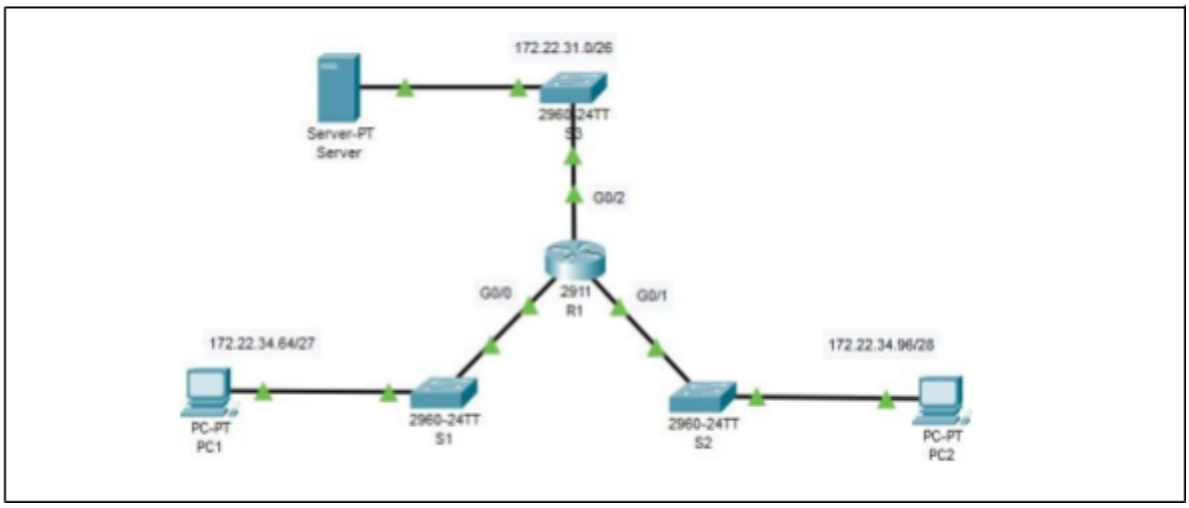
BSCITP602

Name :Harsh Kadu Roll No : 33

PRACTICAL NO 3A

Aim : Configuring Extended ACL's - Scenario 1



Addressing Table :

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

Name :Harsh Kadu Roll No : 33

Step 1 : Place the components as shown in the figure above and setup a network . It is necessary to configure each components placed in the network . Use proper connection wire to connect between the devices.

Part 1 : Configure, Apply and Verify an Extended Numbered ACL.

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

Click on Router R1→ Go to CLI Tab → and Type the following commands one by one :

a. From global configuration mode on **R1**, enter the following command to determine the first valid number for an extended access list.

```
R1>enable
R1#conf t
Enter configuration commands, one per line.  End with CNTL/Z.
R1(config)#access-list ?
  <1-99>     IP standard access list
  <100-199>  IP extended access list
```

b. Add **100** to the command, followed by a question mark.

```
R1(config)#access-list 100 ?
  deny    Specify packets to reject
  permit  Specify packets to forward
  remark  Access list entry comment
```

c. To permit FTP traffic, enter **permit**, followed by a question mark.

```
R1(config)#access-list 100 permit ?
  ahp    Authentication Header Protocol
  eigrp  Cisco's EIGRP routing protocol
  esp    Encapsulation Security Payload
  gre    Cisco's GRE tunneling
  icmp   Internet Control Message Protocol
  ip     Any Internet Protocol
  ospf   OSPF routing protocol
  tcp    Transmission Control Protocol
  udp    User Datagram Protocol
```

d. This ACL permits FTP and ICMP. ICMP is listed above, but FTP is not, because FTP uses TCP. Therefore,enter **tcp** to further refine the ACL help.

```
R1(config)#access-list 100 permit tcp ?
  A.B.C.D  Source address
  any      Any source host
  host     A single source host
```

