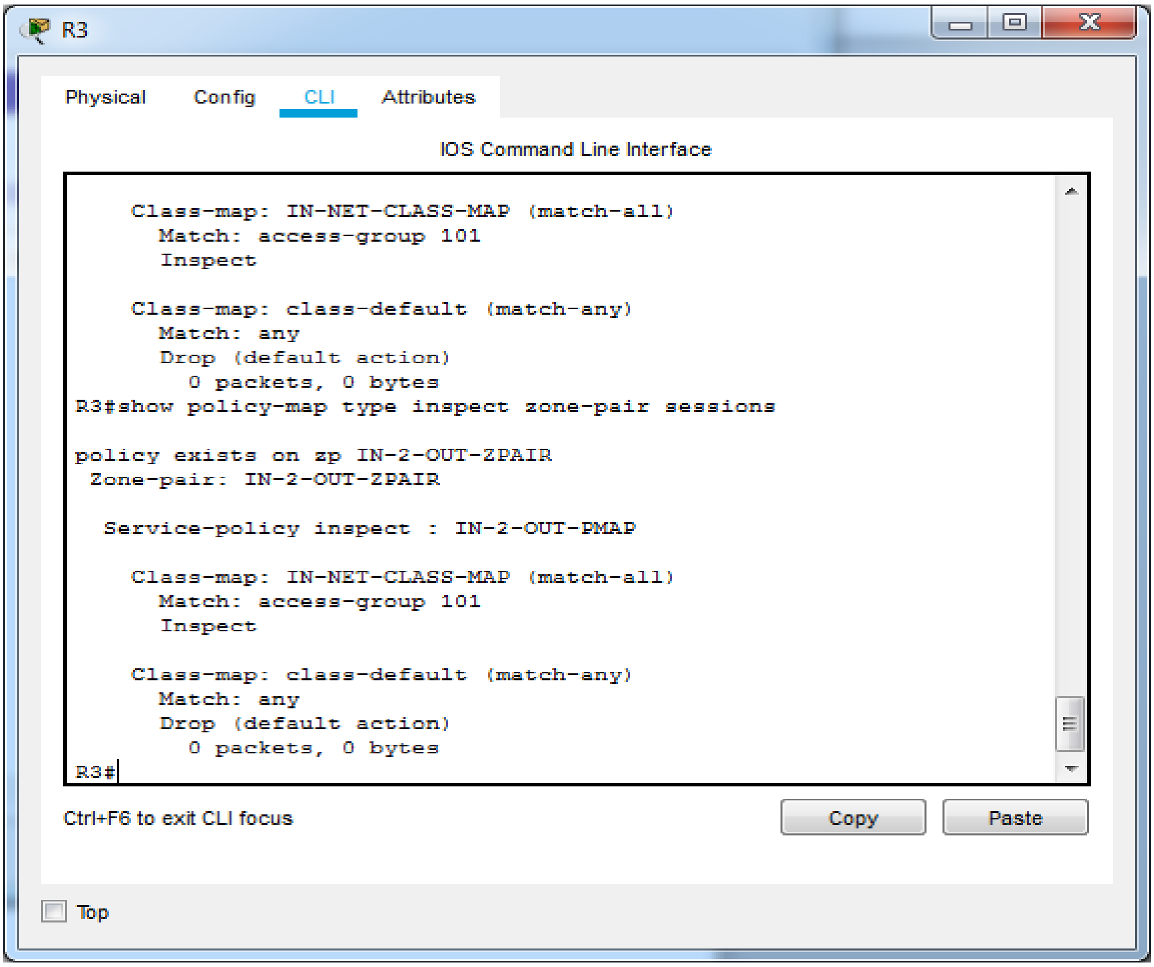
Note: If the HTTP session times out before you execute the command on **R3**, you will have to click the **Go** button on **PC-C** to generate a session between **PC-C** and **PC-A**.



Packet Tracer 6.x

Welcome to Packet Tracer 6.x, the best thing since..... Packet Tracer 5.x.

Quick Links:
A small page
Copyrights
Image page
Image



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

    Class-map: class-default (match-any)
      Match: any
      Drop (default action)
        0 packets, 0 bytes
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

    Class-map: class-default (match-any)
      Match: any
      Drop (default action)
        0 packets, 0 bytes
R3#
```

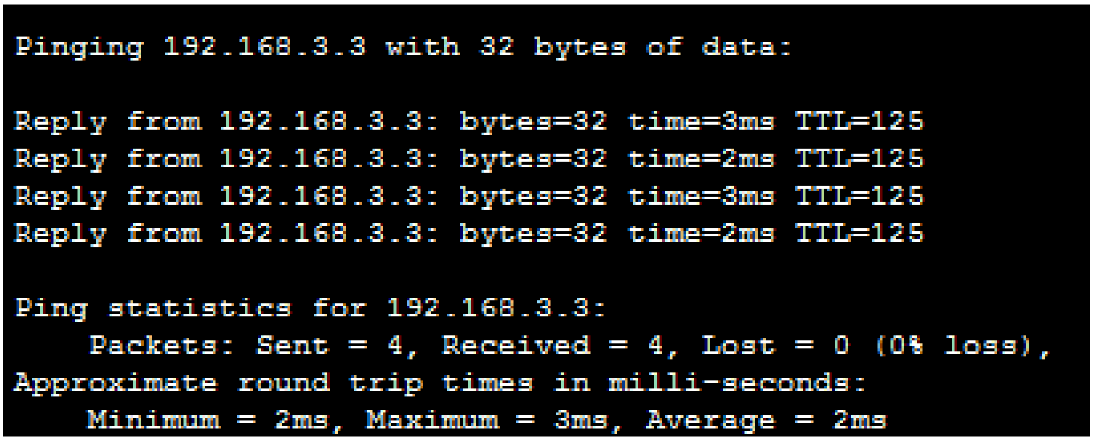
Step 5: Close the browser on PC-C.

G. 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.



```
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=3ms 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
```

Step 2: From R2, ping PC-C.
From **R2**, ping **PC-C** at 192.168.3.3. The ping should fail.

```
User Access Verification

Password:

R2>en
Password:
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 No. 7

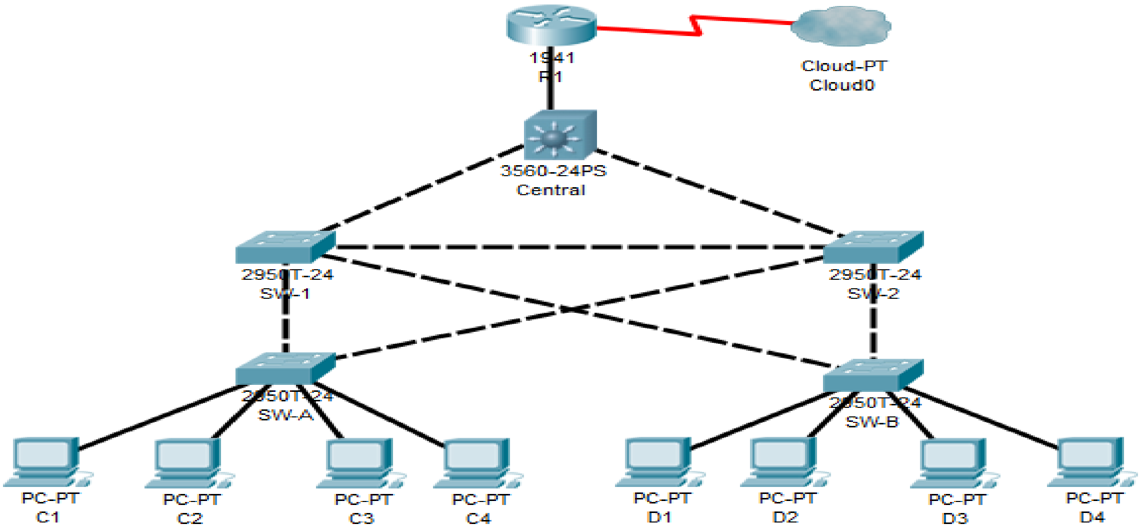
Aim: Configuring Layer 2 Security.

Background / Scenario

There have been a number of attacks on the network recently. For this reason, the network administrator has assigned you the task of configuring Layer 2 security.

For optimum performance and security, the administrator would like to ensure that the root bridge is the 3560 Central switch. To prevent spanning-tree manipulation attacks, the administrator wants to ensure that the STP parameters are secure. To prevent against CAM table overflow attacks, the network administrator has decided to configure port security to limit the number of MAC addresses each switch port can learn. If the number of MAC addresses exceeds the set limit, the administrator would like the port to be shutdown. All switch devices have been preconfigured with the following:

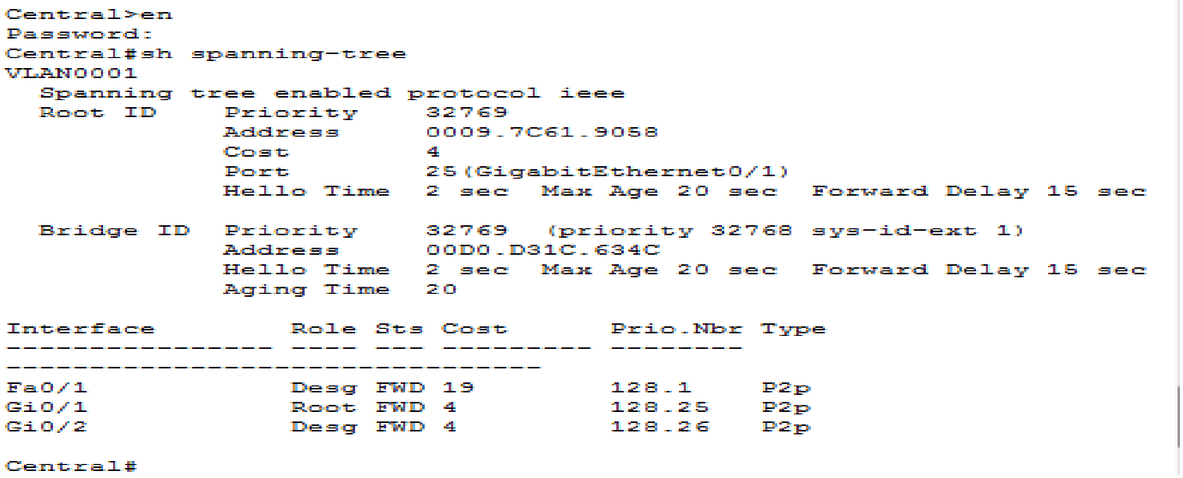
- o Enable password: **ciscoenpa55** o Console password: **ciscoconpa55**
- o SSH username and password: **SSHadmin** / **ciscosshpa55**



A. Configure Root Bridge

Step 1: Determine the current root bridge.

From **Central**, issue the **show spanning-tree** command to determine the current root bridge, to see the ports in use, and to see their status.



```
Central>en
Password:
Central#sh spanning-tree
VLAN0001
  Spanning tree enabled protocol ieee
  Root ID    Priority    32769
             Address     0009.7C61.9058
             Cost        4
             Port        25(GigabitEthernet0/1)
             Hello Time  2 sec  Max Age 20 sec  Forward Delay 15 sec

  Bridge ID  Priority    32769  (priority 32768 sys-id-ext 1)
             Address     00D0.D31C.634C
             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
Gi0/1            Root FWD 4         128.25   P2p
Gi0/2            Desg FWD 4         128.26   P2p

Central#
```

Step 2: Assign Central as the primary root bridge.

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

Central#conf t

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

Central(config)#

Step 3: Assign SW-1 as a secondary root bridge.

Assign **SW-1** as the secondary root bridge using the **spanning-tree vlan 1 root secondary** command.

SW-1>en

Password:

SW-1#conf t

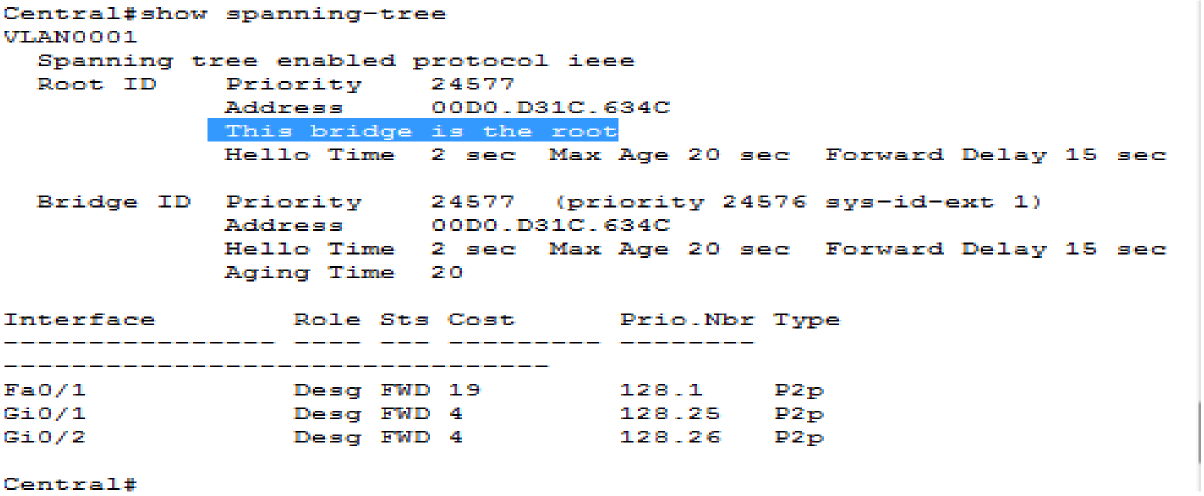
SW-1(config)#spanning-tree vlan 1 root secondary

SW-1(config)#

Step 4: Verify the spanning-tree configuration.

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

Central# show spanning-tree



```
Central#show spanning-tree
VLAN0001
  Spanning tree enabled protocol ieee
  Root ID    Priority    24577
             Address     00D0.D31C.634C
             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     00D0.D31C.634C
             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
Gi0/1            Desg FWD 4         128.25   P2p
Gi0/2            Desg FWD 4         128.26   P2p

Central#
```

B. 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>en
Password:
SW-A#conf t
SW-A(config)#interface range f0/1 - 4
SW-A(config-if-range)#spanning-tree portfast
%Portfast has been configured on FastEthernet0/4 but will only have effect when the
interface is in a non-trunking mode.

SW-B>en
Password:
SW-B#conf t
SW-B(config)#interface range f0/1 - 4
SW-B(config-if-range)#spanning-tree portfast
%Portfast has been configured on FastEthernet0/4 but will only have effect when the
interface is in a non-trunking mode.
```

Step 2: Enable BPDU guard on all access ports.

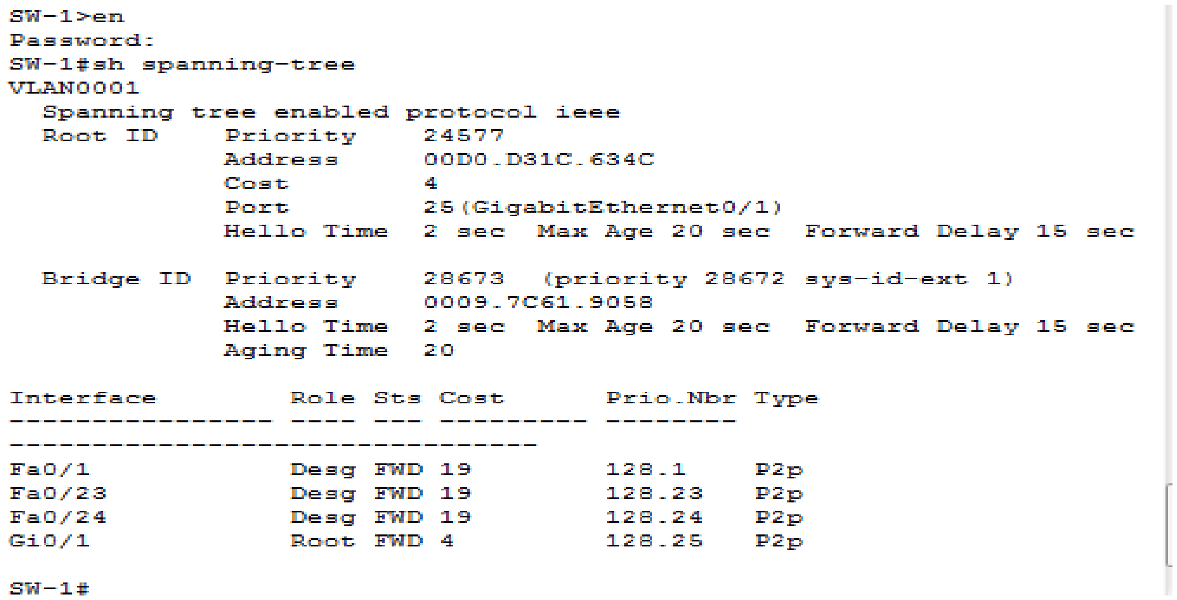
BPDU guard is a feature that can help prevent rogue switches and spoofing on access ports. Enable BPDU guard on **SW-A** and **SW-B** access ports.

