## Access Control Lists (ACLs)

### 1. Objectives

* Use of **Access Control Lists (ACLs)** defined in Routers to control access in a network.
* Difference between **Standard** and **Extended ACLs**.
* Configuring and applying **Standard** and **Extended ACLs** in Cisco routers.

### 2a. Access Control Lists (ACLs)

### The Access Control List (ACL) is a collection of security rules or policies that allows or denies packets after looking at the packet headers and other attributes. Each permit or deny statement in the ACL is referred to as an access control entry (ACE). These ACEs can classify packets by inspecting Layer 2 through Layer 4 headers for a number of parameters, including the following:

### ■ Layer 2 protocol information such as EtherTypes

### ■ Layer 3 protocol information such as ICMP, TCP, or UDP

### ■ Layer 3 header information such as source and destination IP addresses

### ■ Layer 4 header information such as source and destination TCP or UDP ports

### After an ACL has been properly configured, you can apply it to an interface to filter traffic. The security appliance can filter packets in both the inbound and outbound direction on an interface. When an inbound ACL is applied to an interface, the security appliance analyzes packets against the ACEs after receiving them. If a packet is permitted by the ACL, the firewall continues to process the packet and eventually passes the packet to the defined interface.

### 2b. Standard ACLs

**Standard access control lists (ACLs)** are router configuration scripts that control whether a router permits or denies packets based on the **source address only**. Tasks are: **defining filtering criteria**, **configuring standard ACLs**, **applying ACLs to router interfaces**, and **verifying and testing the ACL implementation**.

**Example:**

* + 1. Create an ACL using the number 1 on **R2** with a statement that denies access to the 192.168.20.0/24 network from the 192.168.11.0/24 network.

R2(config)# **access-list 1 deny 192.168.11.0 0.0.0.255**

* + 1. By default, an access list denies all traffic that does not match a rule. To permit all other traffic, configure the following statement:

R2(config)# **access-list 1 permit any**

* + 1. For the ACL to actually filter traffic, it must be applied to some router operation. Apply the ACL by placing it for outbound traffic on the Gigabit Ethernet 0/0 interface.

R2(config)# **interface GigabitEthernet0/0**

R2(config-if)# **ip access-group 1 out**

### 2c. Extended ACLs

**Extended access control lists (ACLs)** are extremely powerful. They offer a much greater degree of control than standard ACLs as to the types of traffic that can be filtered, as well as where the traffic originated and where it is going. Extended ACLs can filter traffic in many different ways. Extended ACLs can filter on **source IP addresses**, **source ports**, **destination IP addresses**, **destination ports**, as well as **various protocols and services**. Tasks are: **defining filtering criteria**, **configuring extended ACLs**, **applying ACLs to router interfaces**, and **verifying and testing the ACL implementation**.

* + 1. Create two access list statements to permit tcp for accessing FTP server at 172.22.34.62 and permit ICMP (ping, etc.) traffic from **172.22.34.64/27** network to **Server** at 172.22.34.62. Note that the access list number remains the same and a specific type of ICMP traffic does not need to be specified.

R1(config)# **access-list 100 permit icmp 172.22.34.64 0.0.0.31 host 172.22.34.62**

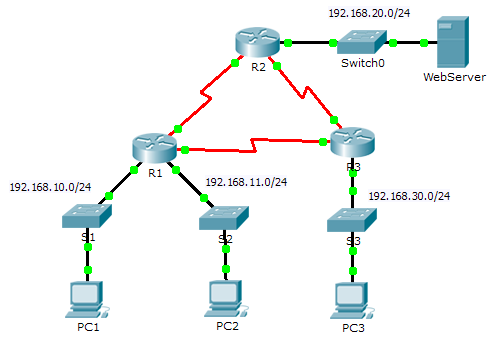
R1(config)# **access-list 100 permit tcp 172.22.34.64 0.0.0.31 host 172.22.34.62 eq ftp**

* + 1. Enter interface configuration mode and apply the ACL.

R1(config)# **interface gigabitEthernet 0/0**

R1(config-if)# **ip access-group 100 in**

1. Topology



1. Addressing Table

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Device | Interface | IP Address | Subnet Mask | Default Gateway |
| R1 | F0/0 | 192.168.10.1 | 255.255.255.0 | N/A |
| F0/1 | 192.168.11.1 | 255.255.255.0 | N/A |
| S0/0/0 | 10.1.1.1 | 255.255.255.252 | N/A |
| S0/0/1 | 10.3.3.1 | 255.255.255.252 | N/A |
| R2 | F0/0 | 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 |
| R3 | F0/0 | 192.168.30.1 | 255.255.255.0 | N/A |
| S0/0/0 | 10.3.3.2 | 255.255.255.252 | 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 |
| WebServer | NIC | 192.168.20.254 | 255.255.255.0 | 192.168.20.1 |

1. Objectives

Part 1: Plan an ACL Implementation

Part 2: Configure, Apply, and Verify a Standard ACL

1. Background / Scenario

Standard access control lists (ACLs) are router configuration scripts that control whether a router permits or denies packets based on the source address. This activity focuses on defining filtering criteria, configuring standard ACLs, applying ACLs to router interfaces, and verifying and testing the ACL implementation. The routers are already configured, including IP addresses and Enhanced Interior Gateway Routing Protocol(EIGRP) routing.