e. Notice that we could filter just for **PC1** by using the **host** keyword or we could allow any **host**. In this case, any device is allowed that has an address belonging to the 172.22.34.64/27 network. Enter the network address, followed by a question mark.

```
R1(config)#access-list 100 permit tcp 172.22.34.64 ?
  A.B.C.D  Source wildcard bits
```

f. Calculate the wildcard mask determining the binary opposite of a subnet mask. Enter the wildcard mask, followed by a question mark.

```
R1(config)#access-list 100 permit tcp 172.22.34.64 0.0.0.31 ?
  A.B.C.D  Destination address
  any      Any destination host
  eq       Match only packets on a given port number
  gt       Match only packets with a greater port number
  host     A single destination host
  lt       Match only packets with a lower port number
  neq      Match only packets not on a given port number
  range    Match only packets in the range of port numbers
```

g. Configure the destination address. In this scenario, we are filtering traffic for a single destination, which is the server. Enter the **host** keyword followed by the server's IP address.

```
R1(config)#access-list 100 permit tcp 172.22.34.64 0.0.0.31 host
172.22.34.62 ?
  dscp         Match packets with given dscp value
  eq           Match only packets on a given port number
  established  established
  gt           Match only packets with a greater port number
  lt           Match only packets with a lower port number
  neq          Match only packets not on a given port number
  precedence   Match packets with given precedence value
  range        Match only packets in the range of port numbers
  <cr>
```

h. Notice that one of the options is **<cr>** (carriage return). In other words, you can press Enter and the statement would permit all TCP traffic. However, we are only permitting FTP traffic; therefore, enter the **eq** keyword, followed by a question mark to display the available options. Then, enter **ftp** and press , **Enter**.

Name :Harsh Kadu Roll No : 33

```
R1(config)#access-list 100 permit tcp 172.22.34.64 0.0.0.31 host
172.22.34.62 eq ?
  <0-65535>  Port number
  ftp        File Transfer Protocol (21)
  pop3       Post Office Protocol v3 (110)
  smtp       Simple Mail Transport Protocol (25)
  telnet     Telnet (23)
  www        World Wide Web (HTTP, 80)
```

```
R1(config)#access-list 100 permit tcp 172.22.34.64 0.0.0.31 host
172.22.34.62 eq ftp
```

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

j. All other traffic is denied, by default.

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

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

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

Step 3 : Verify the ACL Implementation

1. Ping from **PC1** to **Server**.

Click on PC1 → Go to Desktop option → Enter the following commands :

```
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 = 1ms, Average = 0ms
```

2. FTP from **PC1** to **Server**. The username and password are both **cisco**. After successful login exit from the **ftp** service.

Click on PC1 → Go to Desktop option → Enter the following commands :

```
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>quit

221- Service closing control connection.
```

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

Click on PC1 → Go to Desktop option → and type the following commands :

```
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),
```

Part 2 : Configure , Apply and verify an Extended Named ACL.

Step 1 : Configure an ACL to permit HTTP access and ICMP.

Click on Router R1→ Go to CLI Tab → and Type the following commands one by one :

a. Named ACLs start with the **ip** keyword. From global configuration mode of **R1**, enter the following command, followed by a question mark.

```
R1>enable
R1#conf t
Enter configuration commands, one per line.  End with CNTL/Z.
R1(config)#ip access-list ?
  extended  Extended Access List
  standard  Standard Access List
```

b. You can configure named standard and extended ACLs. This access list filters both source and destination IP addresses; therefore, it must be extended. Enter **HTTP_ONLY** as the name. (For Packet Tracer scoring, the name is case-sensitive.)

```
R1(config)#ip access-list extended HTTP_ONLY
```

c. The prompt changes. You are now in extended named ACL configuration mode. All devices on the **PC2** LAN need TCP access. Enter the network address, followed by a question mark.

```
R1(config-ext-nacl)#permit tcp 172.22.34.96 ?
  A.B.C.D  Source wildcard bits
```