```
SW-A#conf t
SW-A(config)#interface range f0/1 - 4
SW-A(config-if-range)#spanning-tree bpduguard enable
SW-A(config-if-range)#

SW-B#conf t
SW-B(config)#interface range f0/1 - 4
SW-B(config-if-range)#spanning-tree bpduguard enable
SW-B(config-if-range)#
```

Note: Spanning-tree BPDU guard can be enabled on each individual port using the **spanning-tree bpduguard enable** command in interface configuration mode or the **spanning-tree portfast bpduguard default** command in global configuration mode. For grading purposes in this activity, please use the **spanning-tree bpduguard enable** command.

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.



```
SW-1>en
Password:
SW-1#sh spanning-tree
VLAN0001
  Spanning tree enabled protocol ieee
  Root ID    Priority    24577
             Address     00D0.D31C.634C
             Cost        4
             Port        25(GigabitEthernet0/1)
             Hello Time  2 sec  Max Age 20 sec  Forward Delay 15 sec

  Bridge ID  Priority    28673  (priority 28672 sys-id-ext 1)
             Address     0009.7C61.9058
             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
Fa0/23           Desg FWD 19        128.23   P2p
Fa0/24           Desg FWD 19        128.24   P2p
Gi0/1            Root FWD 4         128.25   P2p

SW-1#
```

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.

SW-1#conf t
SW-1(config)#interface range f0/23 - 24
SW-1(config-if-range)#spanning-tree guard root
SW-1(config-if-range)#

SW-2>en
Password:
SW-2#conf t
SW-2(config)#interface range f0/23 - 24
SW-2(config-if-range)#spanning-tree guard root
SW-2(config-if-range)#

C. 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.

SW-A>en
Password:
SW-A#conf t
SW-A(config)#interface range f0/1 - 22
SW-A(config-if-range)#switchport mode access
SW-A(config-if-range)#switchport port-security
SW-A(config-if-range)#switchport port-security maximum 2
SW-A(config-if-range)#switchport port-security violation shutdown
SW-A(config-if-range)#switchport port-security mac-address sticky
SW-A(config-if-range)#

```
SW-B>en
Password:
SW-B#conf t
SW-B(config)#interface range f0/1 - 22
SW-B(config-if-range)#switchport mode access
SW-B(config-if-range)#switchport port-security
SW-B(config-if-range)#switchport port-security maximum 2
SW-B(config-if-range)#switchport port-security violation shutdown
SW-B(config-if-range)#switchport port-security mac-address sticky
SW-B(config-if-range)#
```

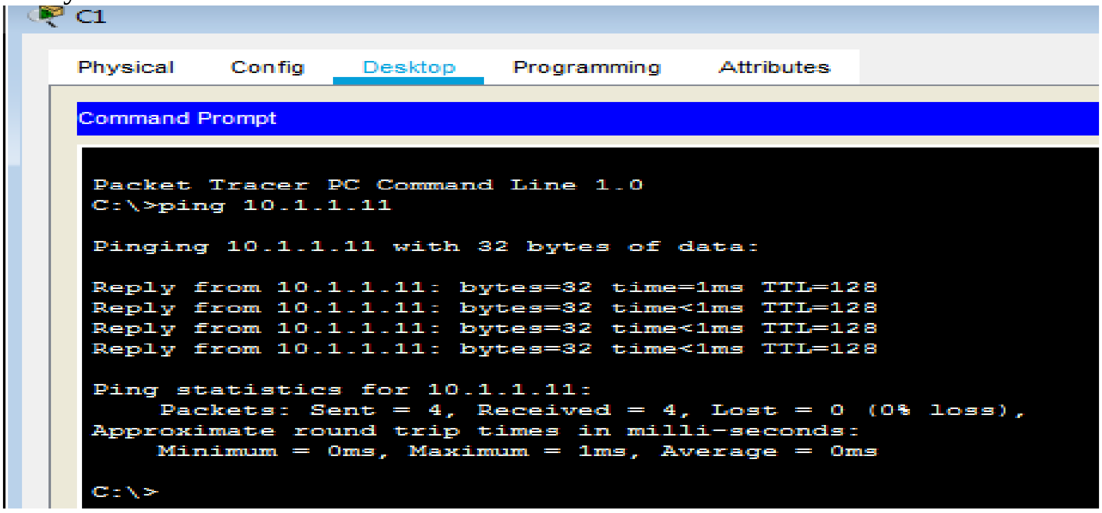
Why is port security not enabled on ports that are connected to other switch devices? Ports connected to other switch devices have a multitude of MAC addresses learned for that single port. Limiting the number of MAC addresses that can be learned on these ports can significantly impact network functionality.

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.

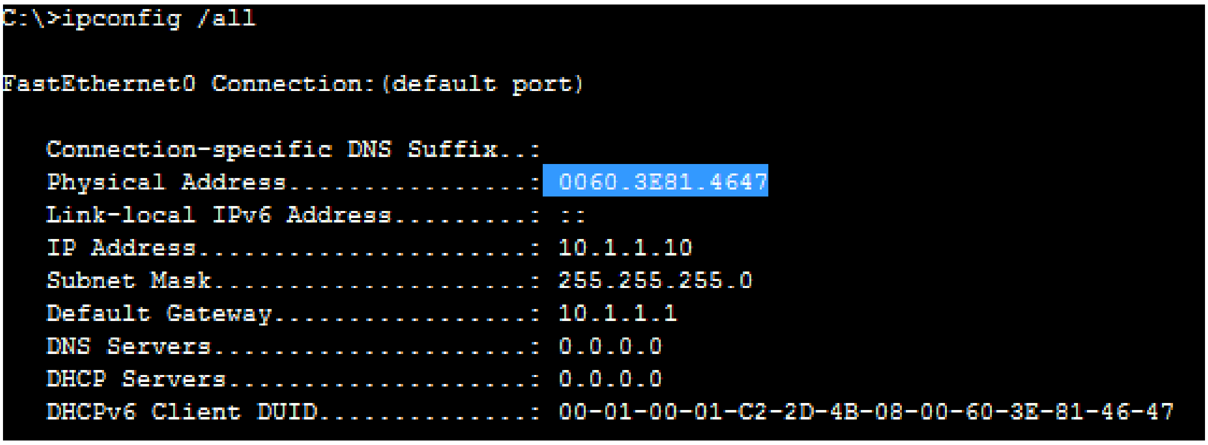
```
SW-A#show port-security interface f0/1
Port Security                : Enabled
Port Status                  : Secure-up
Violation Mode               : Shutdown
Aging Time                   : 0 mins
Aging Type                   : Absolute
SecureStatic Address Aging   : Disabled
Maximum MAC Addresses        : 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
```

b. Ping from **C1** to **C2** and issue the command **show port-security interface f0/1** again to verify that the switch has learned the MAC address for **C1**.



```
Packet Tracer PC Command Line 1.0
C:\>ping 10.1.1.11

Pinging 10.1.1.11 with 32 bytes of data:

Reply from 10.1.1.11: bytes=32 time=1ms TTL=128
Reply from 10.1.1.11: bytes=32 time<1ms TTL=128
Reply from 10.1.1.11: bytes=32 time<1ms TTL=128
Reply from 10.1.1.11: bytes=32 time<1ms TTL=128

Ping statistics for 10.1.1.11:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 0ms, Maximum = 1ms, Average = 0ms

C:\>
```

```
SW-A#show port-security interface f0/1
Port Security                        : Enabled
Port Status                          : Secure-up
Violation Mode                       : Shutdown
Aging Time                           : 0 mins
Aging Type                           : Absolute
SecureStatic Address Aging           : Disabled
Maximum MAC Addresses                : 2
Total MAC Addresses                  : 1
Configured MAC Addresses             : 0
Sticky MAC Addresses                 : 1
Last Source Address:Vlan             : 0060.3E81.4647:1
Security Violation Count             : 0
```



```
C:\>ipconfig /all

FastEthernet0 Connection:(default port)

   Connection-specific DNS Suffix..:
   Physical Address................: 0060.3E81.4647
   Link-local IPv6 Address.........: ::
   IP Address......................: 10.1.1.10
   Subnet Mask.....................: 255.255.255.0
   Default Gateway.................: 10.1.1.1
   DNS Servers.....................: 0.0.0.0
   DHCP Servers....................: 0.0.0.0
   DHCPv6 Client DUID..............: 00-01-00-01-C2-2D-4B-08-00-60-3E-81-46-47
```

Step 3: Disable unused ports.

Disable all ports that are currently unused.

```
SW-A#conf t
SW-A(config)#interface range f0/5 - 22
SW-A(config-if-range)#shutdown

SW-B>en
Password:
SW-B#conf t
SW-B(config)#interface range f0/5 - 22
SW-B(config-if-range)#shutdown
```

Practical No. 8

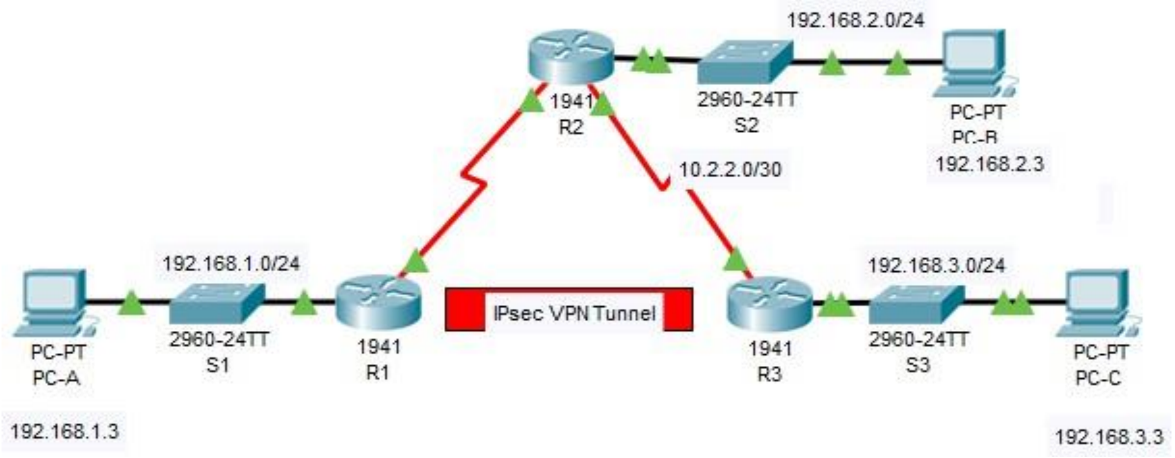
Aim: Configure and verify a site-to-site IPsec VPN using CLI.

Background / Scenario

The network topology shows three routers. Your task is to configure R1 and R3 to support a site-to-site IPsec VPN when traffic flows between their respective LANs. The IPsec VPN tunnel is from R1 to R3 via R2. R2 acts as a pass-through and has no knowledge of the VPN. IPsec provides secure transmission of sensitive information over unprotected networks, such as the Internet. IPsec operates at the network layer and protects and authenticates IP packets between participating IPsec devices (peers), such as Cisco routers. **ISAKMP Phase 1 Policy Parameters**

The routers have been pre-configured with the following:

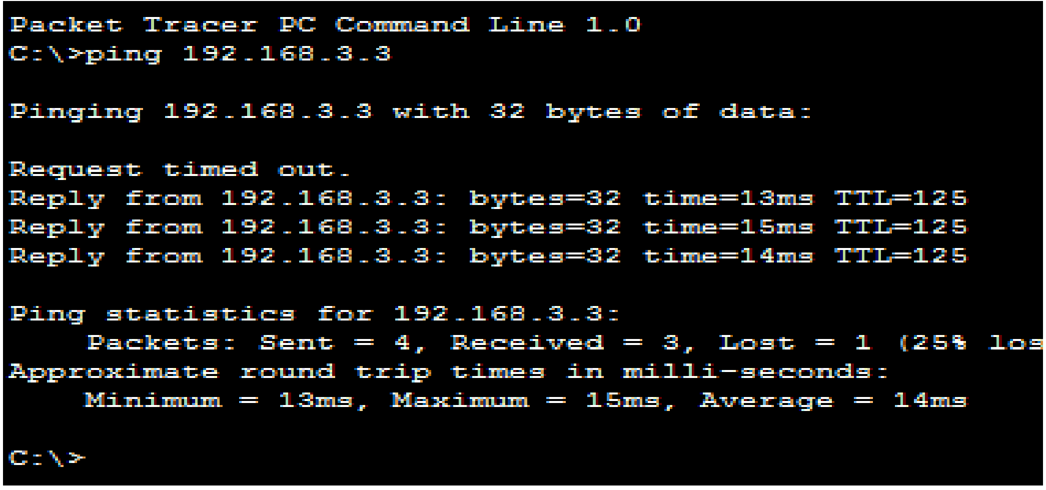
- Password for console line: **ciscoconpa55**
- Password for vty lines: **ciscovtypa55**
- Enable password: **ciscoenpa55**
- SSH username and password: **SSHadmin / ciscosshpa55**
- OSPF 101



A. Configure IPsec Parameters on R1

Step 1: Test connectivity.

Ping from PC-A to PC-C.



```
Packet Tracer PC Command Line 1.0
C:\>ping 192.168.3.3

Pinging 192.168.3.3 with 32 bytes of data:

Request timed out.
Reply from 192.168.3.3: bytes=32 time=13ms TTL=125
Reply from 192.168.3.3: bytes=32 time=15ms TTL=125
Reply from 192.168.3.3: bytes=32 time=14ms TTL=125

Ping statistics for 192.168.3.3:
    Packets: Sent = 4, Received = 3, Lost = 1 (25% los
Approximate round trip times in milli-seconds:
    Minimum = 13ms, Maximum = 15ms, Average = 14ms

C:\>
```

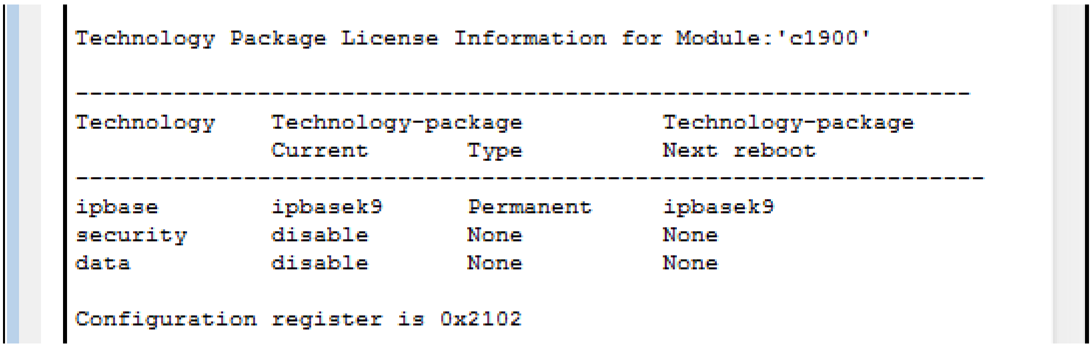
Step 2: Enable the Security Technology package.

a. On R1, issue the **show version** command to view the Security Technology package license information.

User Access Verification

Password:

R1>en
Password:
R1#sh version



```
Technology Package License Information for Module:'c1900'

----------------------------------------------------------------
Technology    Technology-package           Technology-package
              Current       Type           Next reboot
-----------------------------------------------------------------
ipbase        ipbasek9      Permanent      ipbasek9
security      disable       None           None
data          disable       None           None

Configuration register is 0x2102
```

b. If the Security Technology package has not been enabled, use the following command to enable the package.
R1#conf t
Enter configuration commands, one per line. End with CNTL/Z.
R1(config)#license boot module c1900 technology-package securityk9

c. Accept the end-user license agreement.

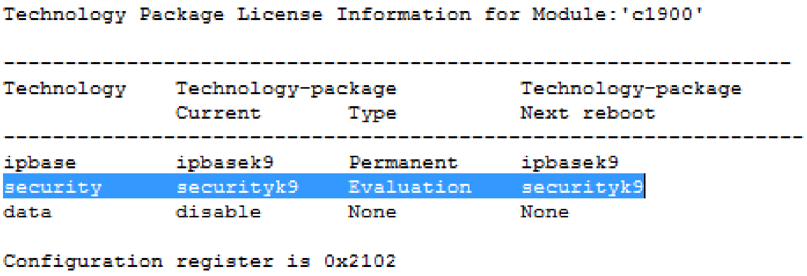
ACCEPT? [yes/no]: yes
R1(config)#: %IOS_LICENSE_IMAGE_APPLICATION-6-LICENSE_LEVEL: Module name = C1900 Next reboot level = securityk9 and License = securityk9

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

```
R1(config)#end
R1#
%SYS-5-CONFIG_I: Configured from console by console

R1#copy running-config startup-config
Destination filename [startup-config]?
Building configuration...
[OK]
R1#
R1#reload
Proceed with reload? [confirm](press enter)
```

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

```
User Access Verification

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

00:00:20: %OSPF-5-ADJCHG: Process 101, Nbr 192.168.2.1 on Serial0/0/0 from
LOADING to FULL, Loading Done
Password:
R1>en
Password:
R1#sh version
```



```
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
```

Step 3: Identify interesting traffic on R1.

Configure ACL 110 to identify the traffic from the LAN on R1 to the LAN on R3 as interesting. This interesting traffic will trigger the IPsec VPN to be implemented when there is traffic between the R1 to R3 LANs. All other traffic sourced from the LANs will not be encrypted. Because of the implicit **deny all**, there is no need to configure a **deny ip any any** statement.

