Lab 06
Configuring Extended and IPv6 ACLs

Name: Syed Asghar Abbas Zaidi	Student ID: 07201

6.1 Objective

The Objectives of this lab are:

- Configure, Apply and Verify an Extended Numbered ACL
- Verify connectivity among devices before firewall configuration.
- Use ACLs to ensure remote access to the routers is available only from management station PC-C.
- Configure and Verify ACLs on R1 and R3 to mitigate attacks.
- Configure, Apply, and Verify an IPv6 ACL

6.2 Introduction

In this scenario, devices on one LAN are allowed to remotely access devices in another LAN using the SSH protocol. Besides ICMP, all traffic from other networks is denied.

The switches and router have also been pre-configured with the following:

- Enable secret password: **ciscoenpa55**
- Console password: **ciscoconpa55**
- Local username and password: **Admin / Adminpa55**

Topology

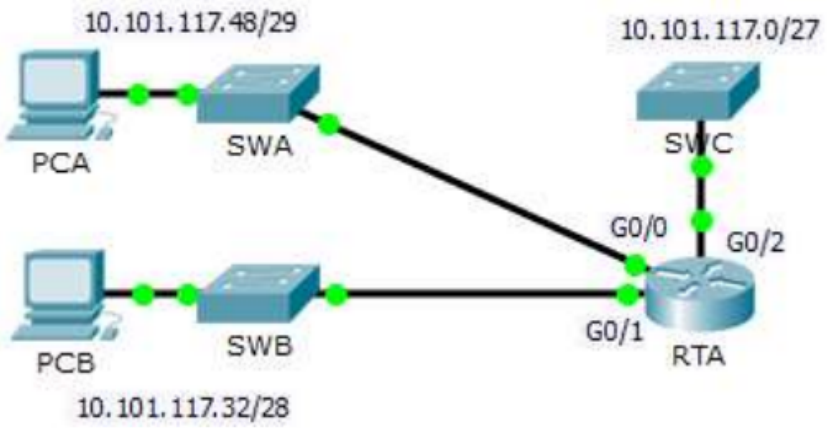


Figure 1: Topology

Addressing Table

Device	Interface	IP Address	Subnet Mask	Default Gateway
RTA	G0/0	10.101.117.49	255.255.255.248	N/A
	G0/1	10.101.117.33	255.255.255.240	N/A
	G0/2	10.101.117.1	255.255.255.224	N/A
PCA	NIC	10.101.117.51	255.255.255.248	10.101.117.49
PCB	NIC	10.101.117.35	255.255.255.240	10.101.117.33
SWA	VLAN 1	10.101.117.50	255.255.255.248	10.101.117.49
SWB	VLAN 1	10.101.117.34	255.255.255.240	10.101.117.33
SWC	VLAN 1	10.101.117.2	255.255.255.224	10.101.117.1

Table 1: Addressing Table

Task 1: Configure, Apply and Verify an Extended Numbered ACL

Configure, apply and verify an ACL to satisfy the following policy:

- SSH traffic from devices on the 10.101.117.32/28 network is allowed to devices on the 10.101.117.0/27 networks.
- ICMP traffic is allowed from any source to any destination.
- All other traffic to 10.101.117.0/27 is blocked.

1. Configure the extended ACL.
   a. From the appropriate configuration mode on RTA, use the last valid extended access list number to configure the ACL. Use the following steps to construct the first ACL statement:
      1. The last extended list number is 199.
      2. The protocol is TCP.
      3. The source network is 10.101.117.32.
      4. The wildcard can be determined by subtracting 255.255.255.240 from 255.255.255.255.
      5. The destination network is 10.101.117.0.
      6. The wildcard can be determined by subtracting 255.255.255.224 from 255.255.255.255.
      7. The protocol is SSH (port 22).

      What is the first ACL statement?

```
RTA(config)#access-list 199 permit tcp 10.101.117.32 0.0.0.15 10.101.117.0 0.0.0.31 eq 22
```

   b. ICMP is allowed, and a second ACL statement is needed. Use the same access list number to permit all ICMP traffic, regardless of the source or destination address. What is the second ACL statement? (Hint: Use the any keywords

```
RTA(config)#access-list 199 permit icmp any any
```

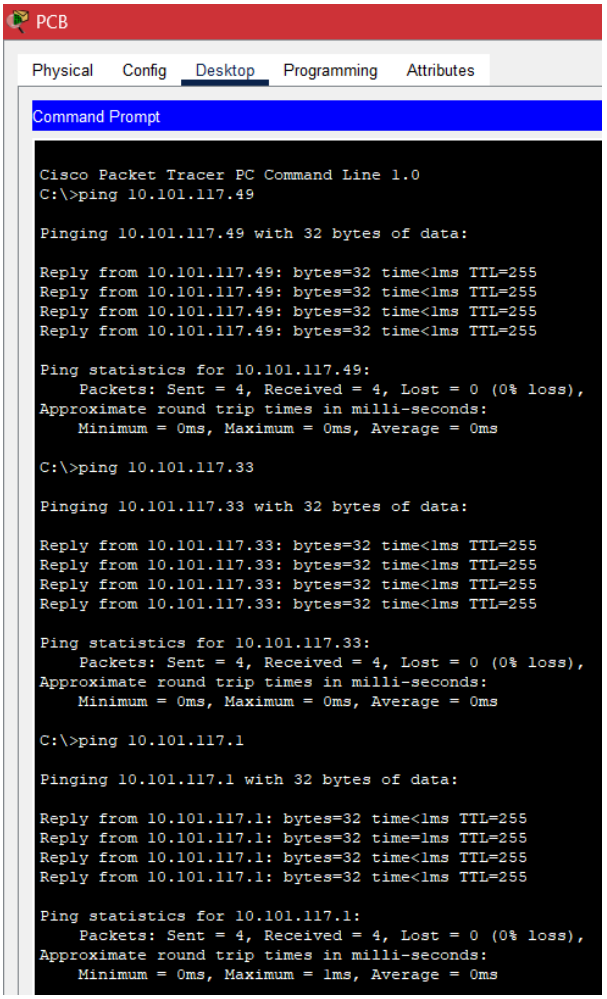
   c. All other IP traffic is denied, by default.

