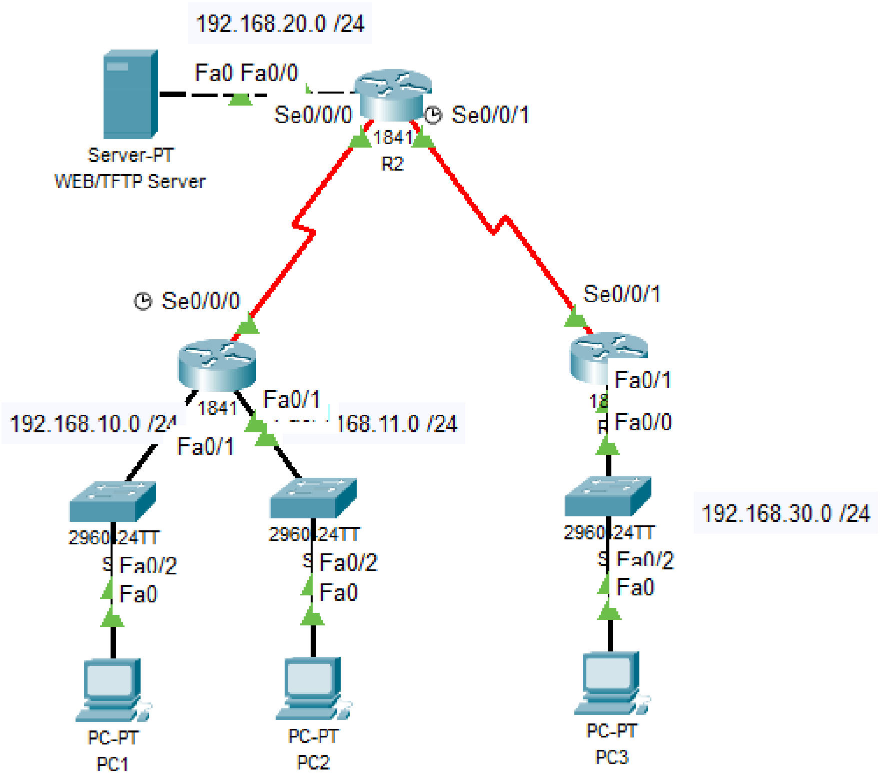
LAB_8

Cisco ACL Configuration Lab Walkthrough

This document provides a step-by-step guide for configuring and troubleshooting Access Control Lists (ACLs) on Cisco devices.

Network Architecture



Addressing Table

Device	Interface	IP Address	Subnet Mask	Default Gateway
R1	Fa0/0	192.168.10.1	255.255.255.0	N/A
	Fa0/1	192.168.11.1	255.255.255.0	N/A
	S0/0/0	10.1.1.1	255.255.255.252	N/A
R2	Fa0/0	192.168.20.1	255.255.255.0	N/A
	S0/0/0	10.1.1.2	255.255.255.252	N/A
	S0/0/1	10.2.2.1	255.255.255.252	N/A
	Lo0	209.165.200.225	255.255.255.224	N/A
R3	Fa0/0	192.168.30.1	255.255.255.0	N/A
	S0/0/1	10.2.2.2	255.255.255.252	N/A
PC1	NIC	192.168.10.10	255.255.255.0	192.168.10.1
PC2	NIC	192.168.11.10	255.255.255.0	192.168.11.1
PC3	NIC	192.168.30.10	255.255.255.0	192.168.30.1
Web Server	NIC	192.168.20.254	255.255.255.0	192.168.20.1

Task 1: Perform Basic Router Configurations

Step 1. Configure the routers and switches.

R1 Configuration:

```
enable
configure terminal
hostname R1
no ip domain-lookup
enable secret class
banner motd #Unauthorized access is prohibited!#
line console 0
 password cisco
 login
line vty 0 15
 password cisco
 login

interface FastEthernet0/0
 ip address 192.168.10.1 255.255.255.0
 no shutdown
interface FastEthernet0/1
 ip address 192.168.11.1 255.255.255.0
 no shutdown
interface Serial0/0/0
 ip address 10.1.1.1 255.255.255.252
 clock rate 64000
 no shutdown

 ip route 192.168.20.0 255.255.255.0 10.1.1.2
 ip route 192.168.30.0 255.255.255.0 10.1.1.2
 ip route 209.165.200.224 255.255.255.224 10.1.1.2
end
write memory
```

R2 Configuration:

R3 Configuration:

Step 2. Configure the PCs and WEB/TFTP Server.