```
R1#
R1#conf t
Enter configuration commands, one per line. End with CNTL/Z.
R1(config)#access-list 110 permit ip 192.168.1.0 0.0.0.255 192.168.3.0 0.0.0.255
```

Step 4: Configure the IKE Phase 1 ISAKMP policy on R1.

Configure the **crypto ISAKMP policy 10** properties on R1 along with the shared crypto key **vpnpa55**. Refer to the ISAKMP Phase 1 table for the specific parameters to configure. Default values do not have to be configured. Therefore, only the encryption method, key exchange method, and DH method must be configured.

Note: The highest DH group currently supported by Packet Tracer is group 5. In a production network, you would configure at least DH 14.

```
R1(config)#crypto isakmp policy 10
R1(config-isakmp)#encryption aes 256
R1(config-isakmp)#authentication pre-share
R1(config-isakmp)#group 5
R1(config-isakmp)#exit
R1(config)#crypto isakmp key vpnpa55 address 10.2.2.2
```

Step 5: Configure the IKE Phase 2 IPsec policy on R1.

a. Create the transform-set VPN-SET to use **esp-aes** and **esp-sha-hmac**.

```
R1(config)#crypto ipsec transform-set VPN-SET esp-aes esp-sha-hmac
```

b. Create the crypto map VPN-MAP that binds all of the Phase 2 parameters together. Use sequence number 10 and identify it as an ipsec-isakmp map.

```
R1(config)#crypto map VPN-MAP 10 ipsec-isakmp
% NOTE: This new crypto map will remain disabled until a peer
and a valid access list have been configured.
R1(config-crypto-map)#description VPN connection to R3
R1(config-crypto-map)#set peer 10.2.2.2
R1(config-crypto-map)#set transform-set VPN-SET
R1(config-crypto-map)#match address 110
R1(config-crypto-map)#exit
```

Step 6: Configure the crypto map on the outgoing interface.

Bind the **VPN-MAP** crypto map to the outgoing Serial 0/0/0 interface.

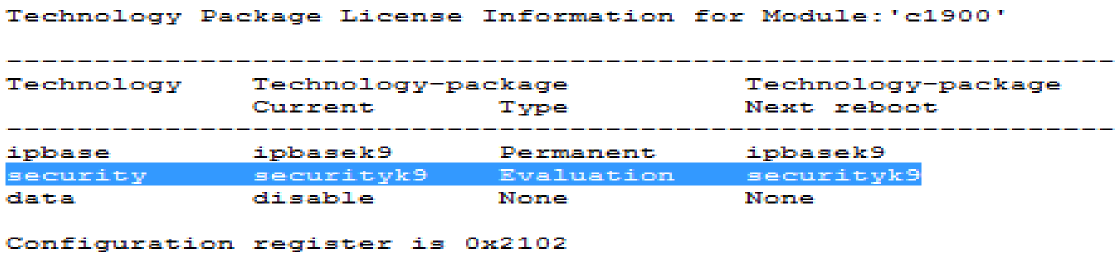
```
R1(config)#int s0/0/0
R1(config-if)#crypto map VPN-MAP
*Jan 3 07:16:26.785: %CRYPTO-6-ISAKMP_ON_OFF: ISAKMP is ON
```

B. Configure IPsec Parameters on R3

Step 1: Enable the Security Technology package.

a. On R3, issue the **show version** command to verify that the Security Technology package license information has been enabled.

```
User Access Verification

Password:

R3>en
Password:
R3#sh version
```



```
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
```

b. If the Security Technology package has not been enabled, enable the package and reload R3.

Step 2: Configure router R3 to support a site-to-site VPN with R1.

Configure reciprocating parameters on R3. Configure ACL 110 identifying the traffic from the LAN on R3 to the LAN on R1 as interesting.

```
R3#conf t
Enter configuration commands, one per line. End with CNTL/Z.
R3(config)#access-list 110 permit ip 192.168.3.0 0.0.0.255 192.168.1.0 0.0.0.255
```

Step 3: Configure the IKE Phase 1 ISAKMP properties on R3.

Configure the crypto ISAKMP policy 10 properties on R3 along with the shared crypto key vpnpa55.

```
R3(config)#crypto isakmp policy 10
R3(config-isakmp)#encryption aes 256
R3(config-isakmp)#authentication pre-share
R3(config-isakmp)#group 5
R3(config-isakmp)#exit
R3(config)#crypto isakmp key vpnpa55 address 10.1.1.2
```

Step 4: Configure the IKE Phase 2 IPsec policy on R3.

a. Create the transform-set VPN-SET to use **esp-aes** and **esp-sha-hmac**.

```
R3(config)#crypto ipsec transform-set VPN-SET esp-aes esp-sha-hmac
```

b. Create the crypto map VPN-MAP that binds all of the Phase 2 parameters together. Use sequence number 10 and identify it as an ipsec-isakmp map.

```
R3(config)#crypto map VPN-MAP 10 ipsec-isakmp
% NOTE: This new crypto map will remain disabled until a peer
and a valid access list have been configured.
R3(config-crypto-map)#description VPN connection to R1
R3(config-crypto-map)#set peer 10.1.1.2
R3(config-crypto-map)#set transform-set VPN-SET
R3(config-crypto-map)#match address 110
R3(config-crypto-map)#exit
```

Step 5: Configure the crypto map on the outgoing interface.

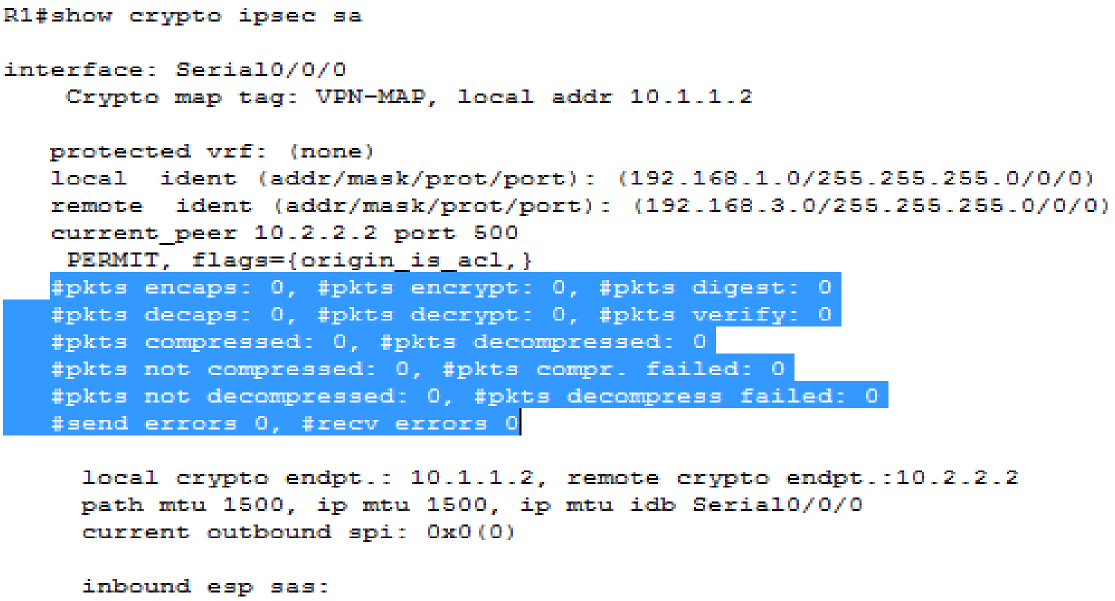
Bind the VPN-MAP crypto map to the outgoing Serial 0/0/1 interface. **Note**: This is not graded.

```
R3(config)#int s0/0/1
R3(config-if)#crypto map VPN-MAP
*Jan 3 07:16:26.785: %CRYPTO-6-ISAKMP_ON_OFF: ISAKMP is ON
```

C. Verify the IPsec VPN

Step 1: Verify the tunnel prior to interesting traffic.

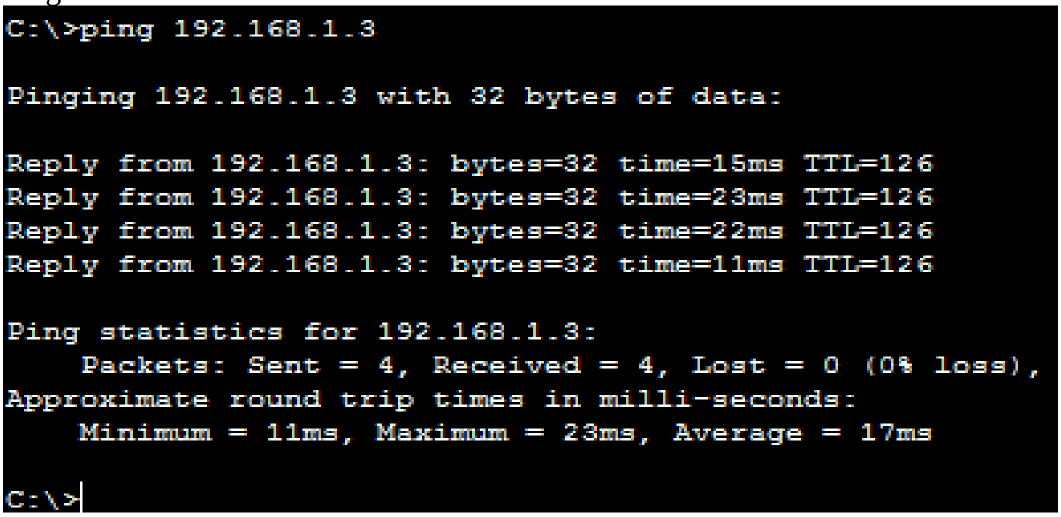
Issue the **show crypto ipsec sa**command on R1. Notice that the number of packets encapsulated, encrypted, decapsulated, and decrypted are all set to 0.



```
R1#show crypto ipsec sa

interface: Serial0/0/0
    Crypto map tag: VPN-MAP, local addr 10.1.1.2

   protected vrf: (none)
   local  ident (addr/mask/prot/port): (192.168.1.0/255.255.255.0/0/0)
   remote  ident (addr/mask/prot/port): (192.168.3.0/255.255.255.0/0/0)
   current_peer 10.2.2.2 port 500
    PERMIT, flags={origin_is_acl,}
   #pkts encaps: 0, #pkts encrypt: 0, #pkts digest: 0
   #pkts decaps: 0, #pkts decrypt: 0, #pkts verify: 0
   #pkts compressed: 0, #pkts decompressed: 0
   #pkts not compressed: 0, #pkts compr. failed: 0
   #pkts not decompressed: 0, #pkts decompress failed: 0
   #send errors 0, #recv errors 0

     local crypto endpt.: 10.1.1.2, remote crypto endpt.:10.2.2.2
     path mtu 1500, ip mtu 1500, ip mtu idb Serial0/0/0
     current outbound spi: 0x0(0)

     inbound esp sas:
```

Step 2: Create interesting traffic.

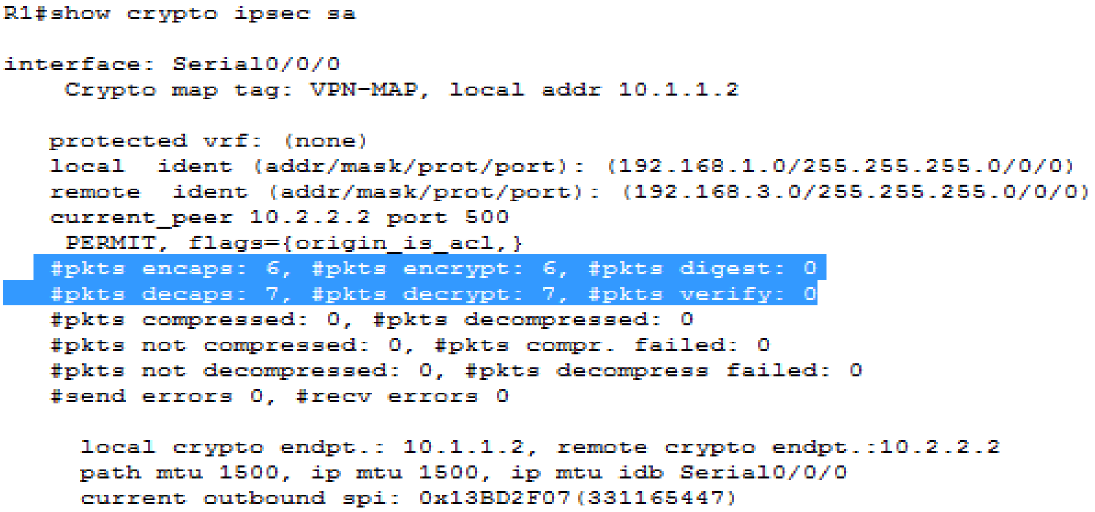
Ping PC-C from PC-A.



```
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=15ms TTL=126
Reply from 192.168.1.3: bytes=32 time=23ms TTL=126
Reply from 192.168.1.3: bytes=32 time=22ms TTL=126
Reply from 192.168.1.3: bytes=32 time=11ms TTL=126

Ping statistics for 192.168.1.3:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 11ms, Maximum = 23ms, Average = 17ms

C:\>
```

Step 3: Verify the tunnel after interesting traffic.

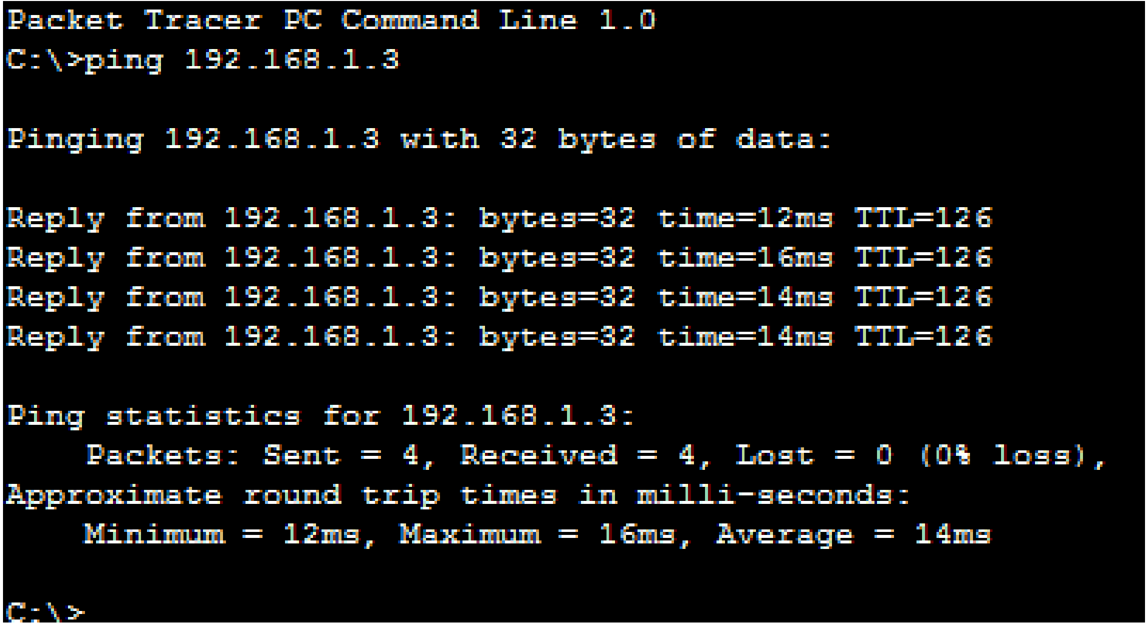
On R1, re-issue the **show crypto ipsec sa**command. Notice that the number of packets is more than 0, which indicates that the IPsec VPN tunnel is working.