d. Enter the wildcard mask, followed by a question mark.

```
R1(config-ext-nacl)#permit tcp 172.22.34.96 0.0.0.15 ?
  A.B.C.D  Destination address
  any      Any destination host
  eq       Match only packets on a given port number
  gt       Match only packets with a greater port number
  host     A single destination host
  lt       Match only packets with a lower port number
  neq      Match only packets not on a given port number
  range    Match only packets in the range of port numbers
```

e. Finish the statement by specifying the server address as you did in Part 1 and filtering **www** traffic.

```
R1(config-ext-nacl)#permit tcp 172.22.34.98 0.0.0.15 host
172.22.34.62 eq www
```

f. Create a second access list statement to permit ICMP (ping, etc.) traffic from **PC2** to **Server**. Note: The prompt remains the same and a specific type of ICMP traffic does not need to be specified.

```
R1(config-ext-nacl)#permit icmp 172.22.34.98 0.0.0.15 host
172.22.34.62
```

g. All other traffic is denied, by default. Exit out of extended named ACL configuration mode.

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

From **R1's** perspective, the traffic that access list **HTTP_ONLY** applies to is inbound from the network connected to Gigabit Ethernet 0/1 interface. Enter the interface configuration mode and apply the ACL.

```
R1(config)#interface gigabitEthernet 0/1
R1(config-if)#ip access-group HTTP_ONLY in
```

Step 3 : Verify the ACL implementation.

a. Ping from **PC2** to **Server**. The ping should be successful, if the ping is 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=3ms 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 = 3ms, Average = 1ms
```

b. FTP from **PC2** to **Server**. The connection should fail.

c. Open the web browser on **PC2** and enter the IP address of **Server** as the URL. The connection should be successful.

BSCITP602

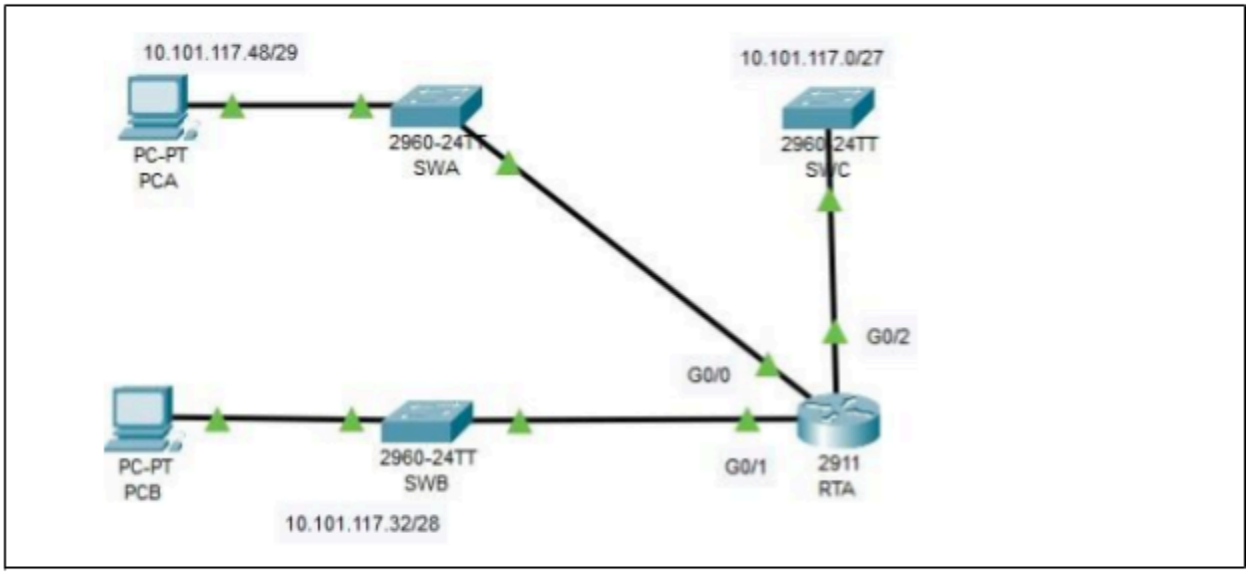
Name :Harsh Kadu Roll No : 33

BSCITP602

Name :Harsh Kadu Roll No : 33

PRACTICAL NO 3B

Aim : Configuring Extended ACLs - Scenario 2



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	VLAN1	10.101.117.50	255.255.255.248	10.101.117.49
SWB	VLAN1	10.101.117.34	255.255.255.240	10.101.117.33

SWC	VLAN1	10.101.117.2	255.255.255.224	10.101.117.1

Step 1 : Place the components as shown in the figure above and setup a network . It is necessary to configure each components placed in the network . Use proper connection wire to connect between the devices.

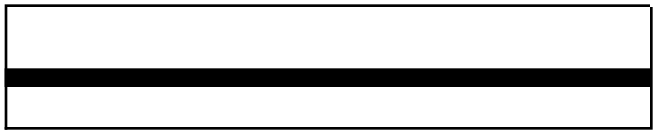
To Configure Switches SWA, SWB and SWC:

For Switch SWA :

Click on Switch SWA → Go to CLI Tab → and type the following commands to set the ip address of Switch.

