An-Najah National University

Networks-Lab
Dr. Muhannad Al-Jabi
Thursday 8:00am – 2:00pm
Summer Semester

Experiment Information		
Experiment Name: ACLs	Experiment Number: #7	
Performed: 14 of July, 2021	Submitted: 24 of July, 2021	
Partner Students		
Taher Anaya	Haron Dwiekat	Ali Moalla

Introduction:

Access lists are essentially a set of conditions that control who has access to what information. They're capable of controlling access to and from network segments. They can be used to filter undesirable packets and to enforce security policies. Network administrators will implement practically any access policy they can think of with the appropriate combination of access lists.
We will deal in this experiment with 2 kinds of ACL's:

Standard access lists:

To filter the network, these solely employ the source IP address in an IP packet. This essentially allows or disallows a whole set of protocols.

Extended access Lists:

These examine the source and destination IP addresses and the protocol field in the Network layer header, and the port number in the Transport layer header.

Objectives:

Students will understand how to use the Standard/Extended Access lists to control access both to/and from network segments.

- On The Source.
- On The Destination.
- Filter specific IP services

Procedure:

Before We Start:

We should build lab as EXP4, then ensure the network:

- ✓ Connectivity between all segment devices
- ✓ You are required to make all the necessary Configuration,
- ✓ Use default static route, then make a Ping from PC1 to all devices.

Note: For more details on to how to build this part, see my EXP4-Report.

Standard IP Access Lists

In this part, we will allow only Host PC1 from network 10.0.11.0 to enter Network 10.0.7.0.

- ➢ Go to router 2611xm and enter global configuration by typing **conf t**.

Why you choose router 2611xm?

Because a standard access list should be placed as close to the destination as practicable, which is router 2611xm in this case.

- From global configuration mode, type **access-list?** To get a list of all the different access lists are available.

```
Router1(config)#access-list ?
  <1-99>            IP standard access list
  <100-199>         IP extended access list
  <1100-1199>       Extended 48-bit MAC address access list
  <1300-1999>       IP standard access list (expanded range)
  <200-299>         Protocol type-code access list
  <2000-2699>       IP extended access list (expanded range)
  <700-799>         48-bit MAC address access list
  dynamic-extended  Extend the dynamic ACL absolute timer
  rate-limit        Simple rate-limit specific access list
```

Write down 4 different ACLs with the corresponding number ranges?

You will find the required list in the previous image.

- Choose an access list number that will allow you to create an IP standard access list. (We choose 7).
- Type These Commands.
  - access-list 7 ?
  - access-list 7 permit ?
  - access-list 7 permit 10.0.11.2
  - access-list 7 permit 10.0.11.2 ?
  - access-list X permit 10.0.11.2 0.0.0.0

```
Router1(config)#access-list 7 ?
  deny    Specify packets to reject
  permit  Specify packets to forward
  remark  Access list entry comment

Router1(config)#access-list 7 permit ?
  Hostname or A.B.C.D  Address to match
  any                  Any source host
  host                 A single host address

Router1(config)#access-list 7 permit 10.0.11.2
Router1(config)#access-list 7 permit 10.0.11.2?
Hostname or A.B.C.D

Router1(config)#access-list 7 permit 10.0.11.2 0.0.0.0
Router1(config)#
```

- Now that the access list is created, so we must apply it to an interface to make it work by type these commands:

(config)#int f0/1

*(config-if)#ip access-group **X** out*

- Verify your access lists with the following commands:

```
Router1#sh access-lists
Standard IP access list 7
    10 permit 10.0.11.2
Router1#
```

Write down the results?

You will find the results in the previous image.

- Now, type this command: **sh run**

```
Router1#sh run
Building configuration...


Press RETURN to get

no service password-encryption6-INFO_LOC: Crypto engine: aim
!
hostname Router1
!
boot-start-marker
boot-end-marker
: In
!i
!i
no aaa new-model*Mar  1 00:01:16
no network-clock-participate slot 1ine: aim 0  State changed to
no network-clock-participate wic 0
!
!:
ip cefd
```

- Ping from Host PC1 (10.0.11.2) to Server (10.0.7.2)

```
Ping statistics for 10.0.7.2:
    Packets: Sent = 4, Received = 0, Lost = 4 (100% loss),

C:\Users\NetworksPC>ping 10.0.7.2

Pinging 10.0.7.2 with 32 bytes of data:
Reply from 10.0.7.2: bytes=32 time<1ms TTL=127
Reply from 10.0.7.2: bytes=32 time<1ms TTL=127
Reply from 10.0.7.2: bytes=32 time<1ms TTL=127
Reply from 10.0.7.2: bytes=32 time<1ms TTL=127

Ping statistics for 10.0.7.2:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 0ms, Maximum = 0ms, Average = 0ms
```

- Ping from routers and PC2 to server (10.0.7.2).

```
C:\Users\NetworksPC>ping 10.0.7.2

Pinging 10.0.7.2 with 32 bytes of data:
Reply from 10.0.12.2: Destination net unreachable.
Reply from 10.0.12.2: Destination net unreachable.
Reply from 10.0.12.2: Destination net unreachable.
Reply from 10.0.12.2: Destination net unreachable.

Ping statistics for 10.0.7.2:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),

C:\Users\NetworksPC>
```

Write Down your results? Comments?

