# **35. EXTENDED ACCESS CONTROL LISTS (EACL)**

## **ANOTHER WAY TO CONFIGURE NUMBERED ACLs**

### **Numbered ACLs in Global Config Mode**

In **DAY 34**, you learned that numbered ACLs are configured in **Global Config mode**:

### **Named ACLs in Separate Config Mode**

You also learned that named ACLs are configured with subcommands in a separate config mode.

### **Modern IOS Configuration**

In modern **IOS**, you can also configure numbered ACLs in the exact same way as named ACLs.

## **ADVANTAGES OF NAMED ACL CONFIG MODE**

* You can easily **DELETE** individual entries in the ACL without specifying an **entry-number** or **sequence-number**.
* You can **insert new entries** between other entries by specifying the **SEQUENCE NUMBER**.

## **RESEQUENCING ACLs**

* There is a **resequencing** function that helps edit ACLs.

The command is:  
 R1(config)# ip access-list resequence \*acl-id starting-seq-num increment\*

## 

## **EXTENDED NUMBERS AND NAMED ACLs**

* **Extended ACLs** function similarly to **Standard ACLs** but can be **NUMBERED or NAMED**.
* **Numbered ACLs** use the following ranges: **100 - 199, 2000 - 2699**.
* Processed from **TOP to BOTTOM**, just like **Standard ACLs**.
* Extended ACLs match traffic based on more parameters:
  + **Layer 4 Protocol / Port**
  + **Source Address**
  + **Destination Address**

### **Extended Numbered ACL**

R1(config)# access-list \*number\* [permit | deny] \*protocol src-ip dest-ip\*

### **Extended Named ACL**

R1(config)# ip access-list extended {name | number}

R1(config-ext-nacl)# {seq-num} {permit | deny} \*protocol src-ip dest-ip\*

## **MATCHING THE PROTOCOL**

* The **IP Protocol Number** is the number used in the **IPv4 Header Protocol field**.
* Examples:
  + (1) ICMP
  + (6) TCP
  + (17) UDP
  + (88) EIGRP
  + (89) OSPF

## **MATCHING THE SOURCE / DESTINATION IP ADDRESS**

Example command:

R1(config-ext-nacl)# deny tcp any 10.0.0.0 0.0.0.255

This denies **all packets** encapsulating a **TCP segment** from **any source** to **10.0.0.0/24**.

## **PRACTICE QUESTIONS:**

1. **ALLOW ALL TRAFFIC**

R1(config-ext-nacl)# permit ip any any

1. **PREVENT 10.0.0.0/16 from SENDING UDP traffic to 192.168.1.1/32**

R1(config-ext-nacl)# deny udp 10.0.0.0 0.0.255.255 host 192.168.1.1

1. **PREVENT 172.16.1.1/32 from pinging hosts in 192.168.0.0/24**

R1(config-ext-nacl)# deny icmp host 172.16.1.1 192.168.0.0 0.0.0.255

## **MATCHING THE TCP / UDP PORT NUMBERS**

* When matching **TCP/UDP**, you can optionally specify the **SOURCE** and/or **DESTINATION PORT NUMBERS**.
* Common operators:
  + **eq** = equal to
  + **gt** = greater than
  + **lt** = less than
  + **neq** = not equal to
  + **range** = range of ports
* You can use either the **port number** or the **well-known port name**.

## **PRACTICE QUESTIONS 2:**

1. **ALLOW TRAFFIC from 10.0.0.0/16 to access the server at 2.2.2.2/32 using HTTPS**

R1(config-ext-nacl)# permit tcp 10.0.0.0 0.0.255.255 host 2.2.2.2 eq 443

1. **PREVENT ALL HOSTS using SOURCE UDP Port Numbers from 20000 to 30000 from accessing the server at 3.3.3.3/32**

R1(config-ext-nacl)# deny udp any range 20000 30000 host 3.3.3.3

1. **ALLOW HOSTS in 172.16.1.0/24 using a TCP SOURCE Port greater than 9999 to access ALL TCP ports on server 4.4.4.4/32 EXCEPT port 23**

R1(config-ext-nacl)# permit tcp 172.16.1.0 0.0.0.255 gt 9999 host 4.4.4.4 neq 23

## **EXAMPLE NETWORK REQUIREMENTS**

1. **Hosts in 192.168.1.0/24 can’t use HTTPS to access SRV1**.
2. **Hosts in 192.168.2.0/24 can’t access 10.0.2.0/24**.
3. **No hosts in 192.168.1.0/24 or 192.168.2.0/24 can ping 10.0.1.0/24 or 10.0.2.0/24**.

### **EXTENDED ACL #1 (Applied at R1 G0/1 INBOUND interface)**

R1(config)# ip access-list extended HTTP\_SRV1

R1(config-ext-nacl)# deny tcp 192.168.1.0 0.0.0.255 host 10.0.1.100 eq 443

R1(config-ext-nacl)# permit ip any any

R1(config-ext-nacl)# int g0/1

R1(config-if)# ip access-group HTTP\_SRV1 in

### **EXTENDED ACL #2 (Applied at R1 G0/2 INBOUND interface)**

R1(config)# ip access-list extended BLOCK\_10.0.2.0

R1(config-ext-nacl)# deny ip 192.168.2.0 0.0.0.255 10.0.2.0 0.0.0.255

R1(config-ext-nacl)# permit ip any any

R1(config-ext-nacl)# int g0/2

R1(config-if)# ip access-group BLOCK\_10.0.2.0 in

### **EXTENDED ACL #3 (Applied at R1 g0/0 OUTBOUND interface)**

R1(config)# ip access-list extended BLOCK\_ICMP

R1(config-ext-nacl)# deny icmp 192.168.1.0 0.0.0.255 10.0.1.0 0.0.0.255

R1(config-ext-nacl)# deny icmp 192.168.1.0 0.0.0.255 10.0.2.0 0.0.0.255

R1(config-ext-nacl)# deny icmp 192.168.2.0 0.0.0.255 10.0.1.0 0.0.0.255

R1(config-ext-nacl)# permit ip any any

R1(config-ext-nacl)# int g0/0

R1(config-if)# ip access-group BLOCK\_ICMP out

## **HOW TO CHECK WHICH EXTENDED ACLs ARE APPLIED TO AN INTERFACE**

Use the following command:

show ip interface <interface>