```
R1#show crypto ipsec sa

interface: Serial0/0/0
    Crypto map tag: VPN-MAP, local addr 10.1.1.2

   protected vrf: (none)
   local  ident (addr/mask/prot/port): (192.168.1.0/255.255.255.0/0/0)
   remote  ident (addr/mask/prot/port): (192.168.3.0/255.255.255.0/0/0)
   current_peer 10.2.2.2 port 500
    PERMIT, flags={origin_is_acl,}
   #pkts encaps: 6, #pkts encrypt: 6, #pkts digest: 0
   #pkts decaps: 7, #pkts decrypt: 7, #pkts verify: 0
   #pkts compressed: 0, #pkts decompressed: 0
   #pkts not compressed: 0, #pkts compr. failed: 0
   #pkts not decompressed: 0, #pkts decompress failed: 0
   #send errors 0, #recv errors 0

     local crypto endpt.: 10.1.1.2, remote crypto endpt.:10.2.2.2
     path mtu 1500, ip mtu 1500, ip mtu idb Serial0/0/0
     current outbound spi: 0x13BD2F07(331165447)
```

Step 4: Create uninteresting traffic.

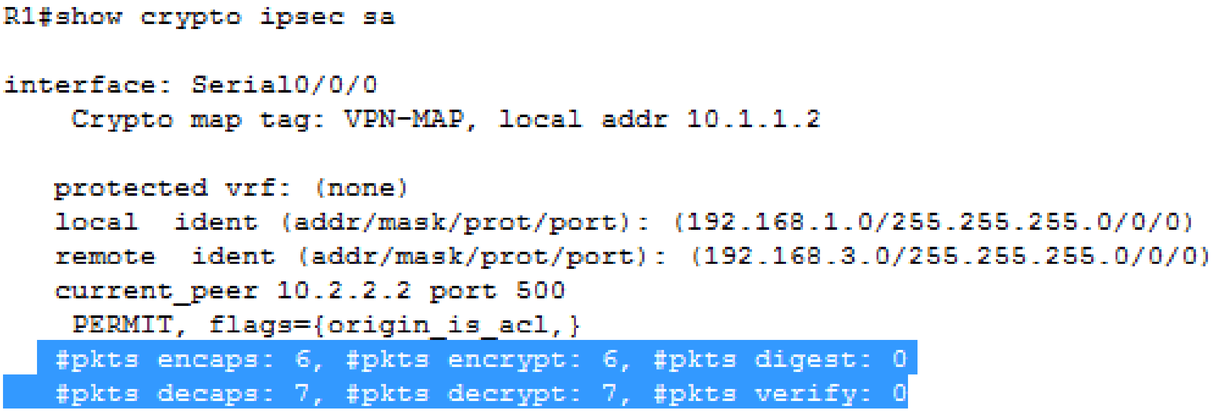
Ping PC-B from PC-A. **Note**: Issuing a ping from router R1 to PC-C or R3 to PC-A is not interesting traffic.



```
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=12ms TTL=126
Reply from 192.168.1.3: bytes=32 time=16ms TTL=126
Reply from 192.168.1.3: bytes=32 time=14ms TTL=126
Reply from 192.168.1.3: bytes=32 time=14ms TTL=126

Ping statistics for 192.168.1.3:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 12ms, Maximum = 16ms, Average = 14ms

C:\>
```

Step 5: Verify the tunnel.

On R1, re-issue the **show crypto ipsec sa**command. Notice that the number of packets has not changed, which verifies that uninteresting traffic is not encrypted.



```
R1#show crypto ipsec sa

interface: Serial0/0/0
    Crypto map tag: VPN-MAP, local addr 10.1.1.2

   protected vrf: (none)
   local  ident (addr/mask/prot/port): (192.168.1.0/255.255.255.0/0/0)
   remote  ident (addr/mask/prot/port): (192.168.3.0/255.255.255.0/0/0)
   current_peer 10.2.2.2 port 500
    PERMIT, flags={origin_is_acl,}
   #pkts encaps: 6, #pkts encrypt: 6, #pkts digest: 0
   #pkts decaps: 7, #pkts decrypt: 7, #pkts verify: 0
```

Practical No. 9

Aim: Configuring ASA basic setting Firewall using CLI.

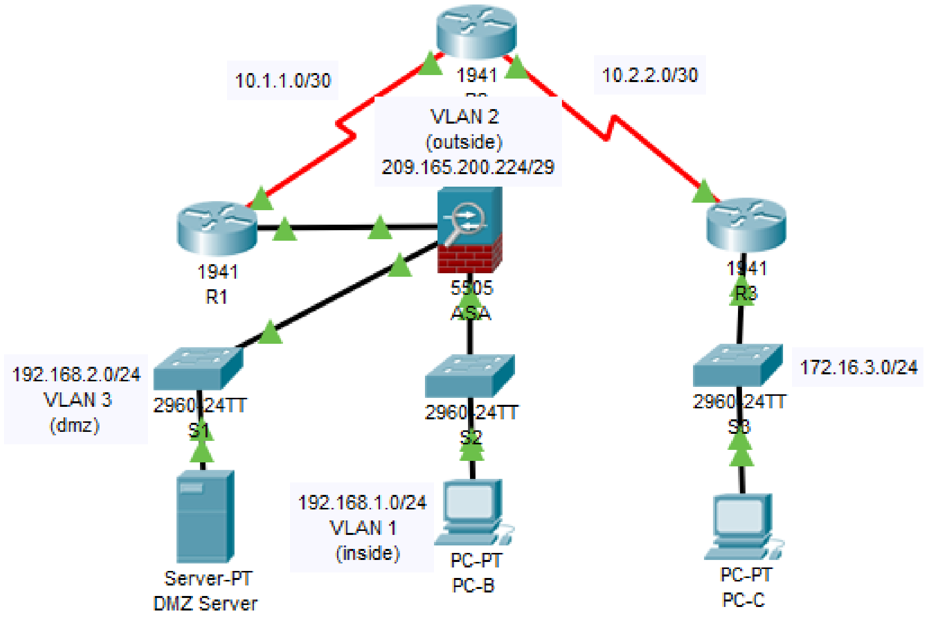
Scenario

Your company has one location connected to an ISP. R1 represents a CPE device managed by the ISP. R2 represents an intermediate Internet router. R3 represents an ISP that connects an administrator from a network management company, who has been hired to remotely manage your network. The ASA is an edge CPE security device that connects the internal corporate network and DMZ to the ISP while providing NAT and DHCP services to inside hosts. The ASA will be configured for management by an administrator on the internal network and by the remote administrator. Layer 3 VLAN interfaces provide access to the three areas created in the activity: Inside, Outside, and DMZ. The ISP assigned the public IP address space of 209.165.200.224/29, which will be used for address translation on the ASA.

All router and switch devices have been preconfigured with the following:

- o Enable password: **ciscoenpa55**
- o Console password: **ciscoconpa55**
- o Admin username and password: **admin/adminpa55**

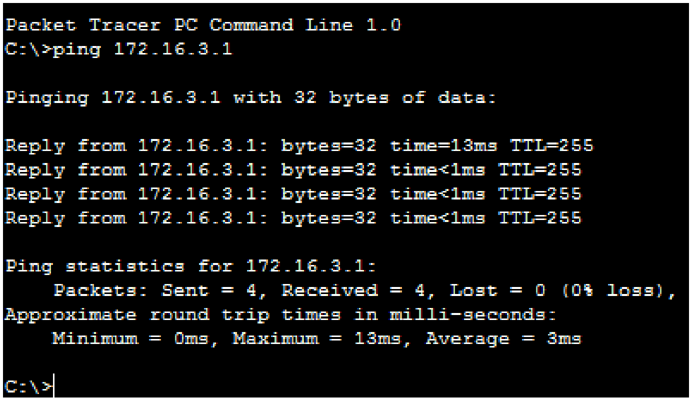
Note: This Packet Tracer activity is not a substitute for the ASA labs. This activity provides additional practice and simulates most of the ASA 5505 configurations. When compared to a real ASA 5505, there may be slight differences in command output or commands that are not yet supported in Packet Tracer.



A. Verify Connectivity and Explore the ASA

Step 1: Verify connectivity.

The ASA is not currently configured. However, all routers, PCs, and the DMZ server are configured. Verify that PC-C can ping any router interface. PC-C is unable to ping the ASA, PC-B, or the DMZ server.



```
Packet Tracer PC Command Line 1.0
C:\>ping 172.16.3.1

Pinging 172.16.3.1 with 32 bytes of data:

Reply from 172.16.3.1: bytes=32 time=13ms TTL=255
Reply from 172.16.3.1: bytes=32 time<1ms TTL=255
Reply from 172.16.3.1: bytes=32 time<1ms TTL=255
Reply from 172.16.3.1: bytes=32 time<1ms TTL=255

Ping statistics for 172.16.3.1:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 0ms, Maximum = 13ms, Average = 3ms

C:\>
```

Step 2: Determine the ASA version, interfaces, and license.

Use the **show version** command to determine various aspects of this ASA device.

ciscoasa>en
Password: (press enter)
ciscoasa#sh version

ciscoasa#

Step 3: Determine the file system and contents of flash memory.

a. Enter privileged EXEC mode. A password has not been set. Press **Enter** when prompted for a password.

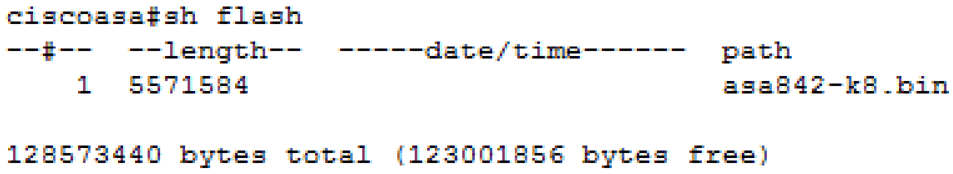
b. Use the **show file system** command to display the ASA file system and determine which prefixes are supported.

ciscoasa#sh file system

File Systems:

Size(b) Free(b) Type Flags Prefixes
* 128573440 123001856 disk rw disk0: flash:

c. Use the **show flash:** or **show disk0:** command to display the contents of flash memory.
ciscoasa#sh flash



```
ciscoasa#sh flash
--#--  --length--  -----date/time------  path
    1  5571584                           asa842-k8.bin

128573440 bytes total (123001856 bytes free)
```

B. Configure ASA Settings and Interface Security Using the CLI

Tip: Many ASA CLI commands are similar to, if not the same, as those used with the Cisco IOS CLI. In addition, the process of moving between configuration modes and submodes is essentially the same.

Step 1: Configure the hostname and domain name.

a. Configure the ASA hostname as CCNAS-ASA.

```
ciscoasa#conf t
ciscoasa(config)#hostname CCNAS-ASA
```

b. Configure the domain name as **ccnasecurity.com**.

```
CCNAS-ASA(config)#domain-name ccnasecurity.com
```

Step 2: Configure the enable mode password.

Use the **enable password** command to change the privileged EXEC mode password to **ciscoenpa55**.

```
CCNAS-ASA(config)#passwd cisco
CCNAS-ASA(config)#enable password ciscoenpa55
```

Step 3: Set the date and time.

Use the **clock set** command to manually set the date and time (this step is not scored).

```
CCNAS-ASA(config)#clock set 21:40:35 january 13 2019
```

Step 4: Configure the inside and outside interfaces.

You will only configure the VLAN 1 (inside) and VLAN 2 (outside) interfaces at this time. The VLAN 3 (dmz) interface will be configured in Part 5 of the activity.

a. Configure a logical VLAN 1 interface for the inside network (192.168.1.0/24) and set the security level to the highest setting of 100.

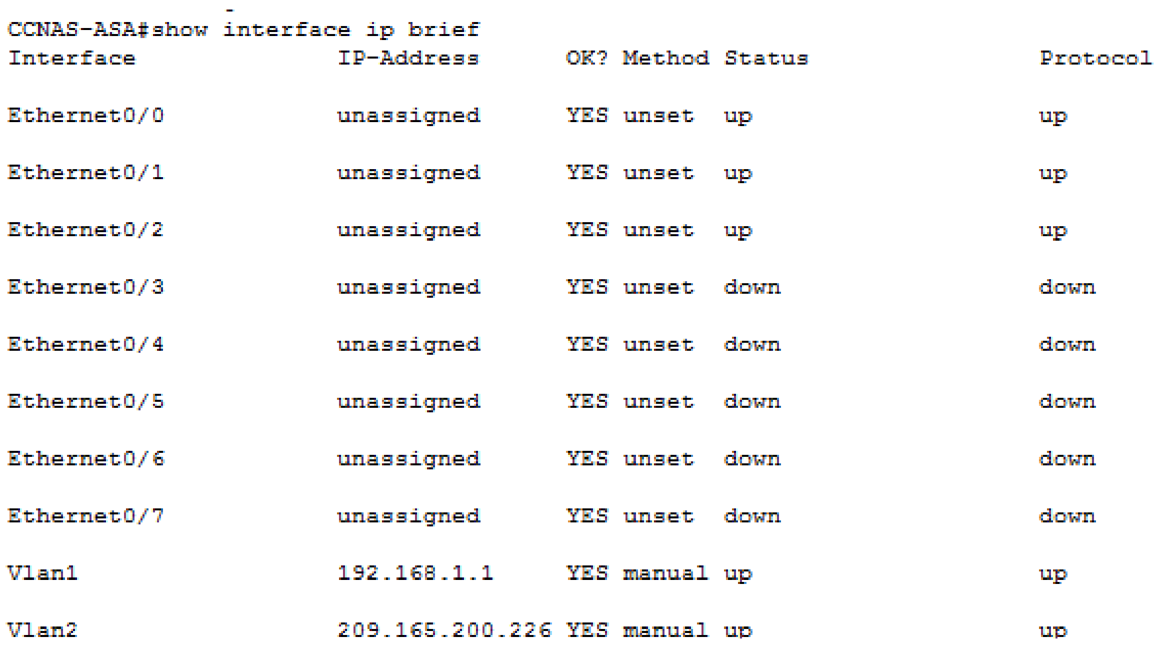
```
CCNAS-ASA(config)#interface vlan 1
CCNAS-ASA(config-if)#nameif inside
CCNAS-ASA(config-if)#ip address 192.168.1.1 255.255.255.0
CCNAS-ASA(config-if)#security-level 100
CCNAS-ASA(config-if)#exit
```

b. Create a logical VLAN 2 interface for the outside network (209.165.200.224/29), set the security level to the lowest setting of 0, and enable the VLAN 2 interface.

```
CCNAS-ASA(config)#interface vlan 2
CCNAS-ASA(config-if)#nameif outside
CCNAS-ASA(config-if)#ip address 209.165.200.226 255.255.255.248
CCNAS-ASA(config-if)#security-level 0
CCNAS-ASA(config-if)#exit
```

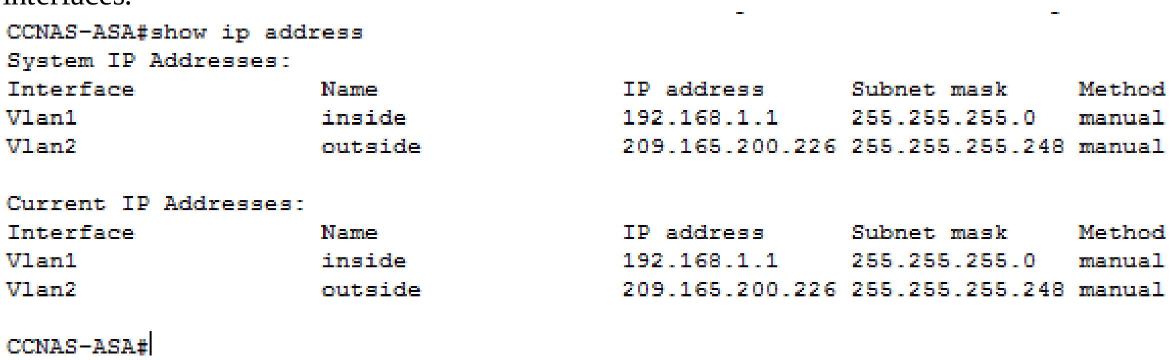
c. Use the following verification commands to check your configurations:

1) Use the **show interface ip brief** command to display the status for all ASA interfaces. **Note**: This command is different from the IOS command **show ip interface brief**. If any of the physical or logical interfaces previously configured are not up/up, troubleshoot as necessary before continuing.



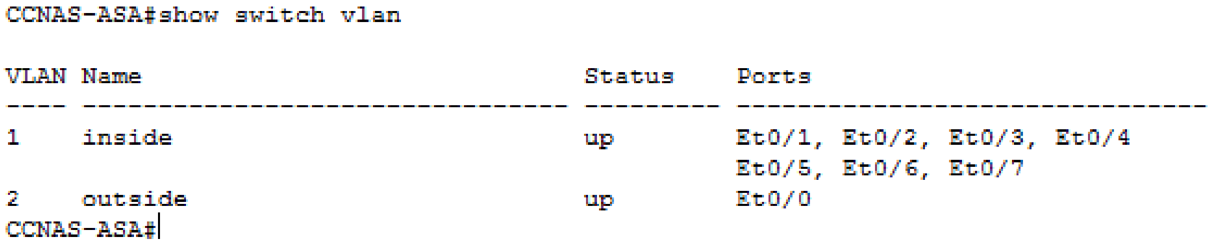
```
CCNAS-ASA#show interface ip brief
Interface                IP-Address      OK? Method Status                Protocol

Ethernet0/0              unassigned      YES unset  up                    up

Ethernet0/1              unassigned      YES unset  up                    up

Ethernet0/2              unassigned      YES unset  up                    up

Ethernet0/3              unassigned      YES unset  down                  down

Ethernet0/4              unassigned      YES unset  down                  down

Ethernet0/5              unassigned      YES unset  down                  down

Ethernet0/6              unassigned      YES unset  down                  down

Ethernet0/7              unassigned      YES unset  down                  down

Vlan1                    192.168.1.1     YES manual up                    up

Vlan2                    209.165.200.226 YES manual up                    up
```

2) Use the **show ip address** command to display the information for the Layer 3 VLAN interfaces.