The access list we established on router 2611xm was to block any host besides PC1 from accessing the 10.0.8.0/24 network, and that's exactly what happened, as shown in the previous two snapshots

Extended IP Access Lists

In this part, you will use an extended IP access list to stop host PC1 from creating a Telnet session to router 2611xm (10.0.12.2). However, the host still should be able to ping to router 2611xm. IP extended lists should be placed closest to the source, adding the extended list on router 2611.

- Remove any access lists on 2611 and add an extended list to 2611. Then enable the line vty (telnet) on router 2611xm

```
Router1#conf t
Enter configuration commands, one per line.  End with CNTL/Z.
Router1(config)#line vty 04
Router1(config-line)#
```

Write down the command?

line vty 0 4

- Choose a number to create an extended IP list (We choose 110)

```
Router1#conf t
Enter configuration commands, one per line.  End with CNTL/Z.
Router1(config)#line vty 04
```

What is the number range?

100 - 199

- Use a deny statement **access-list 110 deny?**

```
Router(config)#access-list 110 deny ?
  ahp    Authentication Header Protocol
  eigrp  Cisco's EIGRP routing protocol
  esp    Encapsulation Security Payload
  gre    Cisco's GRE tunneling
  icmp   Internet Control Message Protocol
  ip     Any Internet Protocol
  ospf   OSPF routing protocol
  tcp    Transmission Control Protocol
  udp    User Datagram Protocol
```

- Use a deny statement **access-list 110 tcp?**

```
Router(config)#access-list 110 deny tcp ?
  A.B.C.D  Source address
  any      Any source host
  host     A single source host
```

- Add the source IP address you want to filter on, and then add the destination host IP address. Use the host command instead of wildcard bits.

```
Router(config)#access-list 110 deny tcp host 10.0.1.2 host 10.0.2.2  ?
  dscp          Match packets with given dscp value
  eq            Match only packets on a given port number
  established   established
  gt            Match only packets with a greater port number
  lt            Match only packets with a lower port number
  neq           Match only packets not on a given port number
  precedence    Match packets with given precedence value
  range         Match only packets in the range of port numbers
  <cr>
```

- At this point, you can add the eq telnet command. The log command can also be used at the end of the command so that whenever the access-list line is hit, a log will be generated on the console.

```
Router1(config-line)#$tcp host 10.0.11.2 host 10.0.12.2 eq telnet log
```

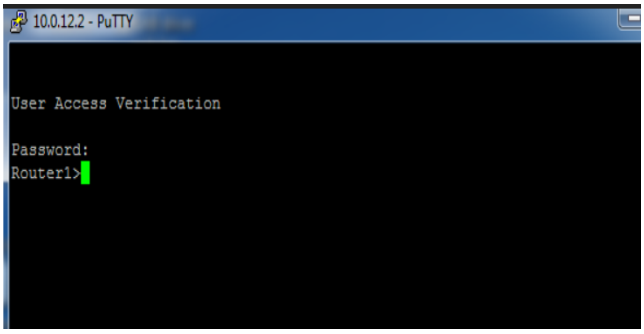
- It is important to add this line next to create a permit statement

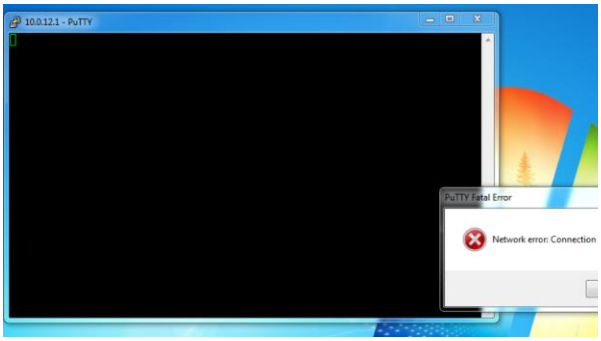
```
Router1(config)#access-list 110 permit ip any any
```

- Apply the access list to the f 0/0 on 2611router to stop the Telnet traffic as soon as it hits the first router interface.

```
RouterB(config)#int f0/0
RouterB(config-if)#ip access-group 110 in
RouterB(config-if)#^Z
```

- We test out work:





- **Conclusion.**

We learned about several sorts of access lists and their importance in this project. Also, how to set up an access list. We also learned how to use an access list to restrict access to a specific port. And we discovered that the expanded access list should be as close to the source as possible. On the other hand, the standard access list must be as close to the destination as practicable.