2. Apply the extended ACL

   The general rule is to place extended ACLs close to the source. However, because access list 199 affects traffic originating from both networks 10.101.117.48/29 and 10.101.117.32/28, the best placement for this ACL might be on interface Gigabit Ethernet 0/2 in the outbound direction. What is the command to apply ACL 199 to the Gigabit Ethernet 0/2 interface?

```
RTA(config)#interface GigabitEthernet 0/2
RTA(config-if)#ip access-group 199 out
```

3. Verify the extended ACL implementation
   a. Ping from **PCB** to all of the other IP addresses in the network. If the pings are unsuccessful, verify the IP addresses before continuing.



```
C:\>ping 10.101.117.51

Pinging 10.101.117.51 with 32 bytes of data:

Request timed out.
Reply from 10.101.117.51: bytes=32 time<1ms TTL=127
Reply from 10.101.117.51: bytes=32 time<1ms TTL=127
Reply from 10.101.117.51: bytes=32 time<1ms TTL=127

Ping statistics for 10.101.117.51:
    Packets: Sent = 4, Received = 3, Lost = 1 (25% loss),
Approximate round trip times in milli-seconds:
    Minimum = 0ms, Maximum = 0ms, Average = 0ms

C:\>ping 10.101.117.35

Pinging 10.101.117.35 with 32 bytes of data:

Reply from 10.101.117.35: bytes=32 time=11ms TTL=128
Reply from 10.101.117.35: bytes=32 time=6ms TTL=128
Reply from 10.101.117.35: bytes=32 time=6ms TTL=128
Reply from 10.101.117.35: bytes=32 time=5ms TTL=128

Ping statistics for 10.101.117.35:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 5ms, Maximum = 11ms, Average = 7ms

C:\>ping 10.101.117.50

Pinging 10.101.117.50 with 32 bytes of data:

Request timed out.
Request timed out.
Reply from 10.101.117.50: bytes=32 time<1ms TTL=254
Reply from 10.101.117.50: bytes=32 time<1ms TTL=254

Ping statistics for 10.101.117.50:
    Packets: Sent = 4, Received = 2, Lost = 2 (50% loss),
Approximate round trip times in milli-seconds:
    Minimum = 0ms, Maximum = 0ms, Average = 0ms
```

```
C:\>ping 10.101.117.34

Pinging 10.101.117.34 with 32 bytes of data:

Request timed out.
Reply from 10.101.117.34: bytes=32 time<1ms TTL=255
Reply from 10.101.117.34: bytes=32 time=9ms TTL=255
Reply from 10.101.117.34: bytes=32 time<1ms TTL=255

Ping statistics for 10.101.117.34:
    Packets: Sent = 4, Received = 3, Lost = 1 (25% loss),
Approximate round trip times in milli-seconds:
    Minimum = 0ms, Maximum = 9ms, Average = 3ms

C:\>ping 10.101.117.2

Pinging 10.101.117.2 with 32 bytes of data:

Request timed out.
Request timed out.
Reply from 10.101.117.2: bytes=32 time<1ms TTL=254
Reply from 10.101.117.2: bytes=32 time<1ms TTL=254

Ping statistics for 10.101.117.2:
    Packets: Sent = 4, Received = 2, Lost = 2 (50% loss),
Approximate round trip times in milli-seconds:
    Minimum = 0ms, Maximum = 0ms, Average = 0ms
```

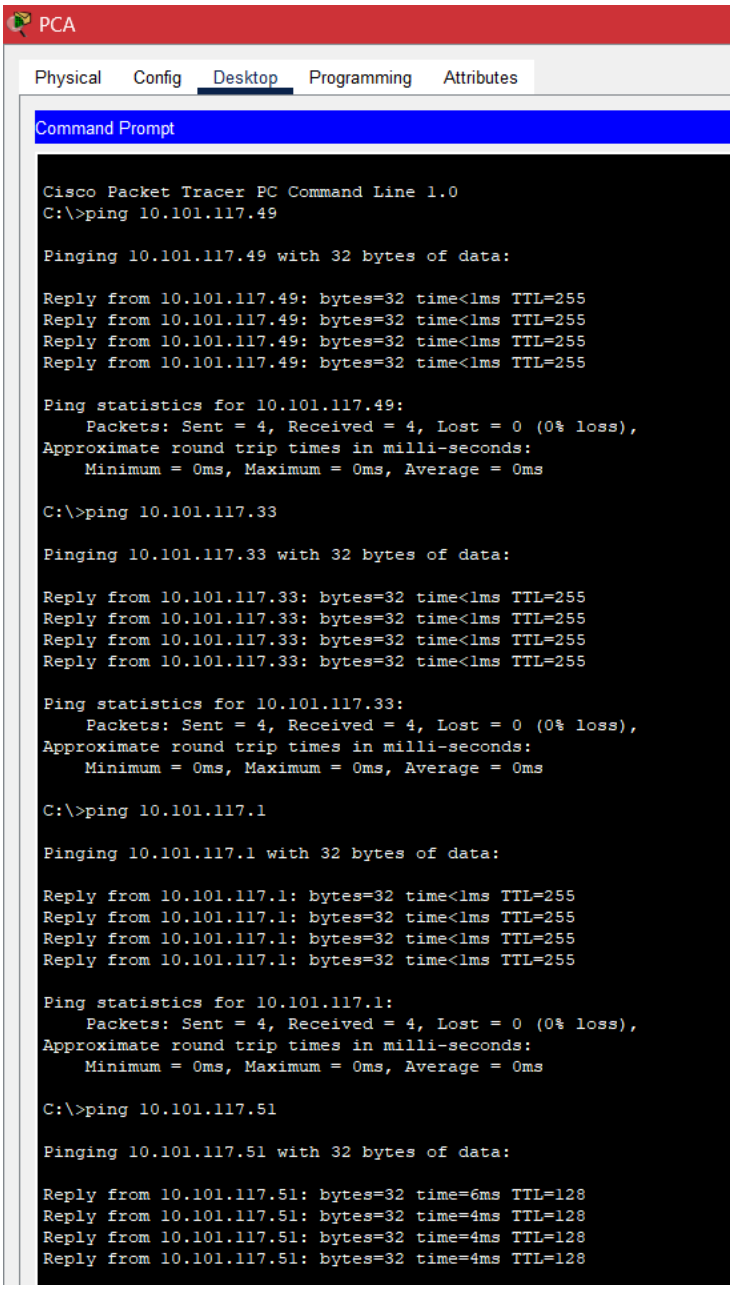
b. SSH from **PCB** to **SWC**. The username is **Admin**, and the password is **Adminpa55**.

```
PC> ssh -l Admin 10.101.117.2
```

c. Exit the SSH session to **SWC**.

```
C:\>ssh -l Admin 10.101.117.2

Password:
% Password:  timeout expired!
% Login invalid

[Connection to 10.101.117.2 closed by foreign host]
C:\>ssh -l Admin 10.101.117.2

Password:



SWC>exit

[Connection to 10.101.117.2 closed by foreign host]
```

d. Ping from **PCA** to all of the other IP addresses in the network. If the pings are unsuccessful, verify the IP addresses before continuing.