```
CCNAS-ASA#show ip address
System IP Addresses:
Interface                Name                   IP address      Subnet mask     Method
Vlan1                    inside                 192.168.1.1     255.255.255.0   manual
Vlan2                    outside                209.165.200.226 255.255.255.248 manual

Current IP Addresses:
Interface                Name                   IP address      Subnet mask     Method
Vlan1                    inside                 192.168.1.1     255.255.255.0   manual
Vlan2                    outside                209.165.200.226 255.255.255.248 manual

CCNAS-ASA#
```

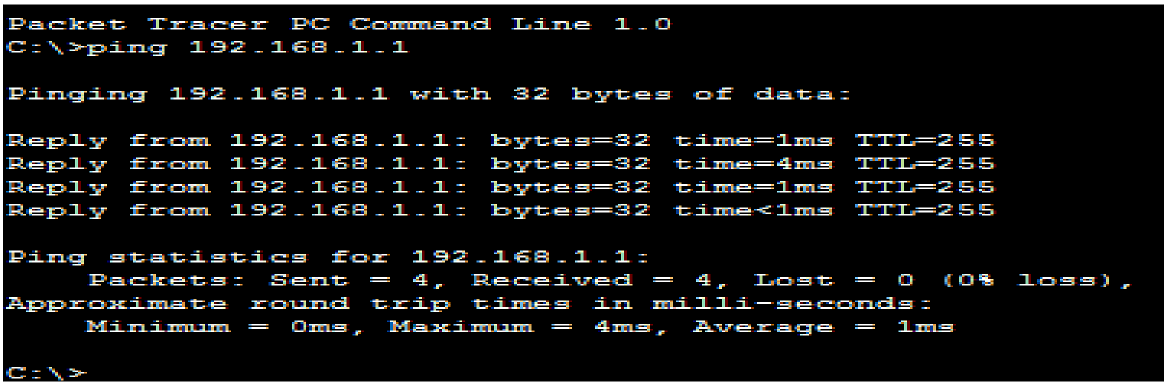
3) Use the **show switch vlan** command to display the inside and outside VLANs configured on the ASA and to display the assigned ports.



```
CCNAS-ASA#show switch vlan

VLAN Name                             Status    Ports
---- -------------------------------- --------- -------------------------------
1    inside                           up        Et0/1, Et0/2, Et0/3, Et0/4
                                                Et0/5, Et0/6, Et0/7
2    outside                          up        Et0/0
CCNAS-ASA#
```

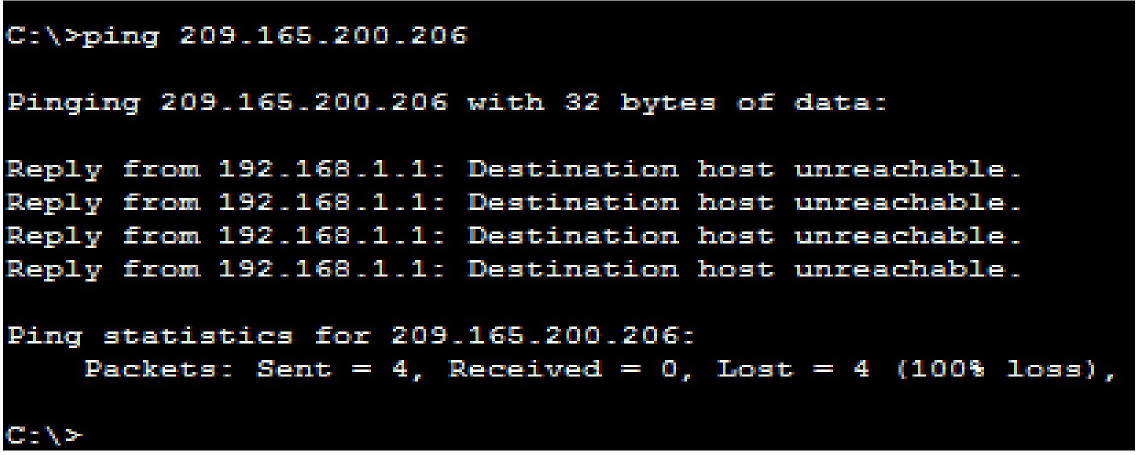
Step 5: Test connectivity to the ASA.

a. You should be able to ping from PC-B to the ASA inside interface address (192.168.1.1). If the pings fail, troubleshoot the configuration as necessary.



```
Packet Tracer PC Command Line 1.0
C:\>ping 192.168.1.1

Pinging 192.168.1.1 with 32 bytes of data:

Reply from 192.168.1.1: bytes=32 time=1ms TTL=255
Reply from 192.168.1.1: bytes=32 time=4ms TTL=255
Reply from 192.168.1.1: bytes=32 time=1ms TTL=255
Reply from 192.168.1.1: bytes=32 time<1ms TTL=255

Ping statistics for 192.168.1.1:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 0ms, Maximum = 4ms, Average = 1ms

C:\>
```

b. From PC-B, ping the VLAN 2 (outside) interface at IP address 209.165.200.226. You should not be able to ping this address.



```
C:\>ping 209.165.200.206

Pinging 209.165.200.206 with 32 bytes of data:

Reply from 192.168.1.1: Destination host unreachable.
Reply from 192.168.1.1: Destination host unreachable.
Reply from 192.168.1.1: Destination host unreachable.
Reply from 192.168.1.1: Destination host unreachable.

Ping statistics for 209.165.200.206:
    Packets: Sent = 4, Received = 0, Lost = 4 (100% loss),

C:\>
```

C. Configure Routing, Address Translation, and Inspection Policy Using the CLI

Step 1: Configure a static default route for the ASA.

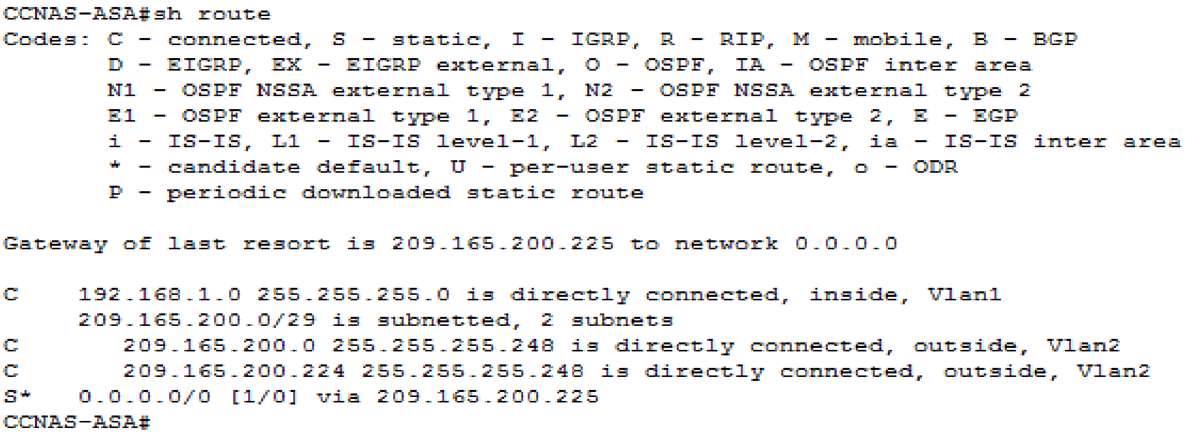
Configure a default static route on the ASA outside interface to enable the ASA to reach external networks.

a. Create a “quad zero” default route using the **route** command, associate it with the ASA outside interface, and point to the R1 G0/0 IP address (209.165.200.225) as the gateway of last resort.

CCNAS-ASA#conf t

CCNAS-ASA(config)#route outside 0.0.0.0 0.0.0.0 209.165.200.225

b. Issue the **show route** command to verify the static default route is in the ASA routing table.



```
CCNAS-ASA#sh route
Codes: C - connected, S - static, I - IGRP, R - RIP, M - mobile, B - BGP
       D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
       N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
       E1 - OSPF external type 1, E2 - OSPF external type 2, E - EGP
       i - IS-IS, L1 - IS-IS level-1, L2 - IS-IS level-2, ia - IS-IS inter area
       * - candidate default, U - per-user static route, o - ODR
       P - periodic downloaded static route

Gateway of last resort is 209.165.200.225 to network 0.0.0.0

C    192.168.1.0 255.255.255.0 is directly connected, inside, Vlan1
     209.165.200.0/29 is subnetted, 2 subnets
C       209.165.200.0 255.255.255.248 is directly connected, outside, Vlan2
C       209.165.200.224 255.255.255.248 is directly connected, outside, Vlan2
S*   0.0.0.0/0 [1/0] via 209.165.200.225
CCNAS-ASA#
```

c. Verify that the ASA can ping the R1 S0/0/0 IP address 10.1.1.1. If the ping is unsuccessful, troubleshoot as necessary.

```
CCNAS-ASA#ping 10.1.1.1

Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 10.1.1.1, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 0/4/21 ms
```

Step 2: Configure address translation using PAT and network objects.

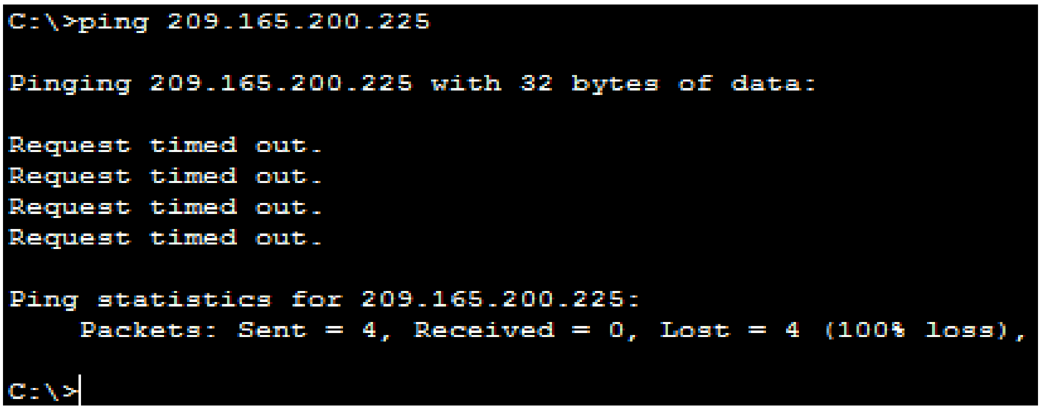
a. Create network object **inside-net** and assign attributes to it using the **subnet** and **nat** commands.

```
CCNAS-ASA#conf t
CCNAS-ASA(config)#object network inside-net
CCNAS-ASA(config-network-object)#subnet 192.168.1.0 255.255.255.0
CCNAS-ASA(config-network-object)#nat (inside,outside) dynamic interface
CCNAS-ASA(config-network-object)#end
```

b. The ASA splits the configuration into the object portion that defines the network to be translated and the actual **nat** command parameters. These appear in two different places in the running configuration. Display the NAT object configuration using the **show run** command.

```
CCNAS-ASA#sh run
CCNAS-ASA#
```

c. From PC-B attempt to ping the R1 G0/0 interface at IP address 209.165.200.225. The pings should fail.



```
C:\>ping 209.165.200.225

Pinging 209.165.200.225 with 32 bytes of data:

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

Ping statistics for 209.165.200.225:
    Packets: Sent = 4, Received = 0, Lost = 4 (100% loss),

C:\>
```

d. Issue the **show nat** command on the ASA to see the translated and untranslated hits. Notice that, of the pings from PC-B, four were translated and four were not. The outgoing pings (echos) were translated and sent to the destination. The returning echo replies were blocked by the firewall policy. You will configure the default inspection policy to allow ICMP in Step 3 of this part of the activity.

```
CCNAS-ASA#sh nat
Auto NAT Policies (Section 2)
1 (inside) to (outside) source dynamic inside-net interface
translate_hits = 4, untranslate_hits = 3
```

Step 3: Modify the default MPF application inspection global service policy.

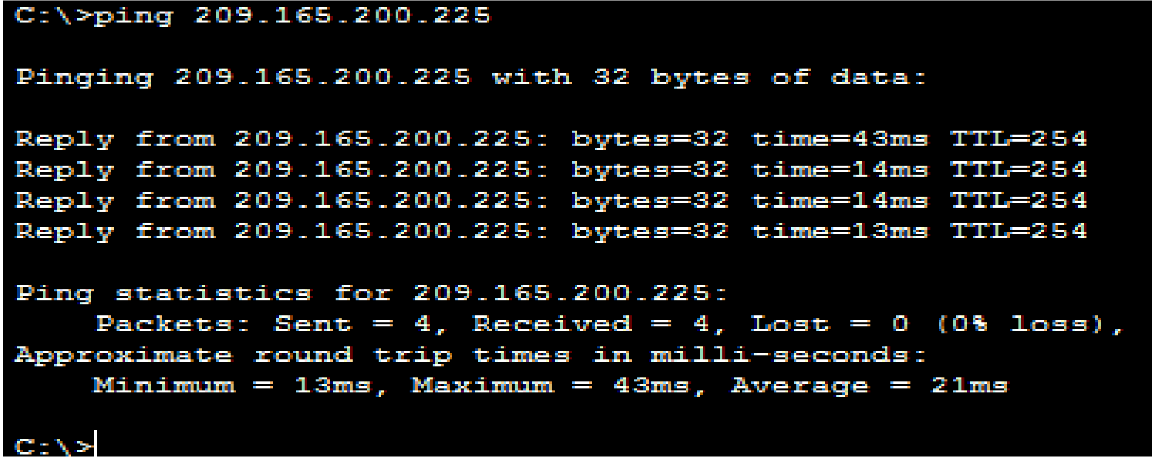
For application layer inspection and other advanced options, the Cisco MPF is available on ASAs.

The Packet Tracer ASA device does not have an MPF policy map in place by default. As a modification, we can create the default policy map that will perform the inspection on inside-to-outside traffic. When configured correctly only traffic initiated from the inside is allowed back in to the outside interface. You will need to add ICMP to the inspection list.

a. Create the class-map, policy-map, and service-policy. Add the inspection of ICMP traffic to the policy map list using the following commands:

```
CCNAS-ASA#conf t
CCNAS-ASA(config)#class-map inspection_default
CCNAS-ASA(config-cmap)#match default-inspection-traffic
CCNAS-ASA(config-cmap)#exit
CCNAS-ASA(config)#policy-map global_policy
CCNAS-ASA(config-pmap)#class inspection_default
CCNAS-ASA(config-pmap-c)#inspect icmp
CCNAS-ASA(config-pmap-c)#exit
CCNAS-ASA(config)#service-policy global_policy global
```

b. From PC-B, attempt to ping the R1 G0/0 interface at IP address 209.165.200.225. The pings should be successful this time because ICMP traffic is now being inspected and legitimate return traffic is being allowed. If the pings fail, troubleshoot your configurations.



```
C:\>ping 209.165.200.225

Pinging 209.165.200.225 with 32 bytes of data:

Reply from 209.165.200.225: bytes=32 time=43ms TTL=254
Reply from 209.165.200.225: bytes=32 time=14ms TTL=254
Reply from 209.165.200.225: bytes=32 time=14ms TTL=254
Reply from 209.165.200.225: bytes=32 time=13ms TTL=254

Ping statistics for 209.165.200.225:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 13ms, Maximum = 43ms, Average = 21ms

C:\>
```

D. Configure DHCP, AAA, and SSH

Step 1: Configure the ASA as a DHCP server.

a. Configure a DHCP address pool and enable it on the ASA inside interface.

