## LAB 3

|  |  |
| --- | --- |
| **3** | **Firewall Implementation, Router Access Control List (ACL)** |

#### Objective(s):

Understand the router firewall: Access Control Lists (ACLs).

#### Background:

Packet filtering at the network level can be achieved by applying the Access Control Lists (ACLs)1 at the router called router firewall. ACLs at the router filter the inbound traffic while it permit or deny packets based on source IP/network and destination IP/network, IP, TCP,UDP protocol information. Generally we use the ACLs to provide a basic level of security for accessing our network. Access lists can allow one host to access a part of network and prevent another host from accessing the same area.

A standard ACL can be used for several purpose. In this lab we will see how it can be used in controlling the unwanted network traffic. With standard ACL, we can define certain conditions for the network traffic passing through the router. By default router does not filter any traffic unless we manually put an ACL.

There are two types of ACLs:

1. **Standard ACL**: permits or denies packets based on source and destination IP address and also based on IP protocol information. Valid ACL ID range is: 1 - 99 or a string. Standard ACL shall be created in three ways viz. i) Classic Numbered

ii) Modern Numbered iii) Modern Named This lab tutorial presents the creation of all three Standard ACLs.

1. **Extended ACL**: it permits or denies packets based on source and destination IP address and also based on IP protocol information. Valid Extended ACL ID range is: 100 - 199 or a string

Access lists of some protocols must be identified by a name, and access lists of other protocols must be identified by a number. Some protocols can be identified by either a name or a number. When a number is used to identify an access list, the number must be within the specific range of numbers that is valid for the protocol. Cisco Access Control Lists are the set of conditions grouped together by name or number. These conditions are used in filtering the traffic passing from router. Through these conditions we can filter the traffic; either when it enters in router or when it exits from router.

When creating an access list, we define criteria that are applied to each packet that is processed by the router; the router decides whether to forward or block each packet on the basis of whether or not the packet matches the criteria.

By default when a router receives a packet in interface, it takes following actions:

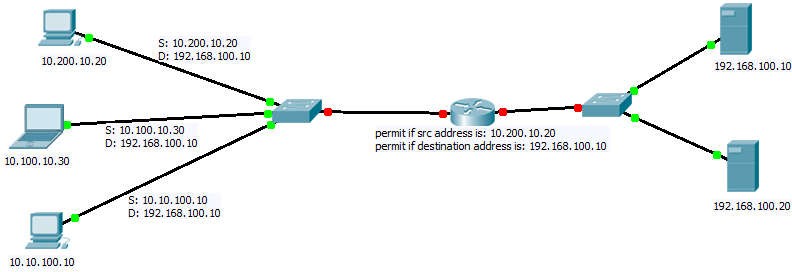
* + Grab destination address from the packet
  + Find an entry for destination address in routing table
  + If match found, forwards the packet from associate interface
  + If no match found, discard the packet immediately.

Suppose for the following network configuration, the router that only 10.0.0.10 has the right to access the 30.0.0.1. To match with this condition router will take following actions:-

* + Grab source and destination address from the packet
  + Match both addresses with given condition
  + If packet is not arrived from 10.200.10.20, drop the packet immediately.

1 <http://www.hp.com/rnd/support/manuals/pdf/release_06628_07110/Bk2_Ch3_ACL.pdf> <http://www.cisco.com/c/en/us/td/docs/ios/12_2/security/configuration/guide/fsecur_c/scfacls.pdf>

* + If packet is not destined to 192.168.100.10, drop the packet immediately.
  + If both condition match, find an entry for destination address in routing table
  + If match found, forwards the packet from associate interface
  + If no match found, discard the packet immediately.



**Tutorial Steps: Consider the case scenario as follows:**

IOE Pulchowk Campus network is divided into four cluster with its own routing domain and Autonomous system Number (say: 3245). We need to configure routers and network with the APNIC provided IP block as: 202.70.91.0/24. For each network section a separate ACL is required. The IP block distributions are as follows (Ref: <http://www.computernetworkingnotes.com/ccna-study-guide/configure-standard-access-control-list-step-by-step-> guide.html )

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Network  Name | Total  Hosts | Network Number | Subnet Mask | Wild Card  Mask | ACL  Number/Name |
| NCR | 12 | 202.70.91.0/28 | 255.255.255.240 | 0.0.0.15 | 10 |
| COMPLEX | 120 | 202.70.91.16/25 | 255.255.255.128 | 0.0.0.127 | COMPLEX |
| CIMISH | 59 | 202.70.91.144/27 | 255.255.255.192 | 0.0.0.63 | CIMISH |
| QUASD | 6 | 202.70.91.208/29 | 255.255.255.248 | 0.0.0.7 | 20 |