PCA
Physical
Config
Desktop
Programming
Attributes
Command Prompt
Cisco Packet Tracer PC Command Line 1.0
C:\>ping 10.101.117.49
Pinging 10.101.117.49 with 32 bytes of data:
Reply from 10.101.117.49: bytes=32 time<1ms TTL=255
Reply from 10.101.117.49: bytes=32 time<1ms TTL=255
Reply from 10.101.117.49: bytes=32 time<1ms TTL=255
Reply from 10.101.117.49: bytes=32 time<1ms TTL=255
Ping statistics for 10.101.117.49:
Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
Minimum = 0ms, Maximum = 0ms, Average = 0ms
C:\>ping 10.101.117.33
Pinging 10.101.117.33 with 32 bytes of data:
Reply from 10.101.117.33: bytes=32 time<1ms TTL=255
Reply from 10.101.117.33: bytes=32 time<1ms TTL=255
Reply from 10.101.117.33: bytes=32 time<1ms TTL=255
Reply from 10.101.117.33: bytes=32 time<1ms TTL=255
Ping statistics for 10.101.117.33:
Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
Minimum = 0ms, Maximum = 0ms, Average = 0ms
C:\>ping 10.101.117.1
Pinging 10.101.117.1 with 32 bytes of data:
Reply from 10.101.117.1: bytes=32 time<1ms TTL=255
Reply from 10.101.117.1: bytes=32 time<1ms TTL=255
Reply from 10.101.117.1: bytes=32 time<1ms TTL=255
Reply from 10.101.117.1: bytes=32 time<1ms TTL=255
Ping statistics for 10.101.117.1:
Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
Minimum = 0ms, Maximum = 0ms, Average = 0ms
C:\>ping 10.101.117.51
Pinging 10.101.117.51 with 32 bytes of data:
Reply from 10.101.117.51: bytes=32 time=6ms TTL=128
Reply from 10.101.117.51: bytes=32 time=4ms TTL=128
Reply from 10.101.117.51: bytes=32 time=4ms TTL=128
Reply from 10.101.117.51: bytes=32 time=4ms TTL=128

```
C:\>ping 10.101.117.35

Pinging 10.101.117.35 with 32 bytes of data:

Reply from 10.101.117.35: bytes=32 time<1ms TTL=127
Reply from 10.101.117.35: bytes=32 time<1ms TTL=127
Reply from 10.101.117.35: bytes=32 time<1ms TTL=127
Reply from 10.101.117.35: bytes=32 time=12ms TTL=127

Ping statistics for 10.101.117.35:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 0ms, Maximum = 12ms, Average = 3ms

C:\>ping 10.101.117.50

Pinging 10.101.117.50 with 32 bytes of data:

Request timed out.
Reply from 10.101.117.50: bytes=32 time<1ms TTL=255
Reply from 10.101.117.50: bytes=32 time<1ms TTL=255
Reply from 10.101.117.50: bytes=32 time<1ms TTL=255

Ping statistics for 10.101.117.50:
    Packets: Sent = 4, Received = 3, Lost = 1 (25% loss),
Approximate round trip times in milli-seconds:
    Minimum = 0ms, Maximum = 0ms, Average = 0ms

C:\>ping 10.101.117.34

Pinging 10.101.117.34 with 32 bytes of data:

Request timed out.
Request timed out.
Reply from 10.101.117.34: bytes=32 time<1ms TTL=254
Reply from 10.101.117.34: bytes=32 time<1ms TTL=254

Ping statistics for 10.101.117.34:
    Packets: Sent = 4, Received = 2, Lost = 2 (50% loss),
Approximate round trip times in milli-seconds:
    Minimum = 0ms, Maximum = 0ms, Average = 0ms

C:\>ping 10.101.117.2

Pinging 10.101.117.2 with 32 bytes of data:

Reply from 10.101.117.2: bytes=32 time<1ms TTL=254
Reply from 10.101.117.2: bytes=32 time<1ms TTL=254
Reply from 10.101.117.2: bytes=32 time<1ms TTL=254
Reply from 10.101.117.2: bytes=32 time<1ms TTL=254

Ping statistics for 10.101.117.2:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
```

e. SSH from **PCA** to **SWC**. The access list causes the router to reject the connection.

```
C:\>ssh -l admin 10.101.117.2

% Connection timed out; remote host not responding
```

f. SSH from **PCA** to **SWB**. The access list is placed on **G0/2** and does not affect this connection. The username is **Admin**, and the password is **Adminpa55**.

g. After logging into **SWB**, do not log out. SSH to **SWC** in privileged EXEC mode.

```
SWB# ssh -l Admin 10.101.117.2
```

```
SWB>ssh -l Admin 10.101.117.2
        ^
% Invalid input detected at '^' marker.

SWB>ssh -l Admin 10.101.117.2
        ^
% Invalid input detected at '^' marker.

SWB>en
Password:
SWB#ssh -l Admin 10.101.117.2

Password:
% Login invalid


Password:
% Login invalid


Password:



SWC>
```

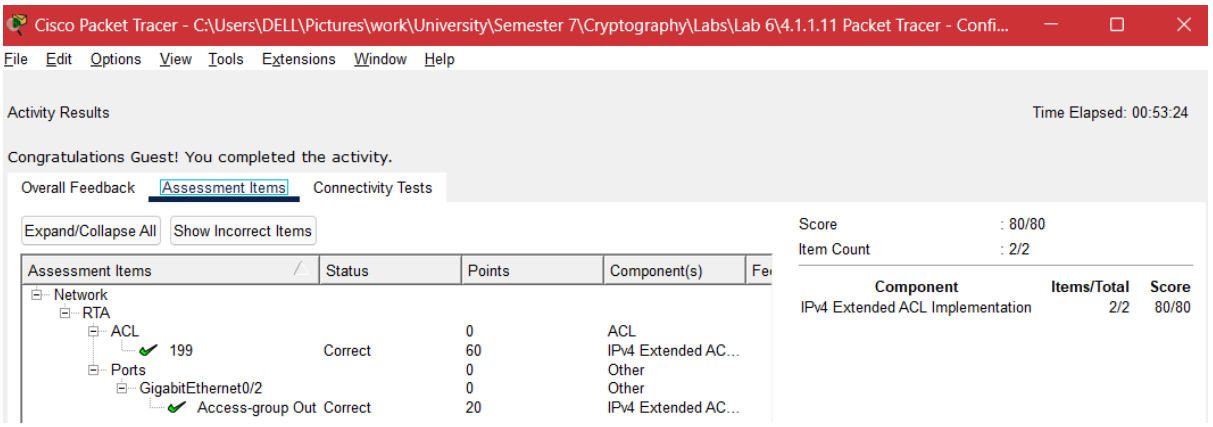
Task 2: Reflection

1. How was PCA able to bypass access list 199 and SSH to SWC?

The process involved two steps: First, PCA used SSH to connect to SWB. From there, SSH access was granted to SWC.

2. What could have been done to prevent PCA from accessing SWC indirectly, while allowing PCB SSH access to SWC?

To block all traffic to 10.101.117.0/27 except for SSH from 10.101.117.32/28, the access list can stay as it is. However, rather than applying it outbound on G0/2, it should be applied inbound on both G0/0 and G0/1 interfaces.



5.3 Configure IP ACLSs to Mitigate Attacks

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:

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

Topology:

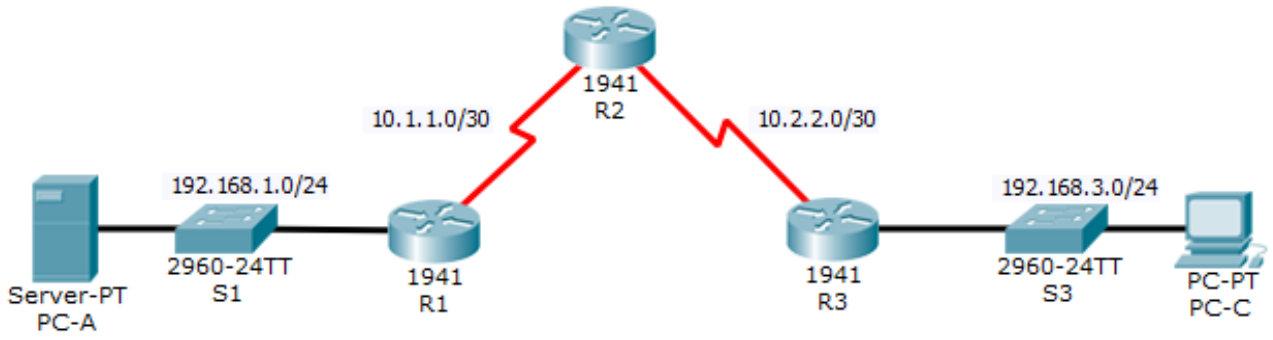


Figure 2

Addressing Table:

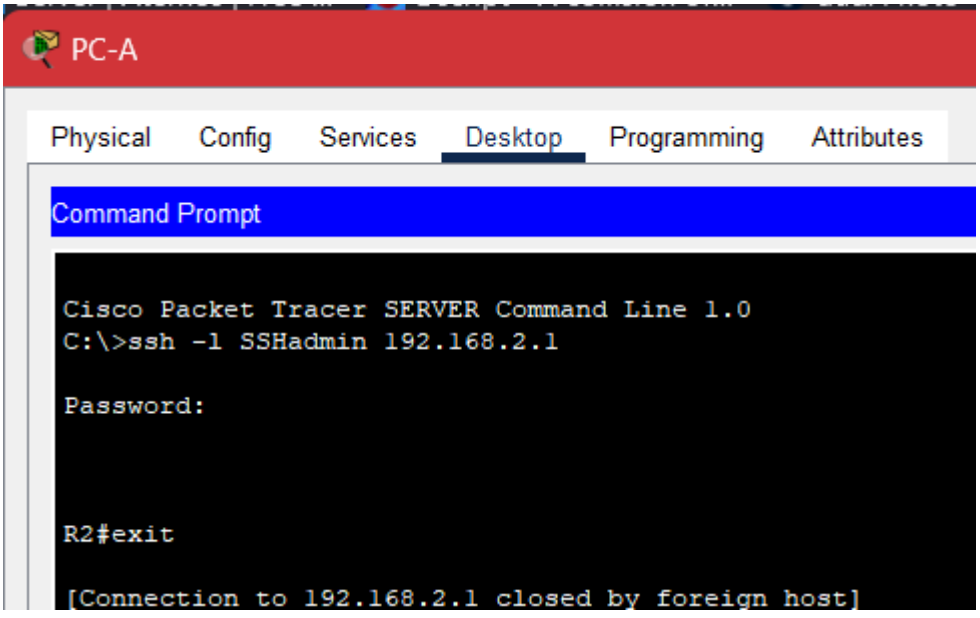
Device	Interface	IP Address	Subnet Mask	Default Gateway	Switch Port
R1	G0/1	192.168.1.1	255.255.255.0	N/A	S1 F0/5
	S0/0/0 (DCE)	10.1.1.1	255.255.255.252	N/A	N/A
R2	S0/0/0	10.1.1.2	255.255.255.252	N/A	N/A
	S0/0/1 (DCE)	10.2.2.2	255.255.255.252	N/A	N/A
	Lo0	192.168.2.1	255.255.255.0	N/A	N/A
R3	G0/1	192.168.3.1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

Table 2

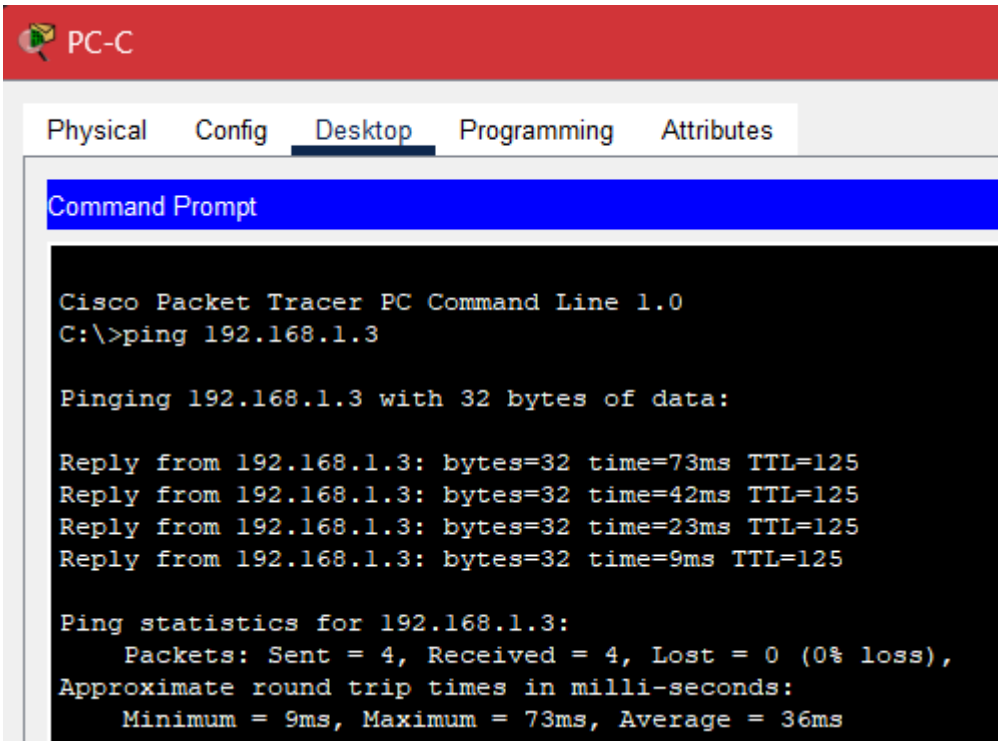
Task 3.1: Verify Basic Network Connectivity

1. From PC-A, verify connectivity to PC-C and R2.
   a. From the command prompt, ping **PC-C** (192.168.3.3).
   b. From the command prompt, establish an SSH session to **R2** Lo0 interface (192.168.2.1) using username **SSHadmin** and password **ciscosshpa55**. When finished, exit the SSH session.

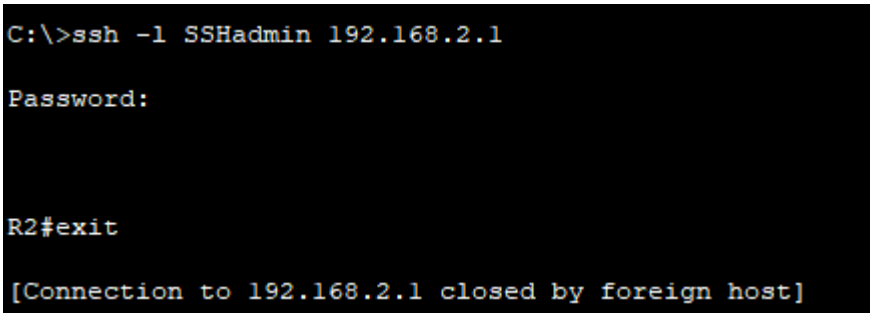
   `SERVER>` **ssh -l SSHadmin 192.168.2.1**