```
CCNAS-ASA(config)#dhcpd address 192.168.1.5-192.168.1.36 inside
```

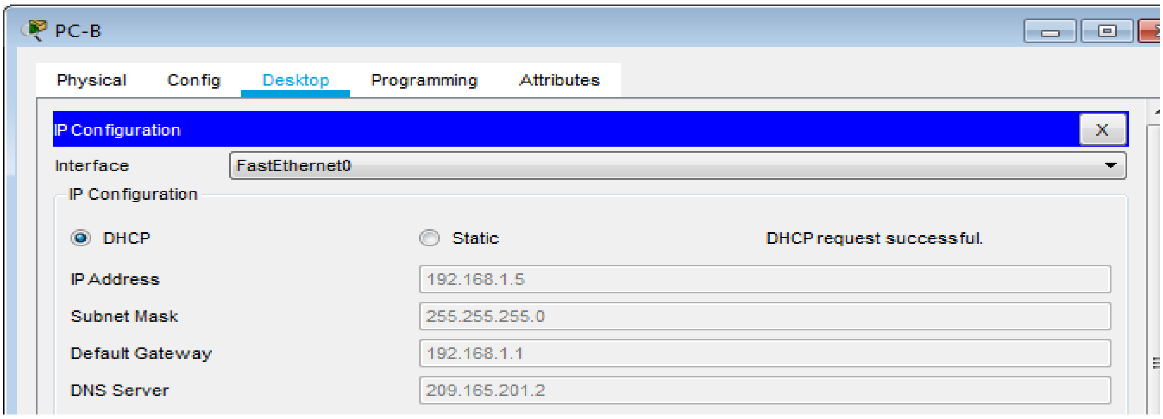
b. (Optional) Specify the IP address of the DNS server to be given to clients.

```
CCNAS-ASA(config)#dhcpd dns 209.165.201.2 interface inside
```

c. Enable the DHCP daemon within the ASA to listen for DHCP client requests on the enabled interface (inside).

```
CCNAS-ASA(config)#dhcpd enable inside
```

d. Change PC-B from a static IP address to a DHCP client, and verify that it receives IP addressing information. Troubleshoot, as necessary to resolve any problems.



IP Configuration

Interface: FastEthernet0

IP Configuration

● DHCP ○ Static DHCP request successful.

IP Address: 192.168.1.5

Subnet Mask: 255.255.255.0

Default Gateway: 192.168.1.1

DNS Server: 209.165.201.2

Step 2: Configure AAA to use the local database for authentication.

a. Define a local user named **admin** by entering the **username** command. Specify a password of **adminpa55**.

```
CCNAS-ASA#conf t
CCNAS-ASA(config)#username admin password adminpa55
```

b. Configure AAA to use the local ASA database for SSH user authentication.

```
CCNAS-ASA(config)#aaa authentication ssh console LOCAL
```

Step 3: Configure remote access to the ASA.

The ASA can be configured to accept connections from a single host or a range of hosts on the inside or outside network. In this step, hosts from the outside network can only use SSH to communicate with the ASA. SSH sessions can be used to access the ASA from the inside network.

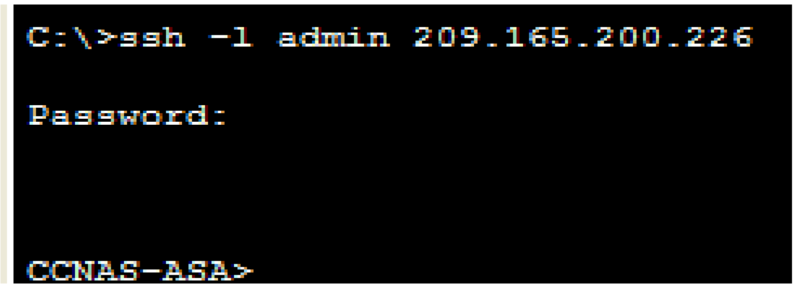
a. Generate an RSA key pair, which is required to support SSH connections. Because the ASA device has RSA keys already in place, enter **no** when prompted to replace them.

```
CCNAS-ASA(config)#crypto key generate rsa modulus 1024
WARNING: You have a RSA keypair already defined named <Default-RSA-Key>.
Do you really want to replace them? [yes/no]: no
ERROR: Failed to create new RSA keys named <Default-RSA-Key>
CCNAS-ASA(config)#
```

b. Configure the ASA to allow SSH connections from any host on the inside network (192.168.1.0/24) and from the remote management host at the branch office (172.16.3.3) on the outside network. Set the SSH timeout to 10 minutes (the default is 5 minutes).

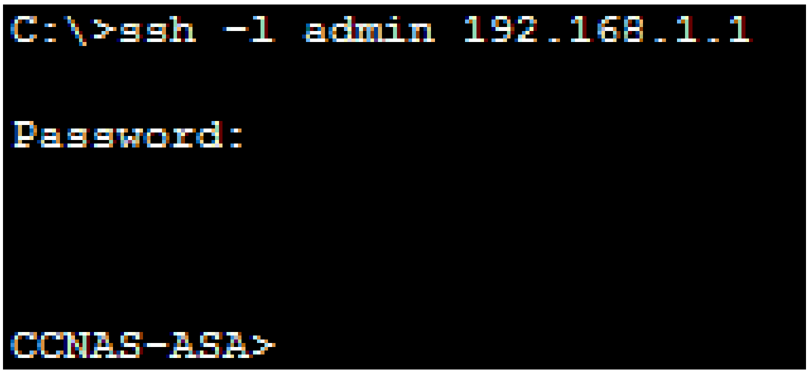
```
CCNAS-ASA(config)#ssh 192.168.1.0 255.255.255.0 inside
CCNAS-ASA(config)#ssh 172.16.3.3 255.255.255.255 outside
CCNAS-ASA(config)#ssh timeout 10
```

c. Establish an SSH session from PC-C to the ASA (209.165.200.226). Troubleshoot if it is not successful.



```
C:\>ssh -l admin 209.165.200.226

Password:


CCNAS-ASA>
```

d. Establish an SSH session from PC-B to the ASA (192.168.1.1). Troubleshoot if it is not successful.



```
C:\>ssh -l admin 192.168.1.1

Password:


CCNAS-ASA>
```

E. Configure a DMZ, Static NAT, and ACLs

R1 G0/0 and the ASA outside interface already use 209.165.200.225 and .226, respectively. You will use public address 209.165.200.227 and static NAT to provide address translation access to the server.

Step 1: Configure the DMZ interface VLAN 3 on the ASA.

a. Configure DMZ VLAN 3, which is where the public access web server will reside. Assign it IP address 192.168.2.1/24, name it **dmz**, and assign it a security level of 70. Because theserver does not need to initiate communication with the inside users, disable forwarding to interface VLAN 1.

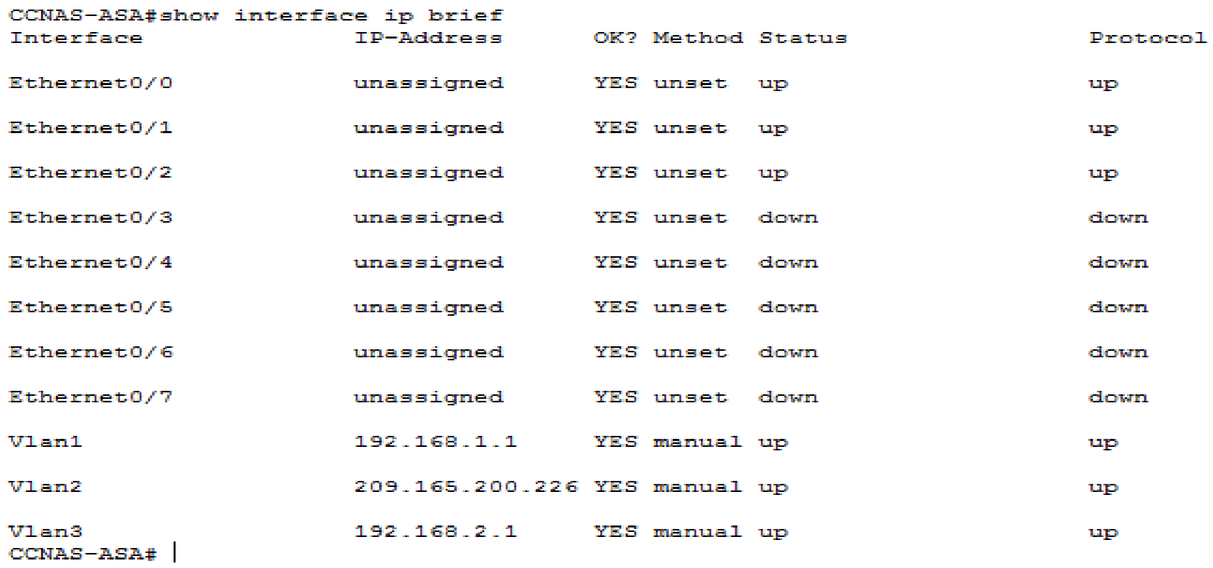
```
CCNAS-ASA(config)#interface vlan 3
CCNAS-ASA(config-if)#ip address 192.168.2.1 255.255.255.0
CCNAS-ASA(config-if)#no forward interface vlan 1
CCNAS-ASA(config-if)#nameif dmz
INFO: Security level for "dmz" set to 0 by default.
CCNAS-ASA(config-if)#security-level 70
```

b. Assign ASA physical interface E0/2 to DMZ VLAN 3 and enable the interface.

```
CCNAS-ASA(config-if)#interface Ethernet0/2
CCNAS-ASA(config-if)#switchport access vlan 3
```

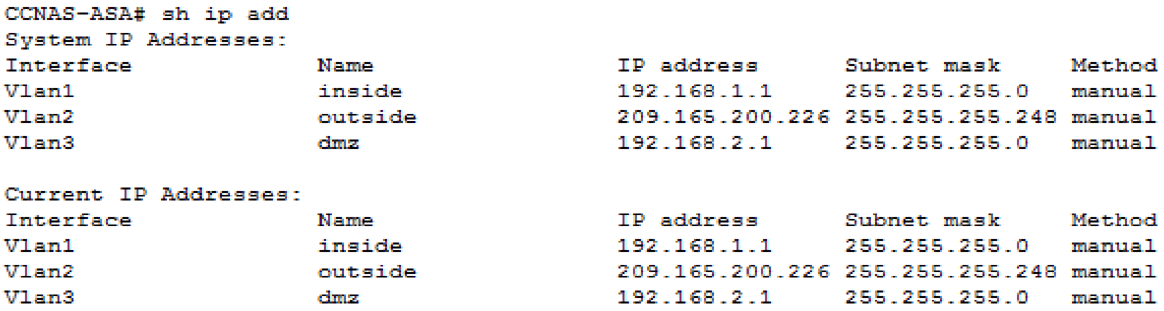
c. Use the following verification commands to check your configurations:

1) Use the **show interface ip brief** command to display the status for all ASA interfaces.



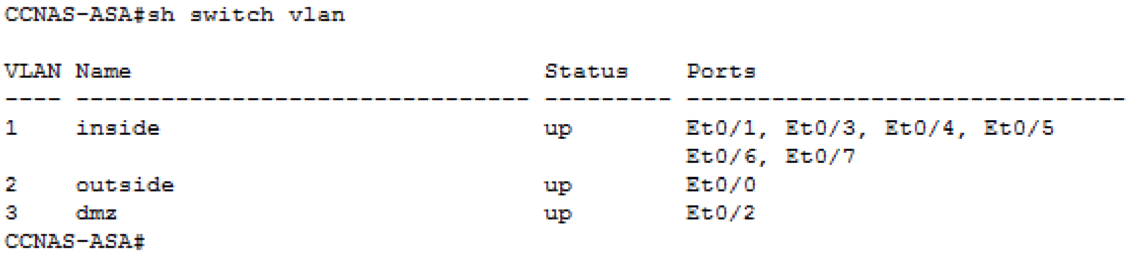
```
CCNAS-ASA#show interface ip brief
Interface              IP-Address      OK? Method Status                 Protocol

Ethernet0/0            unassigned      YES unset  up                     up

Ethernet0/1            unassigned      YES unset  up                     up

Ethernet0/2            unassigned      YES unset  up                     up

Ethernet0/3            unassigned      YES unset  down                   down

Ethernet0/4            unassigned      YES unset  down                   down

Ethernet0/5            unassigned      YES unset  down                   down

Ethernet0/6            unassigned      YES unset  down                   down

Ethernet0/7            unassigned      YES unset  down                   down

Vlan1                  192.168.1.1     YES manual up                     up

Vlan2                  209.165.200.226 YES manual up                     up

Vlan3                  192.168.2.1     YES manual up                     up
CCNAS-ASA# |
```

2) Use the **show ip address** command to display the information for the Layer 3 VLAN interfaces.



```
CCNAS-ASA# sh ip add
System IP Addresses:
Interface              Name                 IP address      Subnet mask     Method
Vlan1                  inside               192.168.1.1     255.255.255.0   manual
Vlan2                  outside              209.165.200.226 255.255.255.248 manual
Vlan3                  dmz                  192.168.2.1     255.255.255.0   manual

Current IP Addresses:
Interface              Name                 IP address      Subnet mask     Method
Vlan1                  inside               192.168.1.1     255.255.255.0   manual
Vlan2                  outside              209.165.200.226 255.255.255.248 manual
Vlan3                  dmz                  192.168.2.1     255.255.255.0   manual
```

3) Use the **show switch vlan** command to display the inside and outside VLANs configured on the ASA and to display the assigned ports.



```
CCNAS-ASA#sh switch vlan

VLAN Name                             Status    Ports
---- -------------------------------- --------- -------------------------------
1    inside                           up        Et0/1, Et0/3, Et0/4, Et0/5
                                                Et0/6, Et0/7
2    outside                          up        Et0/0
3    dmz                              up        Et0/2
CCNAS-ASA#
```

Step 2: Configure static NAT to the DMZ server using a network object.

Configure a network object named **dmz-server** and assign it the static IP address of the DMZ server (192.168.2.3). While in object definition mode, use the **nat** command to specify that this object is used to translate a DMZ address to an outside address using static NAT, and specify a public translated address of 209.165.200.227.

```
CCNAS-ASA#conf t
CCNAS-ASA(config)#object network dmz-server
CCNAS-ASA(config-network-object)#host 192.168.2.3
CCNAS-ASA(config-network-object)#nat (dmz,outside) static 209.165.200.227
CCNAS-ASA(config-network-object)#exit
```

Step 3: Configure an ACL to allow access to the DMZ server from the Internet.

Configure a named access list **OUTSIDE-DMZ** that permits the TCP protocol on port 80 from any external host to the internal IP address of the DMZ server. Apply the access list to the ASA outside interface in the "IN" direction.

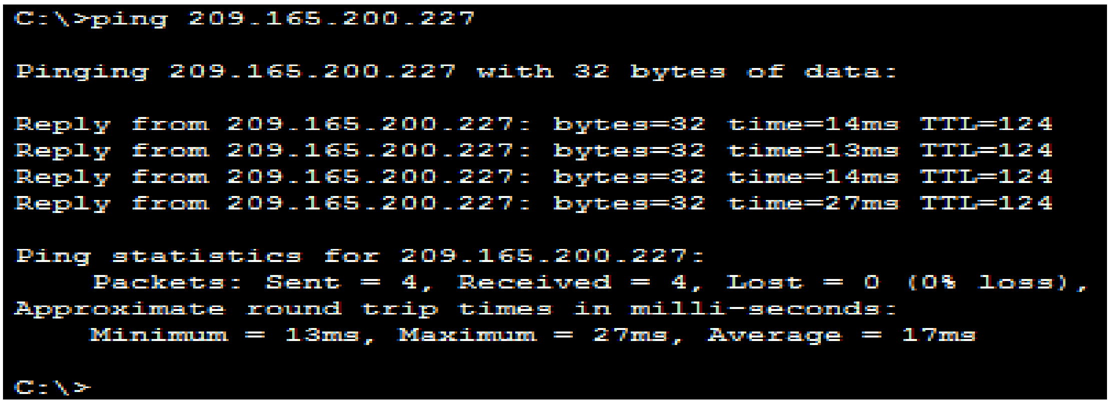
```
CCNAS-ASA#conf t
CCNAS-ASA(config)#access-list OUTSIDE-DMZ permit icmp any host 192.168.2.3
CCNAS-ASA(config)#access-list OUTSIDE-DMZ permit tcp any host 192.168.2.3 eq 80
CCNAS-ASA(config)#access-group OUTSIDE-DMZ in interface outside
```

Note: Unlike IOS ACLs, the ASA ACL permit statement must permit access to the internal private DMZ address. External hosts access the server using its public static NAT address, the ASA translates it to the internal host IP address, and then applies the ACL.

Step 4: Test access to the DMZ server.
At the time this Packet Tracer activity was created, the ability to successfully test outside access to the DMZ web server was not in place; therefore, successful testing is not required.

