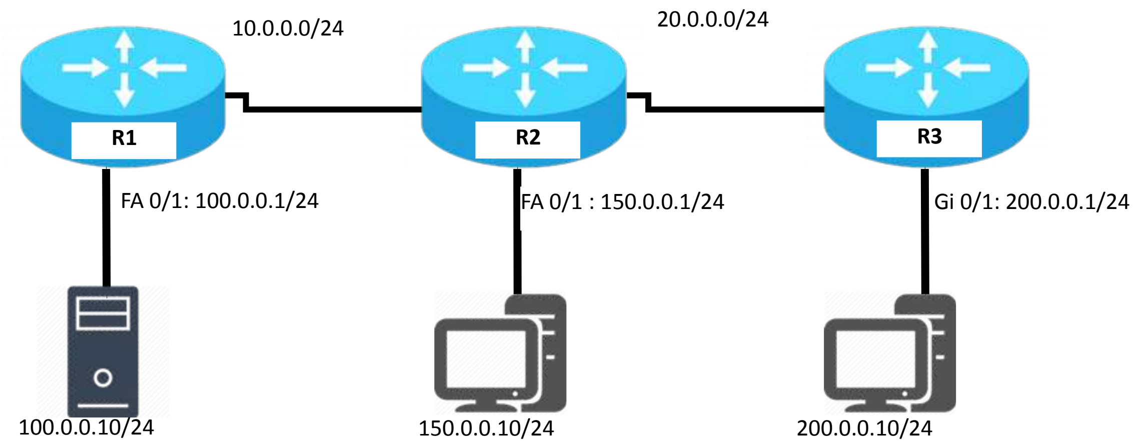
SCENARIO IP ACCESS-LIST

TOPOLOGY DIAGRAM



CASE – 1 [Standard IP Access List]

Objective

To block network 200.0.0.0/24 from accessing the 100.0.0.0/24 network, we would create the following access-list on Router R1 [Closest to the destination]

Commands

```
R1(config)# access-list 1 deny 200.0.0.0 0.0.0.255
R1(config)# access-list 1 permit any
R1(config)# int s 0/0/0
R1(config-if)# ip access-group 10 in
```

Verification

```
R1# show ip access-list
R1# show ip interface
R1# show running-config
```

CASE – 2 [Extended IP Access List]

Objective

Assume there is a web server on the 100.0.0.0/24 network with an IP address of 100.0.0.10. In order to block network 200.0.0.0/24 from accessing anything on the 100.0.0.0 network, EXCEPT for the HTTP port on the web server, we would create the following access-list on Router R3 [closest to the source network]

Commands

```
R3(config)# access-list 101 permit tcp 200.0.0.0 0.0.0.255 host 100.0.0.10 eq 80
R3(config)# access-list 101 deny ip 200.0.0.0 0.0.0.255 100.0.0.0 0.0.0.255
R3(config)# access-list 101 permit ip any any
R3(config)# int G0/1
R3(config-if)# ip access-group 101 in
```

CASE – 3 [ICMP Access List]

Objective

Consider the above scenario. You've been asked to block anyone from the 150.0.0.0/24 network from "pinging" anyone on the 100.0.0.0/24 network. You want to allow everything else, including all other ICMP packets.

Commands

```
R2(config)# access-list 102 deny icmp 150.0.0.0 0.0.0.255 100.0.0.0 0.0.255.255 echo
R2(config)# access-list 102 permit icmp 150.0.0.0 0.0.0.255 100.0.0.0 0.0.0.255
R2(config)# access-list 102 permit ip any any
```

```
R2(config)# int fa0/1
R2(config-if)# ip access-group 102 in
```

CASE – 4 [Telnet Access List]

Objective

Consider the above scenario. Create an access list that prevents anyone from the 150.0.0.0/24network from telneting into Router R1, but allow all other networks telnet access.

Commands

```
R1(config)# access-list 5 deny 150.0.0.0 0.0.0.255
R1(config)# access-list 5 permit any
R1(config)# line vty 0 4
R1(config-line)# access-class 5 in
```

CASE – 5 [Named – Standard IP Access List]

Objective

To block network 200.0.0.0/24 from accessing the 100.0.0.0/24 network, we would create the following access-list on Router R1 [Closest to the destination]

Commands

```
R1(config)# ip access-list standard ipss
R1(config-std-nacl)# deny 200.0.0.0 0.0.0.255
R1(config-std-nacl)# permit any
R1(config)# int s 0/0/0
R1(config-if)# ip access-group ipss in
```

Verification

```
R1# show ip access-list
R1# show ip interface
R1# show running-config
```

CASE – 6 [Named - Extended IP Access List]

Objective

Assume there is a web server on the 100.0.0.0/24 network with an IP address of 100.0.0.10. In order to block network 200.0.0.0/24 from accessing anything on the 100.0.0.0 network, EXCEPT for the HTTP port on the web server, we would create the following access-list on Router R3 [closest to the source network]

Commands

```
R3(config)# ip access-list extended ipss
R3(config-ext-nacl)# permit tcp 200.0.0.0 0.0.0.255 host 100.0.0.10 eq 80
R3(config-ext-nacl)# deny ip 200.0.0.0 0.0.0.255 100.0.0.0 0.0.0.255
R3(config-ext-nacl)# permit ip any any
R3(config)# int G0/1
R3(config-if)# ip access-group ipss in
```

CASE – 6 [Time-Based Access-Lists]

Objective

Access-lists can be based on the time and the day of the week. The first step to creating a time-based access-list is to create a time-range. Next, we must either specify an absolute time, or a periodic time:

Commands

```
Router(config)# time-range BLOCKHTTP
Router(config-time-range)# absolute start 10:00 10 Jan 2019 end 10:00 10 Jan 2020
Router(config-time-range)# periodic weekdays 10:00 to 16:00
```

After we establish our time-range, we must reference it in an access-list:

```
Router(config)# access-list 102 deny any any eq 80 time-range BLOCKHTTP
Router(config)# access-list 102 permit ip any any
```

IP ACCESS-LIST RULES

- ACLs are always processed from top to down in sequential order.
- A packet is compared with ACL conditions until it finds a match.
- Once a match is found for packet, no further comparison will be done for that packet.
- Interface will take action based on match condition. There are two possible actions; permit and Deny.
- If permit condition match, packet will be allowed to pass from interface.
- If deny condition match, packet will be destroyed immediately.
- Every ACL has a default deny statement at end of it.
- If a packet does not meet with any condition, it will be destroyed (by the last deny condition).
- Empty ACL will permit all traffic by default. Implicit deny condition will not work with empty ACL.
- Implicit (default last deny) condition would work only if ACL has at least one user defined condition.
- ACL can filter only the traffic passing from interface. It cannot filter the traffic originated from router on which it has been applied.
- Standard ACL can filter only the source IP address.
- Standard ACL should be placed near the destination devices.
- Extended ACL should be placed near the source devices.
- Each ACL needs a unique number or name.
- We can have only one ACL applied to an interface in each direction; inbound and outbound