```
SWA>enable
SWA#conf t
Enter configuration commands, one per line.  End with CNTL/Z.
SWA(config)#interface vlan1
SWA(config-if)#ip address 10.101.117.50 255.255.255.240
SWA(config-if)#no shut

SWA(config-if)#
%LINK-5-CHANGED: Interface Vlan1, changed state to up

%LINEPROTO-5-UPDOWN: Line protocol on Interface Vlan1, changed state
to up

SWA(config-if)#no shut
SWA(config-if)#exit
SWA(config)#ip default-gateway 10.101.117.49
SWA(config)#^Z
SWA#
%SYS-5-CONFIG_I: Configured from console by console

SWA#exit
```

Using show ip interface brief command it will list all the ip addresses associated with the switch and even vlan1



```
GigabitEthernet0/1     unassigned      YES manual down                  down
GigabitEthernet0/2     unassigned      YES manual down                  down
Vlan1                  10.101.117.50   YES manual up                    up
```

For Switch SWB :

Click on Switch SWB → Go to CLI Tab → and type the following commands to set the ip address of Switch.

```
SWB>enable
SWB#conf t
Enter configuration commands, one per line.  End with CNTL/Z.
SWB(config)#interface vlan 1
SWB(config-if)#ip address 10.101.117.34 255.255.255.240
SWB(config-if)#no shut

SWB(config-if)#
%LINK-5-CHANGED: Interface Vlan1, changed state to up

%LINEPROTO-5-UPDOWN: Line protocol on Interface Vlan1, changed state
to up

SWB(config-if)#ip default-gateway 10.101.117.33
SWB(config)#exit
SWB#
%SYS-5-CONFIG_I: Configured from console by console

SWB#exit
```

Using show ip interface brief command it will list all the ip addresses associated with the switch and even vlan1

```
SWB>show ip interface brief
```

```
GigabitEthernet0/1     unassigned      YES manual down                  down
GigabitEthernet0/2     unassigned      YES manual down                  down
Vlan1                  10.101.117.34   YES manual up                    up
```

For Switch SWC :

Click on Switch SWC → Go to CLI Tab → and type the following commands to set the ip address of Switch.