- PC1 IP Configuration: IP: 192.168.10.10, Mask: 255.255.255.0, GW: 192.168.10.1

- PC2 IP Configuration: IP: 192.168.11.10, Mask: 255.255.255.0, GW: 192.168.11.1
- PC3 IP Configuration: IP: 192.168.30.10, Mask: 255.255.255.0, GW: 192.168.30.1
- Web Server IP Configuration: IP: 192.168.20.254, Mask: 255.255.255.0, GW: 192.168.20.1

Step 3. Check results.

- Verify full IP connectivity using `ping` commands from various devices.

Task 2: Configuring a Standard ACL

Step 1. Create the ACL.

On R3 (blocks traffic from 192.168.11.0/24, permits others):

```
R3(config)# ip access-list standard std-1
R3(config-std-nacl)# deny 192.168.11.0 0.0.0.255
R3(config-std-nacl)# permit any
```

Step 2. Apply the ACL.

On R3 (apply inbound on S0/0/1):

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

Step 3. Test the ACL.

- From PC2, ping PC3 (should fail).
- From PC1, ping PC3 (should succeed).
- On R3, verify matches: `show access-lists`

Task 3: Configuring an Extended ACL

Step 1. Configure a named extended ACL.

On R1 (blocks 192.168.10.0/24 to 209.165.200.225, permits others):

```
R1(config)# ip access-list extended extend-1
R1(config-ext-nacl)# deny ip 192.168.10.0 0.0.0.255 host 209.165.200.225
R1(config-ext-nacl)# permit ip any any
```

Step 2. Apply the ACL.

On R1 (apply outbound on S0/0/0):

```
R1(config)# interface serial 0/0/0
R1(config-if)# ip access-group extend-1 out
```

Step 3. Test the ACL.

- From PC1, ping R2's Loopback0 (209.165.200.225) (should fail).
- From PC1, ping PC3 (should succeed).
- On R1, verify matches: `show ip access-list`

Task 4: Control Access to the Web Server (HTTP/Port 80) with an Extended ACL

Step 1. Configure the ACL.

On R1 (blocks PC1 HTTP access to Web Server):

```
R2(config)# ip access-list extended Block_HTTP
R2(config-ext-nacl)# deny tcp host 192.168.10.10 host 192.168.20.254 eq www
R2(config-ext-nacl)# permit ip any any
```

Step 2. Apply the ACL.

On R2 (apply inbound on Fa0/0):

```
R2(config)# interface Serial0/0/0
R2(config-if)# ip access-group Block_HTTP out
```

Step 3. Test the ACL.

- From PC1, access Web Server via browser (should fail).
- From PC3, access Web Server via browser (should succeed).
- From PC1, ping Web Server (should succeed).
- On R2, verify matches: `show ip access-list`

Task 5: Troubleshooting ACLs

Step 1. Test the ACL.

On R3, view ACL: show running-config | section access-list.

On R3, view interface config: show running-config interface Serial0/0/1.

On R3, remove ACL from inbound S0/0/1:

```
R3(config)# interface serial 0/0/1
R3(config-if)# no ip access-group std-1 in
```

Verify removal: `do show running-config interface Serial0/0/1`.

Step 2. Apply ACL std-1 on S0/0/1 outbound.

On R3, reapply ACL outbound on S0/0/1:

```
R3(config-if)# ip access-group std-1 out
```

Step 3. Test the ACL.

- From PC2, ping PC3 (should succeed).
- On R3, observe ACL counters (should not increment for this traffic): `show ip access-list`.

Step 4. Restore the ACL to its original configuration.

On R3, remove outbound ACL and reapply inbound:

```
R3(config)# interface serial 0/0/1
R3(config-if)# no ip access-group std-1 out
R3(config-if)# ip access-group std-1 in
```

Step 6. Test the ACL.

- Attempt communication from R1 to R2/R3 networks (should be blocked).
- On R2, observe ACL counters (may not increment for implicitly denied traffic): `show ip access-list`.