```
User Access Verification

Password:

R2>en
Password:
R2#conf t
R2(config)#int lo0
%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 add 172.30.1.1 255.255.255.0
R2(config-if)#end

R2#ping 209.165.200.227

Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 209.165.200.227, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 11/17/24 ms

CCNAS-ASA#clear nat counters
```

From PC-C



```
C:\>ping 209.165.200.227

Pinging 209.165.200.227 with 32 bytes of data:

Reply from 209.165.200.227: bytes=32 time=14ms TTL=124
Reply from 209.165.200.227: bytes=32 time=13ms TTL=124
Reply from 209.165.200.227: bytes=32 time=14ms TTL=124
Reply from 209.165.200.227: bytes=32 time=27ms TTL=124

Ping statistics for 209.165.200.227:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 13ms, Maximum = 27ms, Average = 17ms

C:\>
```

```
CCNAS-ASA#sh nat
Auto NAT Policies (Section 2)
1 (dmz) to (outside) source static dmz-server 209.165.200.227
translate_hits = 0, untranslate_hits = 0
2 (inside) to (outside) source dynamic inside-net interface
translate_hits = 0, untranslate_hits = 0
```

```
CCNAS-ASA#sh xlate
1 in use, 1 most used
Flags: D - DNS, e - extended, I - identity, i - dynamic, r - portmap, s - static, T - twice, N -
net-to-net
NAT from dmz:192.168.2.3/32 to outside:209.165.200.227/32 flags s idle 00:13:58, timeout
0:00:00
```

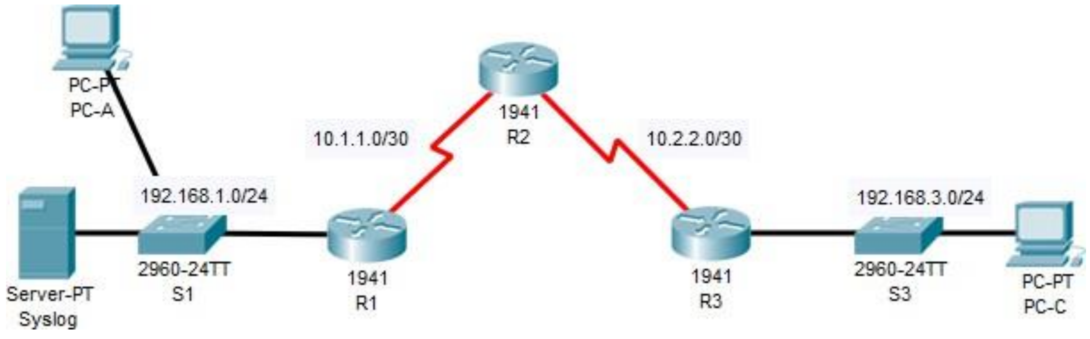
Practical No. 10

Aim: Configuring IOS Intrusion Detection System(IDS).

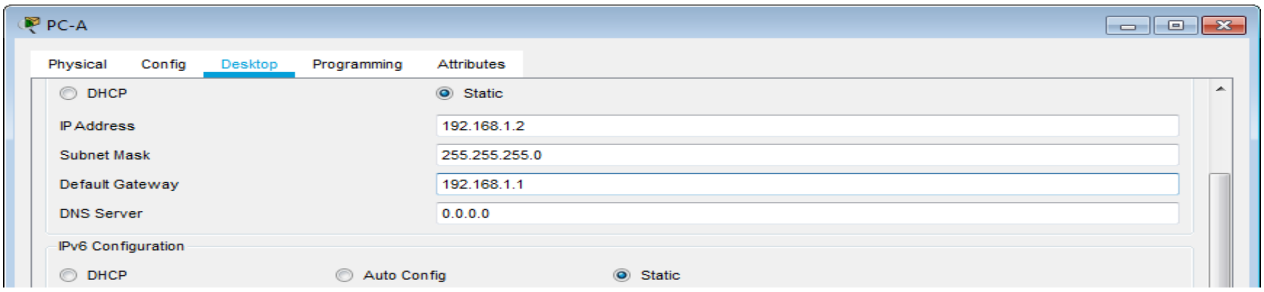
Background / Scenario

Your task is to enable IPS on R1 to scan traffic entering the 192.168.1.0 network. The server labeled Syslog is used to log IPS messages. You must configure the router to identify the syslog server to receive logging messages. Displaying the correct time and date in syslog messages is vital when using syslog to monitor the network. Set the clock and configure the timestamp service for logging on the routers. Finally, enable IPS to produce an alert and drop ICMP echo reply packets inline. The server and PCs have been preconfigured. The routers have also been preconfigured with the following:

- Enable password: ciscoenpa55
- Console password: ciscoconpa55
- SSH username and password: SSHadmin / ciscosshpa55



Take topology of practical 4. Change name of PC-A to Syslog and assign IP add 192.168.1.50. Add a new PC above switch S1, name it as PC-A and assign IP address 192.168.1.2. Connect PC-A from port FastEthernet 0 to switch S1 port FastEthernet 0/1. Change the IP address of PC-C to 192.168.3.2.



○ DHCP ● Static

IP Address 192.168.1.2

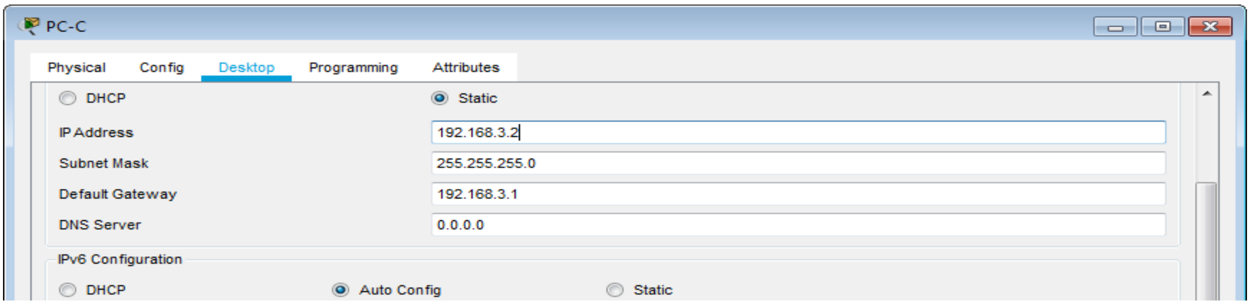
Subnet Mask 255.255.255.0

Default Gateway 192.168.1.1

DNS Server 0.0.0.0

IPv6 Configuration

○ DHCP ○ Auto Config ● Static



○ DHCP ● Static

IP Address 192.168.3.2

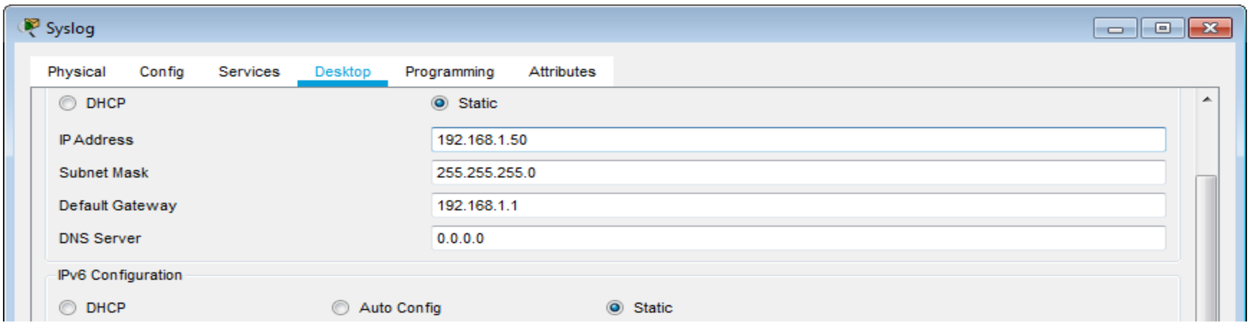
Subnet Mask 255.255.255.0

Default Gateway 192.168.3.1

DNS Server 0.0.0.0

IPv6 Configuration

○ DHCP ● Auto Config ○ Static



○ DHCP ● Static

IP Address 192.168.1.50

Subnet Mask 255.255.255.0

Default Gateway 192.168.1.1

DNS Server 0.0.0.0

IPv6 Configuration

○ DHCP ○ Auto Config ● Static

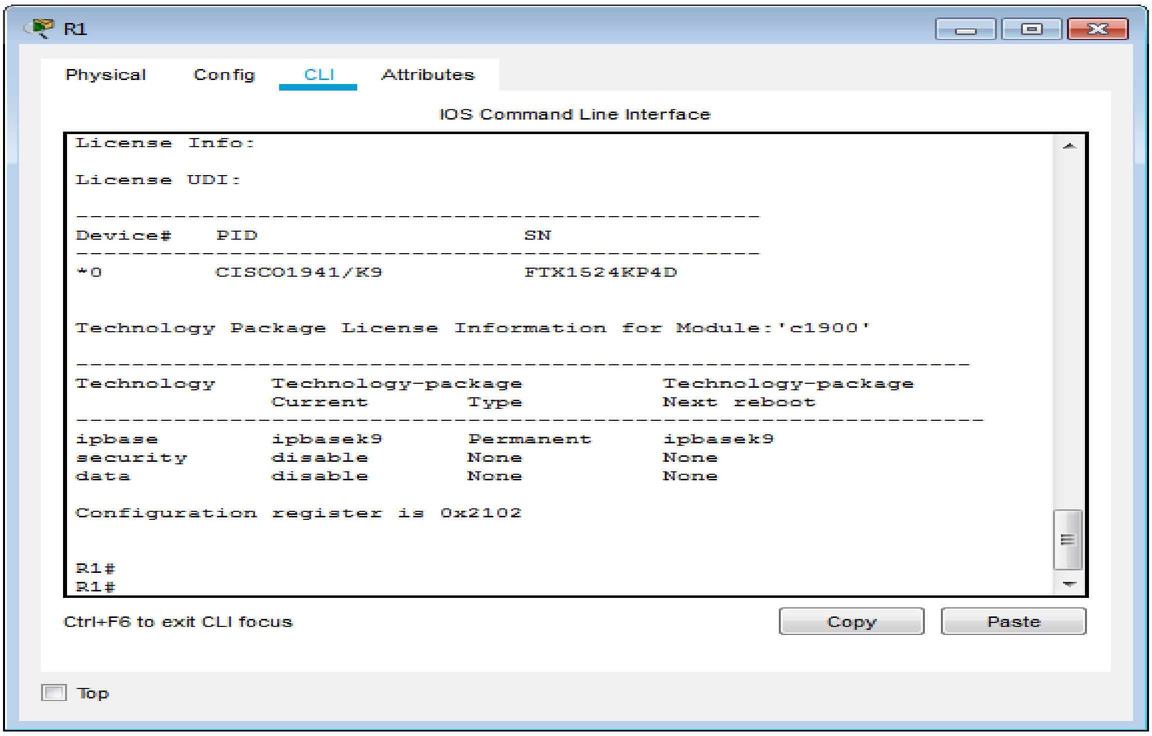
A. Enable IOS IPS

Note: Within Packet Tracer, the routers already have the signature files imported and in place. They are the default xml files in flash. For this reason, it is not necessary to configure the public crypto key and complete a manual import of the signature files.

Step 1: Enable the Security Technology package.

a. On R1, issue the show version command to view the Technology Package license information.

User Access Verification
Password:
R1>en
Password:
R1#sh version



```
License Info:

License UDI:

-------------------------------------------------
Device#   PID                 SN
-------------------------------------------------
*0        CISCO1941/K9        FTX1524KP4D


Technology Package License Information for Module:'c1900'

----------------------------------------------------------------
Technology    Technology-package         Technology-package
              Current       Type         Next reboot
-----------------------------------------------------------------
ipbase        ipbasek9      Permanent    ipbasek9
security      disable       None         None
data          disable       None         None

Configuration register is 0x2102


R1#
R1#
```

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

R1#conf t
R1(config)#license boot module c1900 technology-package securityk9

c. Accept the end user license agreement.

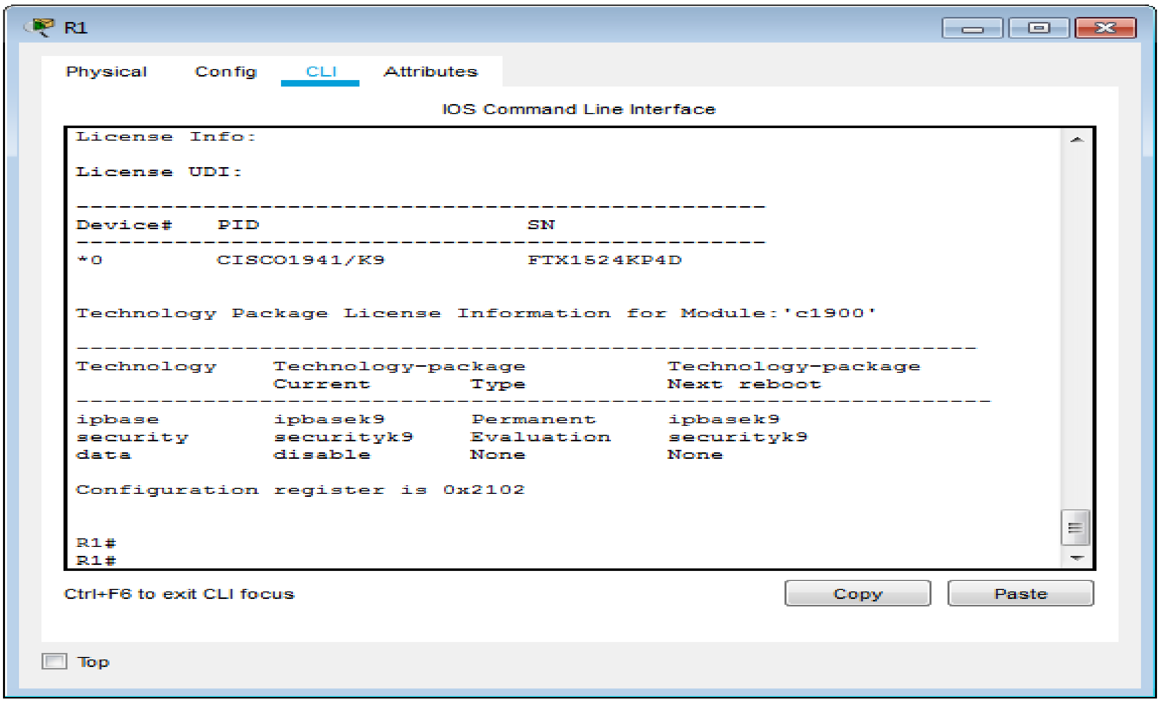
ACCEPT? [yes/no]: yes

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

```
R1(config)#exit
R1#copy run start
Destination filename [startup-config]?
Building configuration...
[OK]

R1#reload
User Access Verification
Password:
%LINEPROTO-5-UPDOWN: Line protocol on Interface Serial0/0/0, changed state to up

R1>en
Password:
```

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

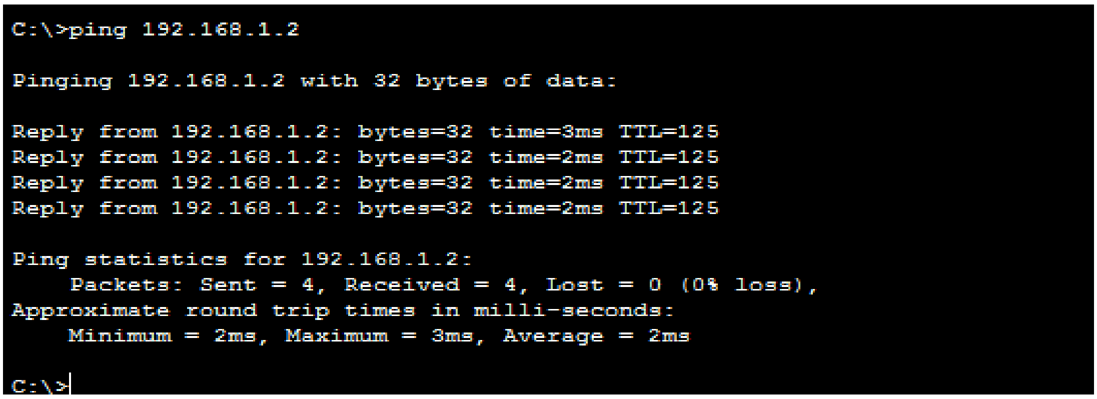
R1#sh version



```
License Info:

License UDI:

-------------------------------------------------
Device#   PID                  SN
-------------------------------------------------
*0        CISCO1941/K9         FTX1524KP4D

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


R1#
R1#
```

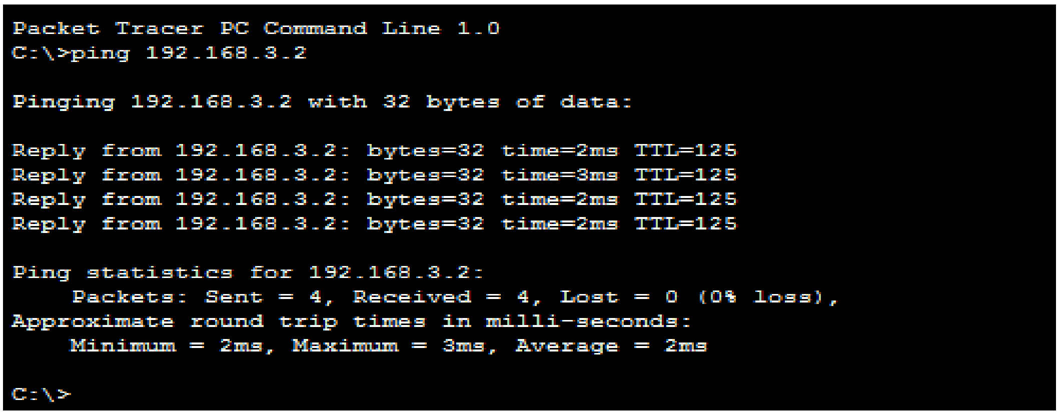
Step 2: Verify network connectivity.

a. Ping from PC-C to PC-A. The ping should be successful.