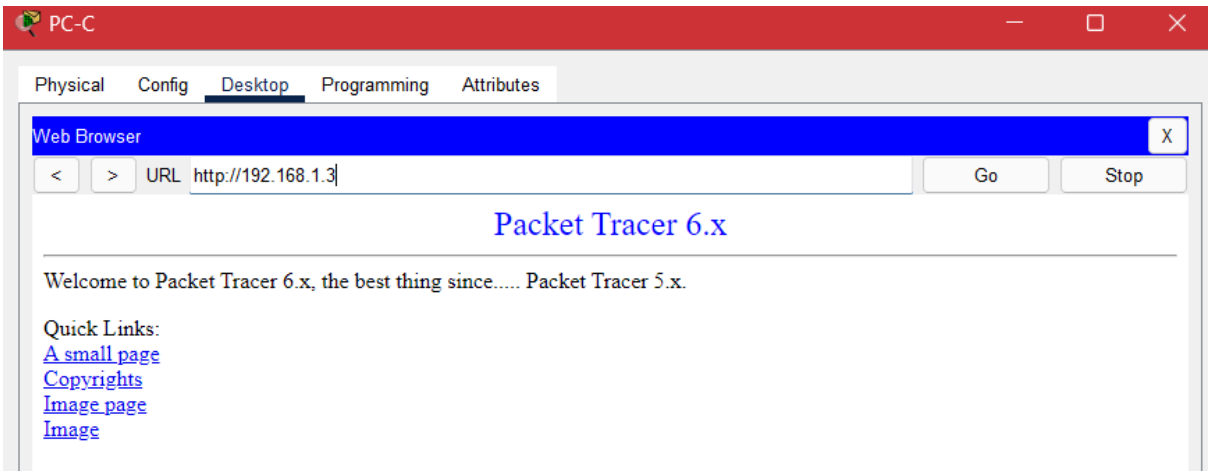
2. From PC-C, verify connectivity to PC-A and R2.
   a. From the command prompt, ping **PC-A** (192.168.1.3).



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



Task 3.2: Secure Access to Routers

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

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

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

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

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

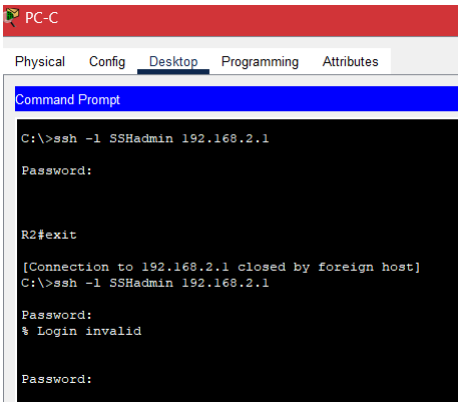
2. Apply ACL 10 to ingress traffic on the VTY lines.

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

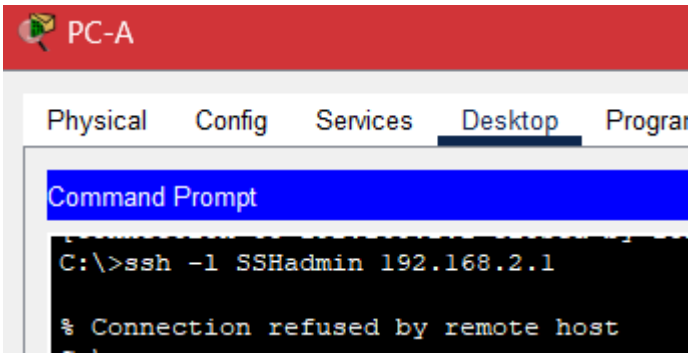
```
R1(config)#line vty 0 4
R1(config-line)#access-class 10 in

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
```

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).



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



Task 3.3: Create a Numbered IP ACL 120 on R1

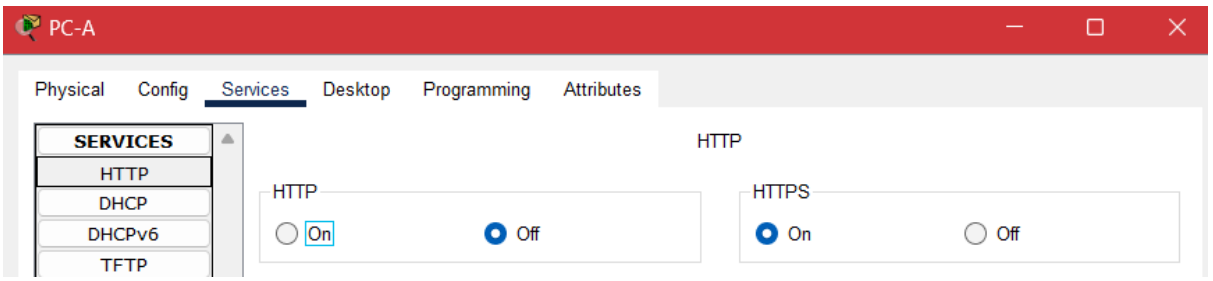
Create an IP ACL numbered 120 with the following rules:

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

Note: Check Results will not show a correct configuration for ACL 120 until you modify it in Part 4

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.



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

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

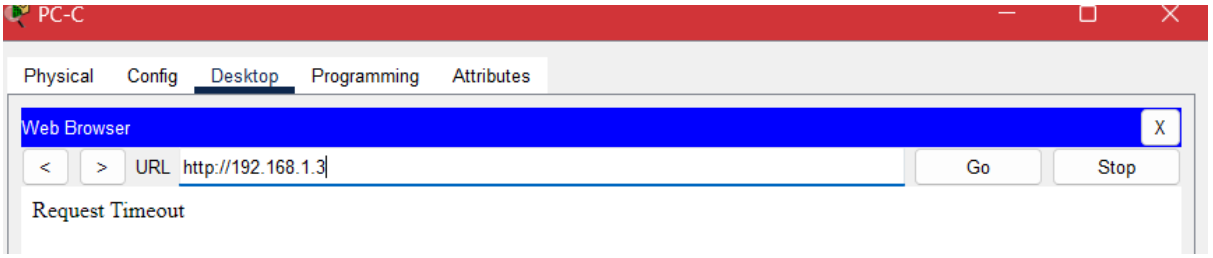
```
R1(config)#access-list 120 permit udp any host 192.168.1.3 eq domain
R1(config)#access-list 120 permit tcp any host 192.168.1.3 eq smtp
R1(config)#access-list 120 permit tcp any host 192.168.1.3 eq ftp
R1(config)#access-list 120 deny tcp any host 192.168.1.3 eq 443
R1(config)#access-list 120 permit tcp host 192.168.3.3 host 10.1.1.1 eq 22
```

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

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

```
R1(config)#interface s0/0/0
R1(config-if)#ip access-group 120 in
```

