**1. Introduction to Access Control Lists (ACLs)**

* **Purpose:**

ACLs function like a stateless firewall by filtering packets at **Layer 3 (Network)** and **Layer 4 (Transport)**.

* **Features:**
  + Filter traffic based on source/destination IP, port, and protocol.
  + Consist of **Access Control Entries (ACEs)**.
  + Checked sequentially—no match leads to an **implicit "deny all"**.

**2. Types of ACLs**

* **Standard ACLs:**
  + Filter traffic based on **source IP address** only.
  + Range: **1–99** or **1300–1999**.
  + Applied close to the **destination**.
* **Extended ACLs:**
  + More advanced; filter based on **source/destination IP**, port, or protocol.
  + Range: **100–199** or **2000–2699**.
  + Applied close to the **source**.

**3. Operation of ACLs**

* **Inbound ACLs:**
  + Traffic is filtered **before routing lookup**.
  + A **permit** allows routing; a **deny** discards the packet.
* **Outbound ACLs:**
  + Traffic is filtered **after routing lookup**.
  + A **permit** sends the packet out; a **deny** discards it.

**4. Using Wildcard Masks**

* **Definition:**

Wildcard masks determine the range of IP addresses filtered.

* **Calculation:**

**Wildcard Mask = 255.255.255.255 - Subnet Mask.**

* + Example: For **192.168.10.16/28**, wildcard mask = 0.0.0.15.
  + **Permits range**: **192.168.10.16 to 192.168.10.31**.

**5. ACL Syntax and Configuration**

* **Standard ACL Syntax:**

**R1(config)# access-list <number> [permit|deny] <source IP> <wildcard mask>**

* **Named ACL Syntax:**

**R1(config)# ip access-list standard <name>**

**R1(config-std-nacl)# {deny|permit} <source IP> <wildcard mask>**

* **Application to Interfaces:**

**R1(config)# interface s0/0/0**

**R1(config-if)# ip access-group <ACL number/name> [in|out]**

**6. Examples of ACLs**

* **Numbered ACL:**

**Permit traffic from 192.168.30.0/24:**

**R1(config)# access-list 10 permit 192.168.30.0 0.0.0.255**

**R1(config)# access-list 10 deny any**

* **Apply to an interface:**

**R1(config)# interface s0/0/0**

**R1(config-if)# ip access-group 10 in**

* **Named ACL:**
  + **Deny a specific host:**

**R1(config)# ip access-list standard TEST**

**R1(config-std-nacl)# deny host 192.168.30.1**

**R1(config-std-nacl)# permit any**

**R1(config)# interface s0/0/0**

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

**7. Placement of ACLs**

* **Standard ACLs:**
* Place close to the **destination** to avoid filtering legitimate traffic.
* **Extended ACLs:**
* Place close to the **source** to block unnecessary traffic early.

**8. Securing VTY Lines**

* **Example:**
  + **Allow Telnet/SSH from 192.168.30.0/24:**

**R1(config)# access-list 10 permit 192.168.30.0 0.0.0.255**

**R1(config)# access-list 10 deny any**

**R1(config)# line vty 0 15**

**R1(config-vty)# login**

**R1(config-vty)# password secret**

**R1(config-vty)# access-class 10 in**

**9. Editing and Viewing ACLs**

**R1# show access-lists**

* **To insert a specific rule, use line numbers:**

**R1(config)# ip access-list 1**

**R1(config-std-nacl)# 15 deny 192.168.2.0 0.0.0.31**

**1. Standard Numbered ACL Configuration**

* **Create a numbered ACL to permit or deny certain traffic:**

**R1(config)#** **access-list 10 permit 192.168.30.0 0.0.0.255**

**R1(config)#** **access-list 10 deny any**

* **Apply the ACL to an interface in the inbound direction**:

**R1(config)#** **interface s0/0/0**

**R1(config-if)#** **ip access-group 10 in**

**2. Permit or Deny Traffic with Specific Rules**

* **Allow all traffic:**

**R1(config)#** **access-list 1 permit 0.0.0.0 255.255.255.255**

**Alternatively, the following is valid as well:**

# R1(config)# access-list 1 permit any

* **Deny traffic from specific subnets and permit all other traffic:**

**R1(config)#** **access-list 10 deny 192.168.30.0 0.255.255.255**

**R1(config)#** **access-list 10 permit any**

* **Deny traffic from a specific host:**

**R1(config)#** **access-list 10 deny 192.168.30.1 0.0.0.0**

* **Deny a specific host AND permit all other traffic:**

**R1(config)#** **access-list 10 deny host 192.168.30.1**

**R1(config)#** **access-list 10 permit any**

**3. Named ACL Configuration**

* **Create a named ACL with multiple rules:**

**R1(config)#** **ip access-list standard TEST**

**R1(config-std-nacl)#** **deny host 192.168.30.1**

**R1(config-std-nacl)#** **permit any**

* **Apply the named ACL to an interface:**

**R1(config)#** **interface s0/0/0**

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

**4. Securing VTY Lines**

* **Allow Telnet/SSH access to a specific network:**

**R1(config)#** **access-list 10 permit 192.168.30.0 0.0.0.255**

**R1(config)#** **access-list 10 deny any**

* **Apply the ACL to all VTY lines:**

**R1(config)#** **line vty 0 15**

**R1(config-vty)#** **login**

**R1(config-vty)#**  **password secret**

**R1(config-vty)#** **access-class 10 in**

**5. Editing and Viewing ACLs**

* **Display all configured access control lists:**

**R1#** **show access-lists**

* **Insert an additional rule into an existing ACL:**

**R1(config)#** **ip access-list 1**

**R1(config-std-nacl)#** **15 deny 192.168.2.0 0.0.0.31**

**6. Wildcard Mask Calculation and Examples**

* **Wildcard mask calculation:**

Wildcard Mask = 255.255.255.255 - Subnet Mask

* Example for network **192.168.10.16/28**:

**R1(config)#** **access-list 10 permit 192.168.10.16 0.0.0.15**