1. Plan an ACL Implementation
   1. Investigate the current network configuration.

Before applying any ACLs to a network, it is important to confirm that you have full connectivity. Verify that the network has full connectivity by choosing a PC and pinging other devices on the network. You should be able to successfully ping every device.

* 1. Evaluate two network policies and plan ACL implementations.
     1. The following network policies are implemented on **R2**:
* The 192.168.11.0/24 network is not allowed access to the **WebServer** on the 192.168.20.0/24 network.
* All other access is permitted.

To restrict access from the 192.168.11.0/24 network to the **WebServer** at 192.168.20.254 without interfering with other traffic, an ACL must be created on **R2**. The access list must be placed on the outbound interface to the **WebServer**. A second rule must be created on **R2** to permit all other traffic.

* + 1. The following network policies are implemented on **R3**:
* The 192.168.10.0/24 network is not allowed to communicate to the 192.168.30.0/24 network.
* All other access is permitted.

To restrict access from the 192.168.10.0/24 network to the 192.168.30/24 network without interfering with other traffic, an access list will need to be created on **R3**. The ACL must placed on the outbound interface to **PC3**. A second rule must be created on **R3** to permit all other traffic.

1. Configure, Apply, and Verify a Standard ACL
   1. Configure and applya numbered standard ACL on R2.
      1. Create an ACL using the number 1 on **R2** with a statement that denies access to the 192.168.20.0/24 network from the 192.168.11.0/24 network.

R2(config)# **access-list 1 deny 192.168.11.0 0.0.0.255**

* + 1. By default, an access list denies all traffic that does not match a rule. To permit all other traffic, configure the following statement:

R2(config)# **access-list 1 permit any**

* + 1. For the ACL to actually filter traffic, it must be applied to some router operation. Apply the ACL by placing it for outbound traffic on the Gigabit Ethernet 0/0 interface.

R2(config)# **interface GigabitEthernet0/0**

R2(config-if)# **ip access-group 1 out**

* 1. Configure and apply a numbered standard ACL on R3.
     1. Create an ACL using the number 1 on **R3** with a statement that denies access to the 192.168.30.0/24 network from the **PC1** (192.168.10.0/24) network.

R3(config)# **access-list 1 deny 192.168.10.0 0.0.0.255**

* + 1. By default, an ACL denies all traffic that does not match a rule. To permit all other traffic, create a second rule for ACL 1.

R3(config)# **access-list 1 permit any**

* + 1. Apply the ACL by placing it for outbound traffic on the Gigabit Ethernet 0/0 interface.

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

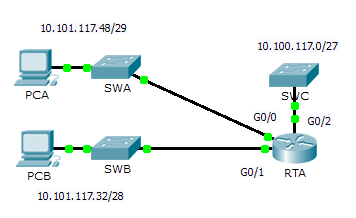
R3(config-if)# **ip access-group 1 out**

* 1. Verify ACL configuration and functionality.
     1. On **R2** and **R3**, enter the **show access-list** command to verify the ACL configurations. Enter the **show run** or **show ip interface gigabitethernet 0/0** command to verify the ACL placements.
     2. With the two ACLs in place, network traffic is restricted according to the policies detailed in Part 1. Use the following tests to verify the ACL implementations:
* A ping from 192.168.10.10 to 192.168.11.10 succeeds.
* A ping from 192.168.10.10 to 192.168.20.254 succeeds.
* A ping from 192.168.11.10 to 192.168.20.254 fails.
* A ping from 192.168.10.10 to 192.168.30.10 fails.
* A ping from 192.168.11.10 to 192.168.30.10 succeeds.
* A ping from 192.168.30.10 to 192.168.20.254 succeeds.

### 2c. Extended ACLs

Extended access control lists (ACLs)

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 |
| SWC | VLAN1 | 10.101.117.2 | 255.255.255.224 | 10.101.117.1 |

Objectives

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

Part 2: Reflection Questions

Background / Scenario

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

1. Configure, Apply and Verify an Extended Numbered ACL

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

* Telnet traffic from devices on the 10.101.117.32/28 network is allowed to devices on the 10.100.117.0/27 networks.
* ICMP traffic is allowed from any source to any destination
* All other traffic is blocked.
  1. Configure the extended ACL.
     1. 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 Telnet.

What is the first ACL statement?

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

* + 1. 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)

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

* + 1. All other IP traffic is denied, by default.
  1. Apply the extended ACL.

The general rule is to place extended ACLs close to the source. However, since 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?

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

* 1. Verify the extended ACL implementation.
     1. Ping from**PCB** to all of the other IP addresses in the network. If the pings are unsuccessful, verify the IP addresses before continuing.
     2. Telnet from **PCB** to **SWC**. The password is **cisco**.
     3. Exit the Telnet service of the **SWC**.
     4. Ping from**PCA** to all of the other IP addresses in the network. If the pings are unsuccessful, verify the IP addresses before continuing.
     5. Telnet from **PCA** to **SWC**. The access list causes the router to reject the connection.
     6. Telnet from **PCA** to **SWB**. The access list is placed on **G0/2** and does not affect this connection.
     7. After logging into **SWB,** do not log out. Telnet to **SWC**.

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

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

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

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**Demonstrate your work to the instructors.** **Signature of the Instructor**