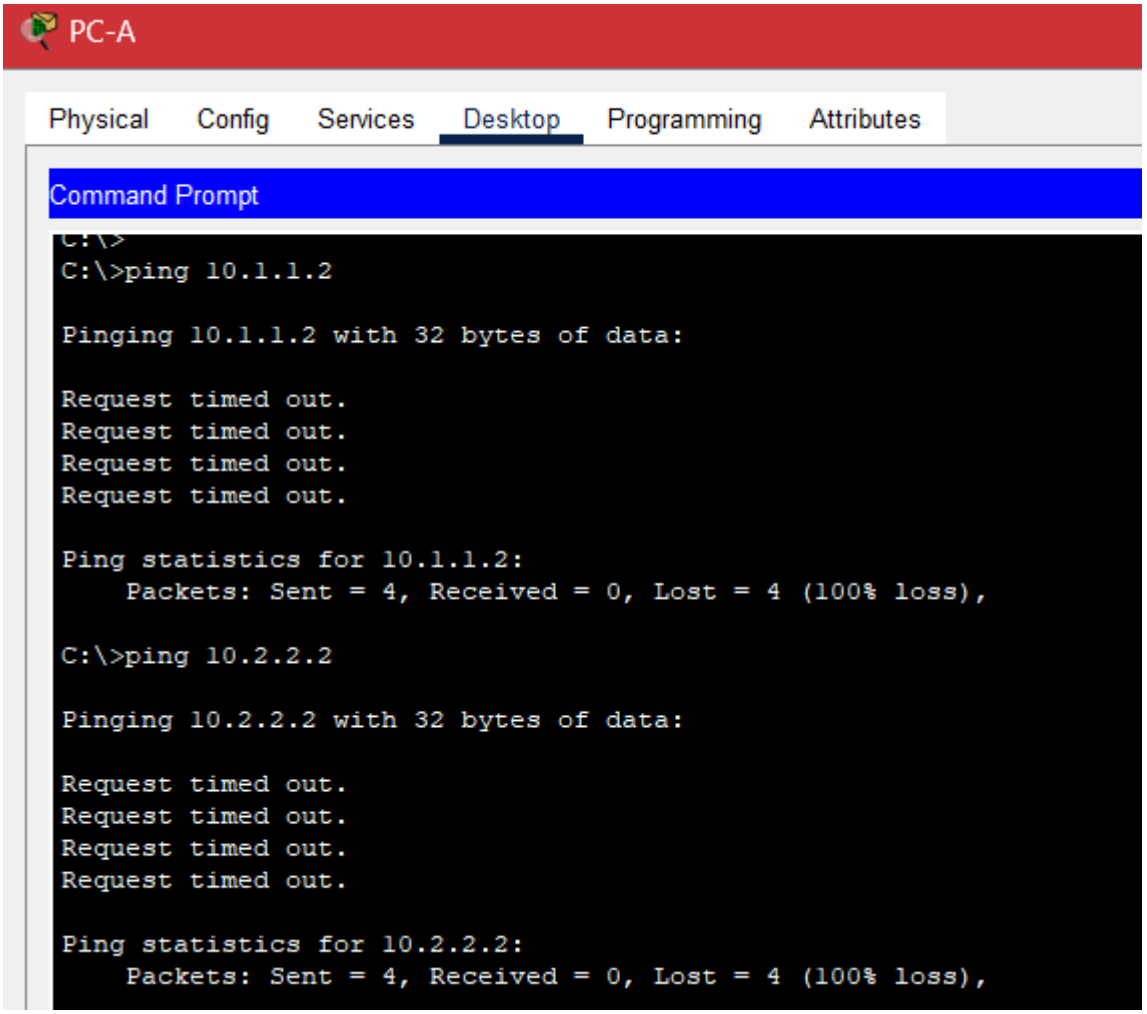
4. Verify that PC-C cannot access PC-A via HTTPS using the web browser.



Task 3.4: 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.

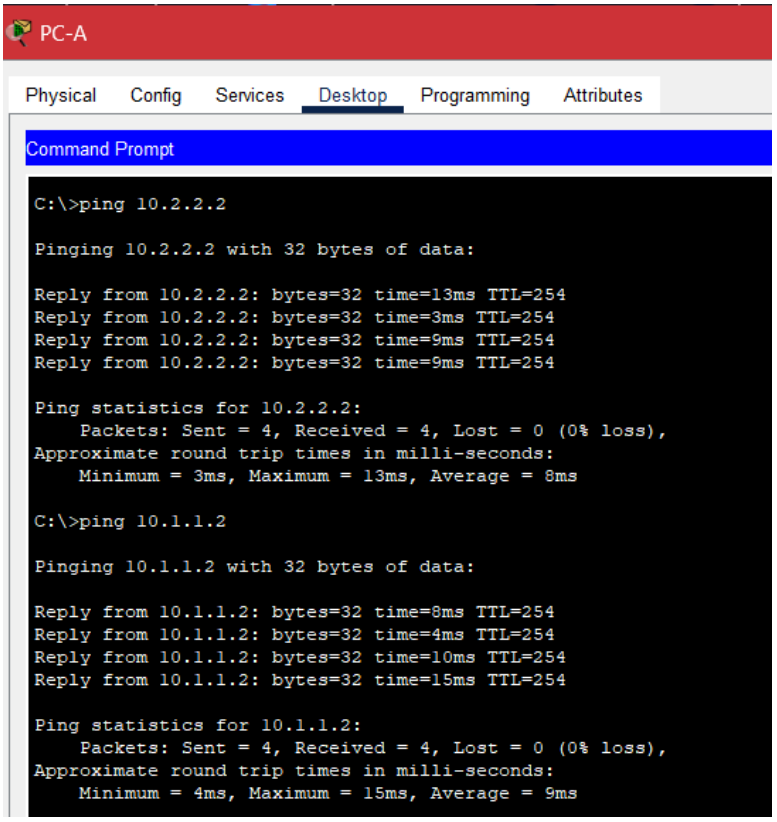
1. Verify that PC-A cannot successfully ping the loopback interface on R2



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

```
R1#config t
Enter configuration commands, one per line.  End with CNTL/Z.
R1(config)#access-list 120 permit icmp any any echo-reply
R1(config)#access-list 120 permit icmp any any unreachable
R1(config)#access-list 120 deni icmp any any
                           ^
% Invalid input detected at '^' marker.

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

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



Task 3.5: Create a Numbered IP ACL 110 on R3

Deny all outbound packets with source address outside the range of internal IP addresses on **R3**.

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

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

```
R3>en
Password:
R3#config 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
```

2. **Apply the ACL to interface G0/1.**

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

Use the **ip access-group** command to apply the access list to incoming traffic on interface G0/1.

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

Task 3.6: 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**.

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 lab, 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
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 100deny ip 192.168.0.0 0.0.255.255 any
                           ^
% Invalid input detected at '^' marker.

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
```