```
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=3ms 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 = 3ms, Average = 2ms

C:\>
```

b. Ping from PC-A to PC-C. The ping should be successful.



```
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=3ms 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 = 3ms, Average = 2ms

C:\>
```

Step 3: Create an IOS IPS configuration directory in flash.

On R1, create a directory in flash using the mkdir command. Name the directory ipsdir

```
R1#mkdir ipsdir
Create directory filename [ipsdir]?
Created dir flash:ipsdir
```

Step 4: Configure the IPS signature storage location.

On R1, configure the IPS signature storage location to be the directory you just created.

```
R1#conf t
R1(config)#ip ips config location flash:ipsdir
```

Step 5: Create an IPS rule.

On R1, create an IPS rule name using the ip ips name name command in global configuration mode. Name the IPS rule iosips.

```
R1(config)#ip ips name iosips
```

Step 6: Enable logging.

IOS IPS supports the use of syslog to send event notification. Syslog notification is enabled by default. If logging console is enabled, IPS syslog messages display.

a. Enable syslog if it is not enabled.

```
R1(config)#ip ips notify log
```

b. Verify that the timestamp service for logging is enabled on the router using the show run command. Enable the timestamp service if it is not enabled

```
R1(config)#service timestamp log datetime msec
```

c. Send log messages to the syslog server at IP address 192.168.1.50.

```
R1(config)#logging host 192.168.1.50
```

Step 7: Configure IOS IPS to use the signature categories.

Retire the all signature category with the retired true command (all signatures within the signature release). Unretire the IOS_IPS Basic category with the retired false command

```
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](enter)
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 8: Apply the IPS rule to an interface.

Apply the IPS rule to an interface with the ip ips name direction command in interface configuration mode. Apply the rule outbound on the G0/1 interface of R1. After you enable IPS, some log messages will be sent to the console line indicating that the IPS engines are being initialized.

```
R1(config)#int g0/1
R1(config-if)#ip ips iosips out
R1(config-if)#
*Mar 01, 00:04:52.044: %IPS-6-ENGINE_BUILDS_STARTED: 00:04:52 UTC Mar 01 1993
*Mar 01, 00:04:52.044: %IPS-6-ENGINE_BUILDING: atomic-ip - 3 signatures - 1 of 13
engines
*Mar 01, 00:04:52.044: %IPS-6-ENGINE_READY: atomic-ip - build time 8 ms - packets for
this engine will be scanned
*Mar 01, 00:04:52.044: %IPS-6-ALL_ENGINE_BUILDS_COMPLETE: elapsed time 8 ms
R1(config-if)#exit
```

B. Modify the Signature

Step 1: Change the event-action of a signature.

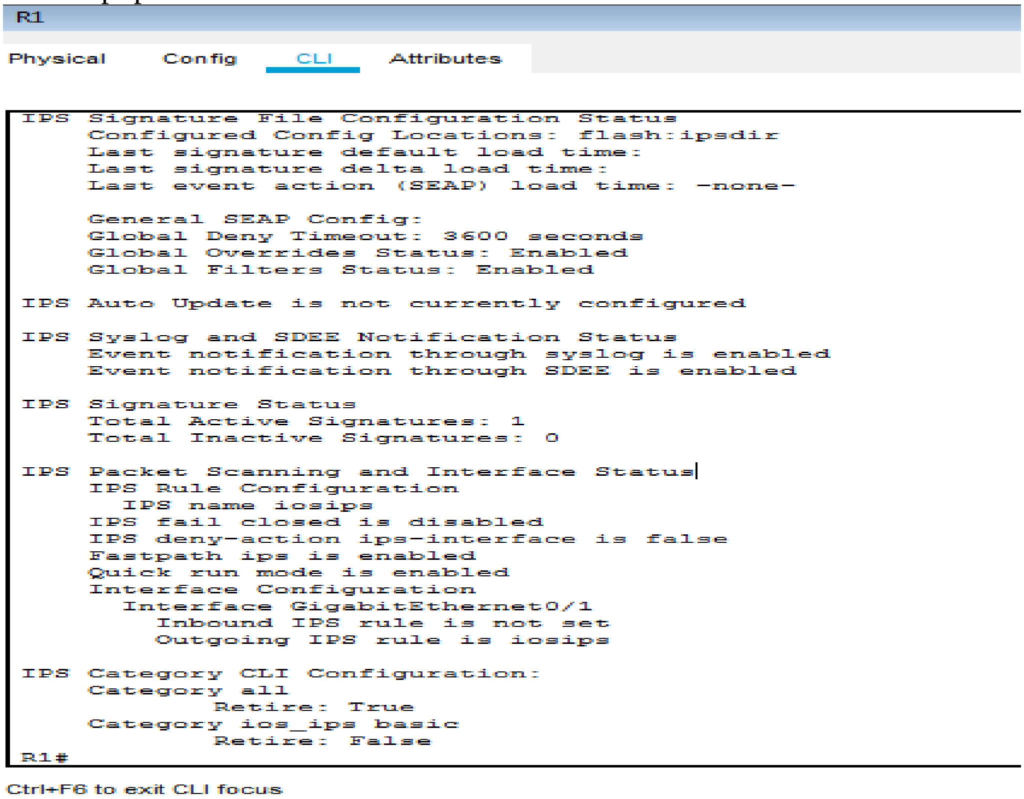
Un-retire the echo request signature (signature 2004, subsig ID 0), enable it, and change the signature action to alert and drop.

```
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](enter)
%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
R1#
*Mar 01, 00:08:18.088: SYS-5-CONFIG_I: Configured from console by console
*Mar 01, 00:08:18.088: %SYS-6-LOGGINGHOST_STARTSTOP: Logging to host 192.168.1.50 port 514 started - CLI initiated

Step 2: Use show commands to verify IPS.

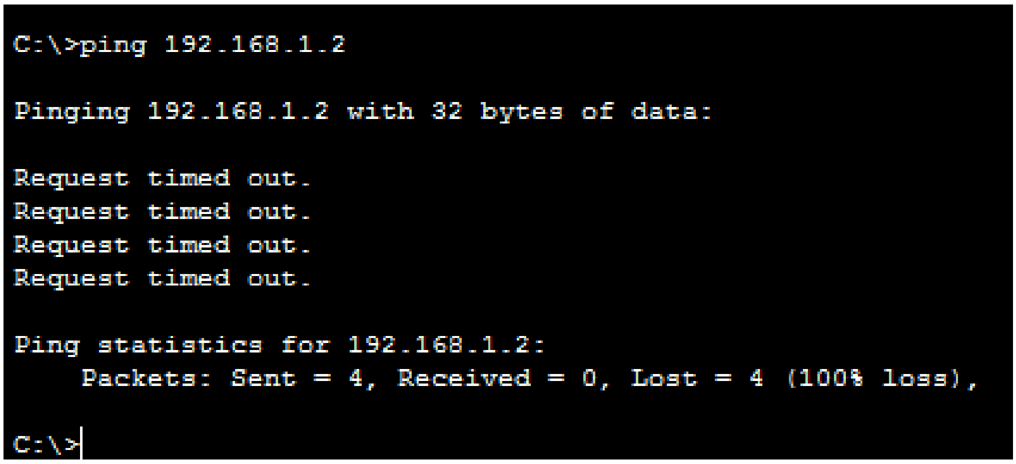
R1#sh ip ips all



```
IPS Signature File Configuration Status
    Configured Config Locations: flash:ipsdir
    Last signature default load time:
    Last signature delta load time:
    Last event action (SEAP) load time: -none-

    General SEAP Config:
    Global Deny Timeout: 3600 seconds
    Global Overrides Status: Enabled
    Global Filters Status: Enabled

IPS Auto Update is not currently configured

IPS Syslog and SDEE Notification Status
    Event notification through syslog is enabled
    Event notification through SDEE is enabled

IPS Signature Status
    Total Active Signatures: 1
    Total Inactive Signatures: 0

IPS Packet Scanning and Interface Status
    IPS Rule Configuration
      IPS name iosips
    IPS fail closed is disabled
    IPS deny-action ips-interface is false
    Fastpath ips is enabled
    Quick run mode is enabled
    Interface Configuration
      Interface GigabitEthernet0/1
        Inbound IPS rule is not set
        Outgoing IPS rule is iosips

IPS Category CLI Configuration:
    Category all
          Retire: True
    Category ios_ips basic
          Retire: False
R1#
```

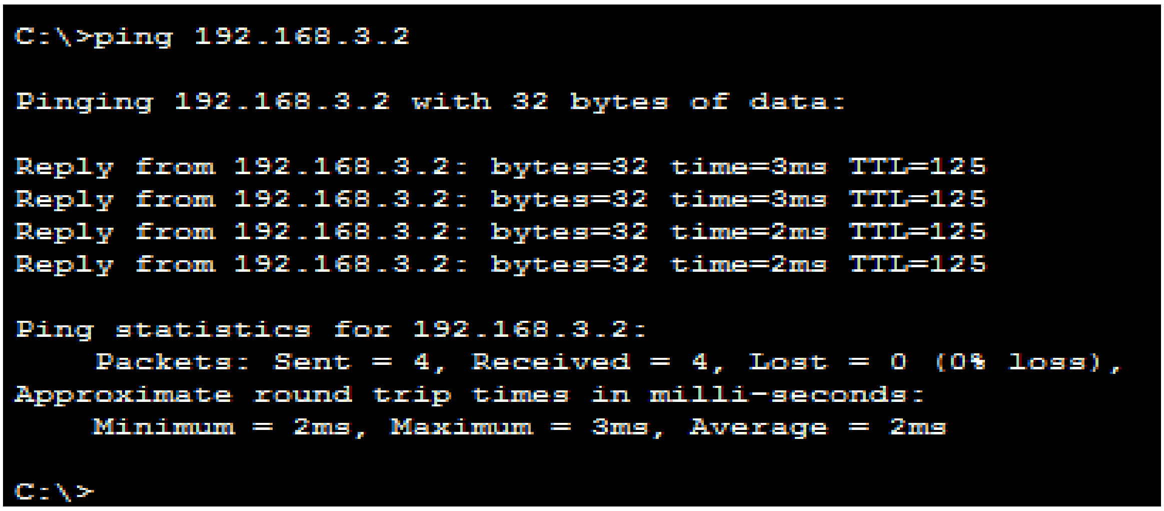
Step 3: Verify that IPS is working properly.

a. From PC-C, attempt to ping PC-A. The pings should fail. This is because the IPS rule for event-action of an echo request was set to “deny- packetinline”.



```
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:\>
```

b. From PC-A, attempt to ping PC-C. 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.



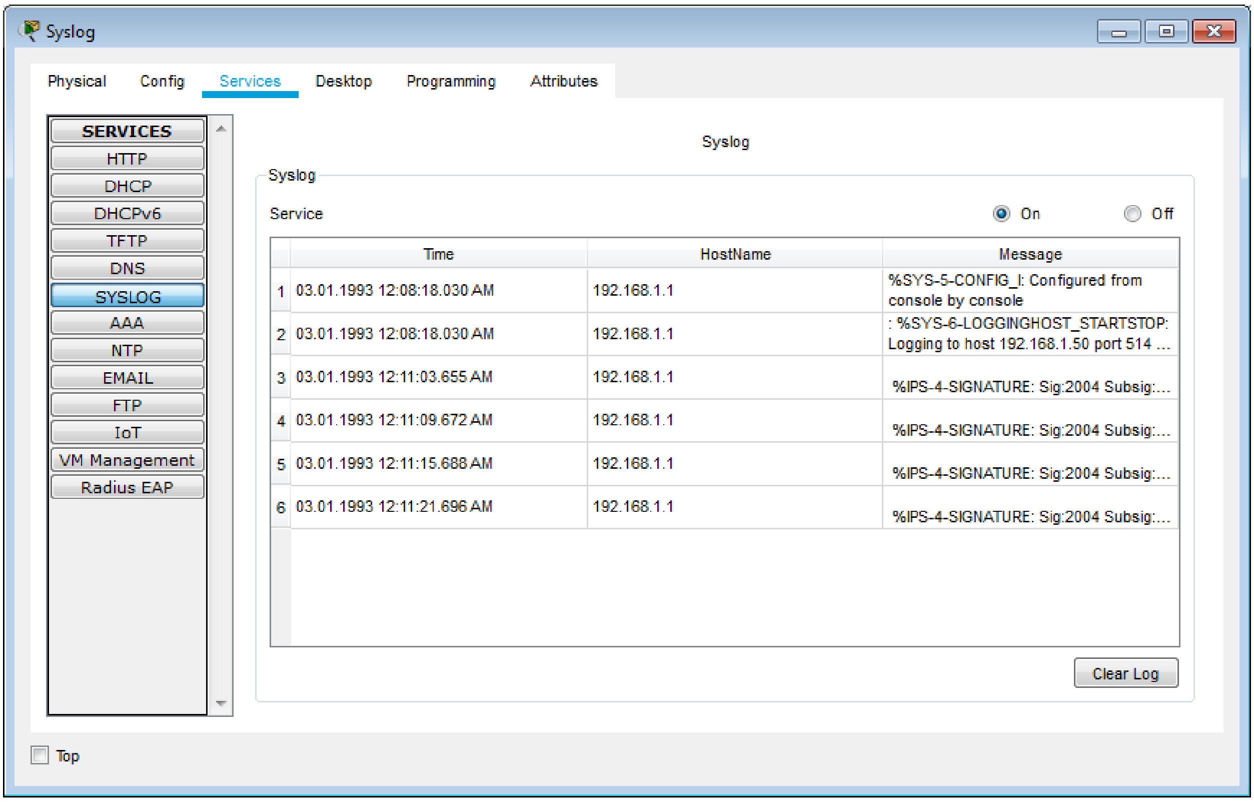
```
C:\>ping 192.168.3.2

Pinging 192.168.3.2 with 32 bytes of data:

Reply from 192.168.3.2: bytes=32 time=3ms TTL=125
Reply from 192.168.3.2: bytes=32 time=3ms 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 = 3ms, Average = 2ms

C:\>
```

Step 4: 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

Syslog

Service ● On ○ Off

	Time	HostName	Message
1	03.01.1993 12:08:18.030 AM	192.168.1.1	%SYS-5-CONFIG_I: Configured from console by console
2	03.01.1993 12:08:18.030 AM	192.168.1.1	: %SYS-6-LOGGINGHOST_STARTSTOP: Logging to host 192.168.1.50 port 514 ...
3	03.01.1993 12:11:03.655 AM	192.168.1.1	%IPS-4-SIGNATURE: Sig:2004 Subsig:...
4	03.01.1993 12:11:09.672 AM	192.168.1.1	%IPS-4-SIGNATURE: Sig:2004 Subsig:...
5	03.01.1993 12:11:15.688 AM	192.168.1.1	%IPS-4-SIGNATURE: Sig:2004 Subsig:...
6	03.01.1993 12:11:21.696 AM	192.168.1.1	%IPS-4-SIGNATURE: Sig:2004 Subsig:...