```
SWC>enable
SWC#conf t
Enter configuration commands, one per line.  End with CNTL/Z.
SWC(config)#interface vlan 1
SWC(config-if)#ip address 10.101.117.2 255.255.255.224
SWC(config-if)#no shut

SWC(config-if)#
%LINK-5-CHANGED: Interface Vlan1, changed state to up

%LINEPROTO-5-UPDOWN: Line protocol on Interface Vlan1, changed state
to up

SWC(config-if)#ip default-gateway 10.101.117.1
SWC(config)#exit
SWC#
%SYS-5-CONFIG_I: Configured from console by console

SWC#exit
```

Using show ip interface brief command it will list all the ip addresses associated with the switch and even vlan1

```
SWC>show ip interface brief
```

```
GigabitEthernet0/1     unassigned      YES manual down                  down
GigabitEthernet0/2     unassigned      YES manual down                  down
Vlan1                  10.101.117.2    YES manual up                    up
```

To check Router Configuration :

Click on Switch Router RTA → Go to CLI Tab → and type the following commands to check the configuration.

Name :Harsh Kadu Roll No : 33

```
RTA>show ip interface brief
Interface              IP-Address      OK? Method Status
Protocol
GigabitEthernet0/0     10.101.117.49   YES manual up
up
GigabitEthernet0/1     10.101.117.33   YES manual up
up
GigabitEthernet0/2     10.101.117.1    YES manual up
up
Vlan1                  unassigned      YES unset  administratively
down down
RTA>
```

Part 1 : Configure, Apply and Verify an Extended Numbered ACL.

Step 1 : Configure the extended ACL.

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).

Click on Router RTA → Go to CLI Tab → and type the following command:

```
RTA>enable
RTA#conf t
Enter configuration commands, one per line.  End with CNTL/Z.
RTA(config)#access-list 199 permit tcp 10.101.117.32 0.0.0.15 10.101.117.0
0.0.0.31 eq telnet
RTA(config)#access-list 199 permit icmp any any
```

Step 2 : Apply the Extended ACL.

Click on Router RTA → Go to CLI Tab → and type the following command:

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

Step 3 : Verify the extended ACL implementation.

a. Ping from **PCB** to all of the other IP addresses in the network.

Click on PCB → Go to Desktop option → and check the connection:

```
PC>ping 10.101.117.2

Pinging 10.101.117.2 with 32 bytes of data:

Reply from 10.101.117.2: bytes=32 time=0ms TTL=254
Reply from 10.101.117.2: bytes=32 time=0ms TTL=254
Reply from 10.101.117.2: bytes=32 time=0ms TTL=254
Reply from 10.101.117.2: bytes=32 time=0ms TTL=254

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

PC>ping 10.101.117.35

Pinging 10.101.117.35 with 32 bytes of data:

Reply from 10.101.117.35: bytes=32 time=7ms TTL=128
Reply from 10.101.117.35: bytes=32 time=0ms TTL=128
Reply from 10.101.117.35: bytes=32 time=13ms TTL=128
Reply from 10.101.117.35: bytes=32 time=11ms 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 = 0ms, Maximum = 13ms, Average = 7ms

PC>ping 10.101.117.51

Pinging 10.101.117.51 with 32 bytes of data:

Reply from 10.101.117.51: bytes=32 time=0ms TTL=127
Reply from 10.101.117.51: bytes=32 time=1ms TTL=127
Reply from 10.101.117.51: bytes=32 time=0ms 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 = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 0ms, Maximum = 1ms, Average = 0ms
```

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

Click on PCB → Go to Desktop option → and type the following command:

```
PC>telnet 10.101.117.2
Trying 10.101.117.2 ...Open


User Access Verification

Username: Admin
Password:
SWC>

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

And then Exit the telnet session to SWC.

c. Telnet from **PCA** to **SWC.** The access list causes the router to reject the connection. Click on PCA → Go to Desktop option → and type the following command:

```
Packet Tracer PC Command Line 1.0
PC>telnet 10.101.117.2
Trying 10.101.117.2 ...
% Connection timed out; remote host not responding
PC>
```

Part 2 : Reflection Questions

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

➢ Two steps were used: First, PCA used Telnet to access SWB. From SWB, Telnet was allowed to SWC.

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

➢ Because it was requested to block all traffic to 10.101.117.0/27 except Telnet traffic originating from 10.101.117.32/28 the access list could be written as is. Instead of applying the ACL to G0/2 outbound apply the same ACL to both G0/0 and G0/1 inbound.