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

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

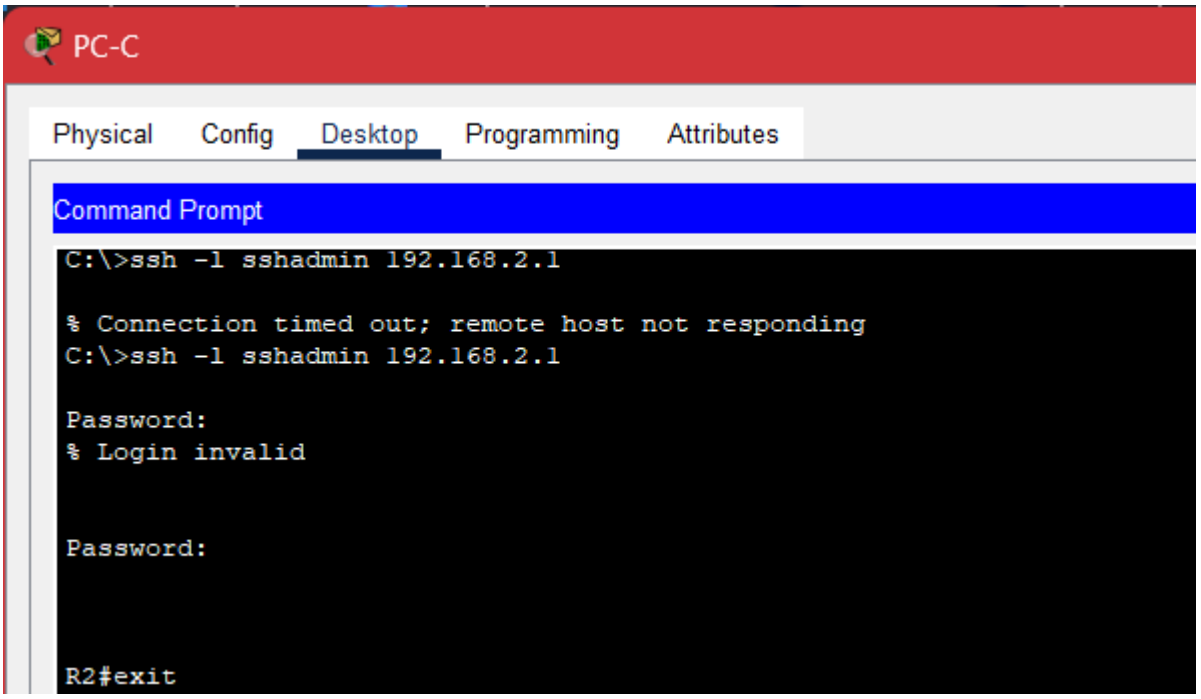
```
R3(config)#interface s0/0/1
R3(config-if)#ip access-group 100 in
```

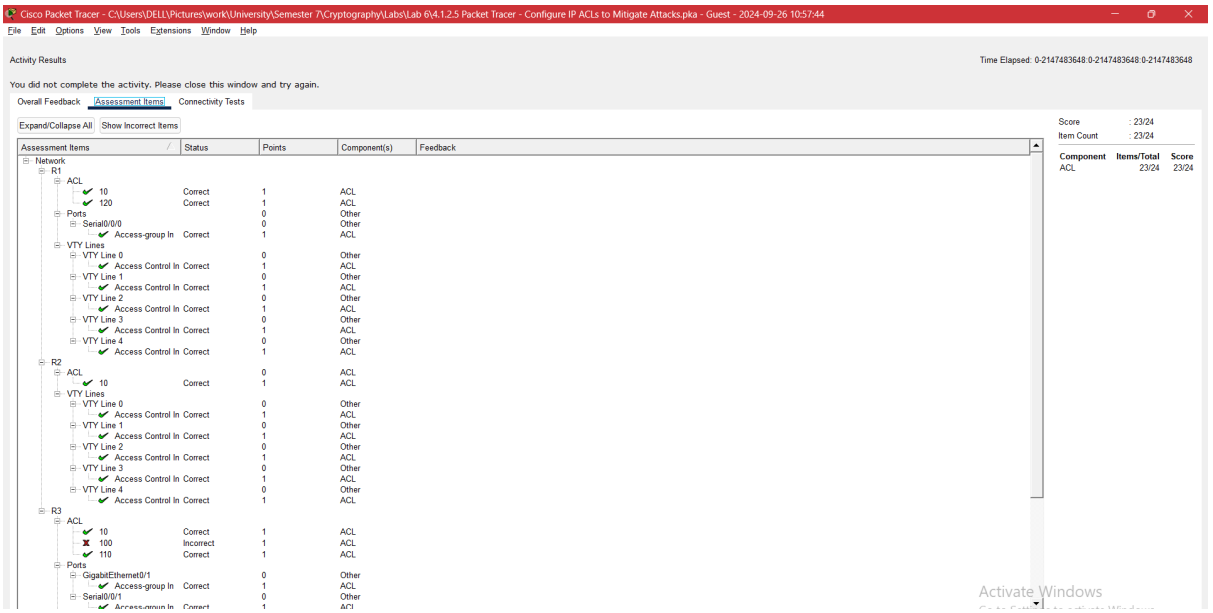
3. Confirm that the specified traffic entering interface Serial 0/0/1 is handled correctly.

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.

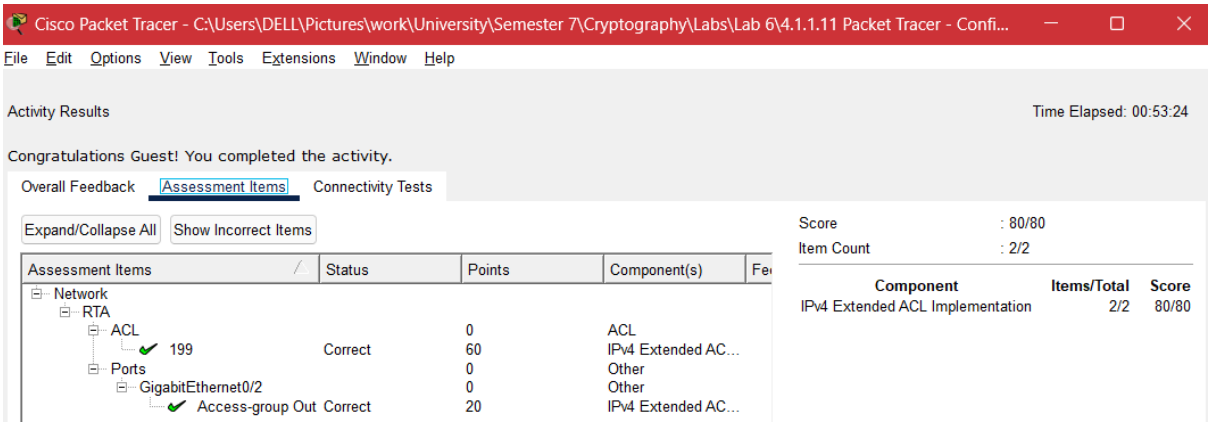
```
C:\>ping 192.168.1.3

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),
```

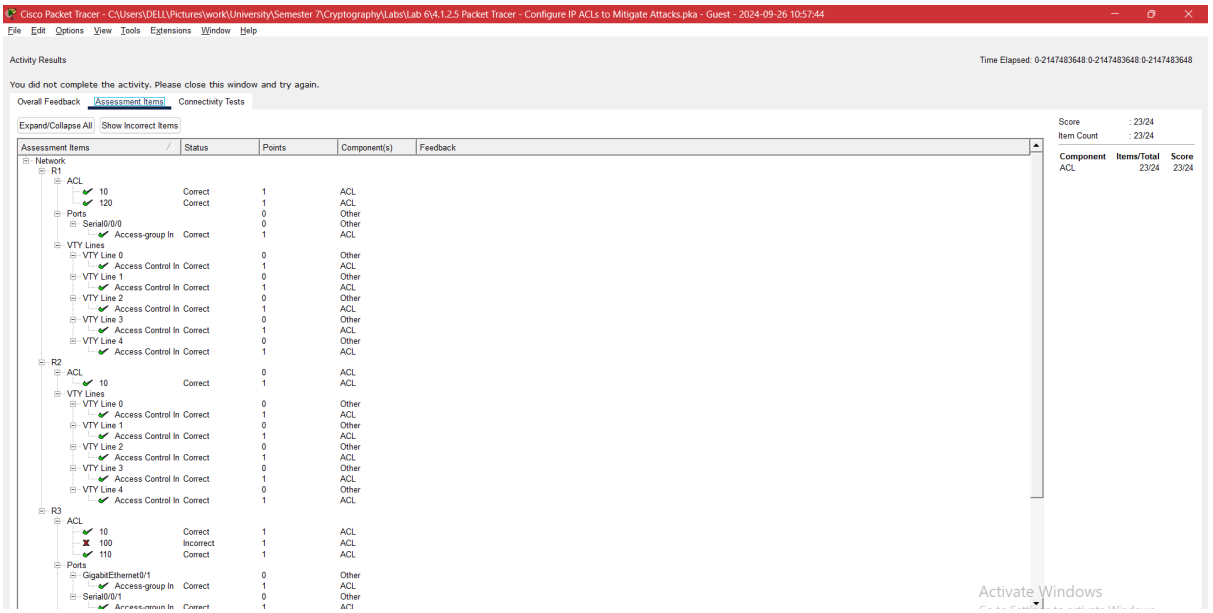
b. Establish an SSH session to 192.168.2.1 from **PC-C**.





PROOF OF COMPLETION OF EVERYTHING




Cisco Packet Tracer - C:\Users\DELL\Pictures\work\University\Semester 7\Cryptography\Labs\Lab 6\4.1.2.5 Packet Tracer - Configure IP ACLs to Mitigate Attacks.pka - Guest - 2024-09-26 10:57:44
File Edit Options View Tools Extensions Window Help
Activity Results
Time Elapsed: 0-2147483648:0-2147483648:0-2147483648
You did not complete the activity. Please close this window and try again.
Overall Feedback
Assessment Items
Connectivity Tests
Expand/Collapse All
Show Incorrect Items
Score : 23/24
Item Count : 23/24
Component Items/Total Score
ACL 23/24 23/24
Assessment Items
Status
Points
Component(s)
Feedback
Network
R1
ACL
10 Correct 1 ACL
120 Correct 1 ACL
Ports 0 Other
Serial0/0/0 0 Other
Access-group In Correct 1 ACL
VTY Lines
VTY Line 0 0 Other
Access Control In Correct 1 ACL
VTY Line 1 0 Other
Access Control In Correct 1 ACL
VTY Line 2 0 Other
Access Control In Correct 1 ACL
VTY Line 3 0 Other
Access Control In Correct 1 ACL
VTY Line 4 0 Other
Access Control In Correct 1 ACL
R2
ACL 0 ACL
10 Correct 1 ACL
VTY Lines
VTY Line 0 0 Other
Access Control In Correct 1 ACL
VTY Line 1 0 Other
Access Control In Correct 1 ACL
VTY Line 2 0 Other
Access Control In Correct 1 ACL
VTY Line 3 0 Other
Access Control In Correct 1 ACL
VTY Line 4 0 Other
Access Control In Correct 1 ACL
R3
ACL
10 Correct 1 ACL
100 Incorrect 1 ACL
110 Correct 1 ACL
Ports
GigabitEthernet0/1 0 Other
Access-group In Correct 1 ACL
Serial0/0/1 0 Other
Activate Windows

6.4 - Configuring IPv6 ACLs – Optional

Topology:

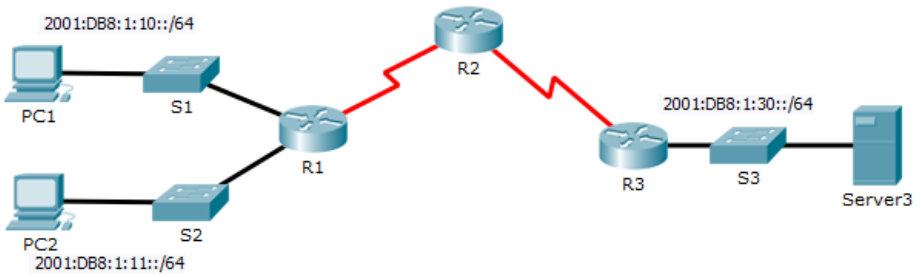


Figure 3

Addressing Table:

Device	Interface	IPv6 Address/Prefix	Default Gateway
Server3	NIC	2001:DB8:1:30::30/64	FE80::30

Table 3

Task 4.1: 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.

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(config)# deny tcp any host 2001:DB8:1:30::30 eq www
R1(config)# deny tcp any host 2001:DB8:1:30::30 eq 443
```

b. Allow all other IPv6 traffic to pass.

2. **Apply the ACL to the correct interface.**

   Apply the ACL on the interface closest to the source of the traffic to be blocked.

```
R1(config-if)# ipv6 traffic-filter BLOCK_HTTP in
```

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

Task 4.2: 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.

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.

   b. Allow all other IPv6 traffic to pass.

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.

3. Verify that the proper access list functions.
   a. Ping from **PC2** to 2001:DB8:1:30::30. The ping should fail.
   b. Ping from **PC1** to 2001:DB8:1:30::30. The ping should fail.

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

Assessment Rubric
Lab 06
Configuring Extended and IPv6 ACLs

Name: Syed Asghar Abbas Zaidi	**Student ID: 07201**

Points Distribution

Task No.	**LR 2 Simulation**	**LR9 Report**
Task 1	20	
Task 2	30	
Task 3	30	
Task 4	-	
Total	/80	/10
CLO Mapped	CLO 1	CLO1

Affective Domain Rubric		**Points**	**CLO Mapped**
AR 7	Report Submission	/10	CLO 1

CLO	**Total Points**	**Points Obtained**
1	90	
1	10	
Total	**100**	

For description of different levels of the mapped rubrics, please refer the provided Lab Evaluation Assessment Rubrics and Affective Domain Assessment Rubrics.

Lab Evaluation Assessment Rubric

#	Assessment Elements	Level 1: Unsatisfactory Points 0-1	Level 2: Developing Points 2	Level 3:Good Points 3	Level 4:Exemplary Points 4
LR2	**Program/Code / Simulation Model/ Network Model**	Program/code/simulation model/network model does not implement the required functionality and has several errors. The student is not able to utilize even the basic tools of the software.	Program/code/simulation model/network model has some errors and does not produce completely accurate results. Student has limited command on the basic tools of the software.	Program/code/simulation model/network model gives correct output but not efficiently implemented or implemented by computationally complex routine.	Program/code/simulation /network model is efficiently implemented and gives correct output. Student has full command on the basic tools of the software.
LR5	**Results & Plots**	Figures/ graphs / tables are not developed or are poorly constructed with erroneous results. Titles, captions, units are not mentioned. Data is presented in an obscure manner.	Figures, graphs and tables are drawn but contain errors. Titles, captions, units are not accurate. Data presentation is not too clear.	All figures, graphs, tables are correctly drawn but contain minor errors or some of the details are missing.	Figures / graphs / tables are correctly drawn and appropriate titles/captions and proper units are mentioned. Data presentation is systematic.
LR9	**Report**	All the in-lab tasks are not included in report and / or the report is submitted too late.	Most of the tasks are included in report but are not well explained. All the necessary figures / plots are not included. Report is submitted after due date.	Good summary of most the in-lab tasks is included in report. The work is supported by figures and plots with explanations. The report is submitted timely.	Detailed summary of the in-lab tasks is provided. All tasks are included and explained well. Data is presented clearly including all the necessary figures, plots and tables.