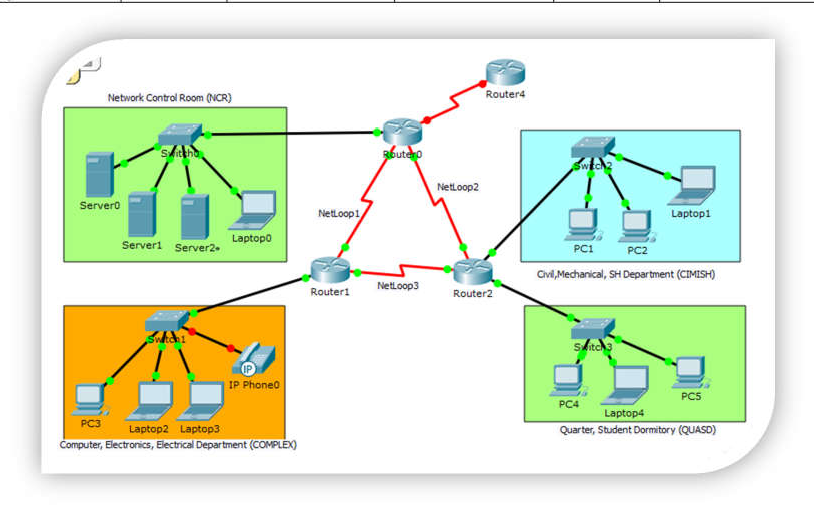


Figure: Network Diagram (Topology)

The ip address setting plan is as follows:

|  |  |  |
| --- | --- | --- |
| **Device Name** | **Interface Name** | **IP Address** |
| Router 0 | Serial 0/0/0 | 202.70.91.253/30 |
| Serial 0/0/1 | 202.70.91.249/30 |
| Gig 0/0 | 202.70.91.1/28 |
| Router 1 | Serial 0/0/0 | 202.70.91.254/30 |
| Serial 0/0/1 | 202.70.91.241/30 |
| Gig 0/0 | 202.70.91.17/25 |
| Router 2 | Serial 0/0/0 | 202.70.91.250/30 |
| Serial 0/0/1 | 202.70.91.242/30 |
| Gig 0/0 | 202.70.91.145/26 |
| Gig 0/1 | 202.70.91.209/28 |
| Server 0-3 | Server port | 202.70.91.2-4/28 |
| Laptop 0 | Client port | 202.70.91.5/28 |
| PC3 | Client port | 202.70.91.18/25 |
| Laptop 2-3 | Client port | 202.70.91.19-20/25 |
| PC1 | Client port | 202.70.91.146/26 |
| PC2 | Client port | 202.70.91.147/26 |
| Laptop-1 | Client port | 202.70.91.148/26 |
| PC4 | Client port | 202.70.91.210/29 |
| Laptop-4 | Client port | 202.70.91.211/29 |
| PC5 | Client port | 202.70.91.212/29 |

Based on the ip distribution and network topology sketched, 1) create network topology in your packet tracer tool 2) Assign essential IP addresses as shown in above table with given network. 3) Configure RIPv2 protocol in R1, R2 and R2 for IP routing. The ACL requirement shall be as follows.

1. COMPLEX network section should be able to access only NCR. It should not be able to access other networks (QUASD & CIMISH).

1. One user (PC1) from CIMISH network section should not be able to access anything except its own section.
2. One user (Laptop1) from CIMISH network section should be able to access all other networks but not COMPLEX network section. (at complex acl: deny 202.70.91.150 0.0.0.0)

It is assumed that you are able to configure each device ip address in the given network topology and also able to enable RIP practiced in previous labs that is the pre-requisite for this lab. You are now provided with tips about how to enable ACL in the network based on the above requirements.

|  |  |
| --- | --- |
| **On Router-0, create the following ACL**  *Enable. vbbbbb vb conf ter*  *ip access-list standard 10 deny 202.70.91.16 0.0.0.127*  *deny 202.70.91.208 0.0.0.7*  *permit any ctrl+Z*  *wr* | **On Router-1, create the following ACL**  *enable conf ter*  *ip access-list standard complex deny host 202.70.91.150 permit any*  *ctrl+z wr* |
| **On Router-0, create the following ACL** | After creating those ACLs, now test the connectivity as |
| *enable* | per the requirements (the ACL has not yet been applied) |
| *conf ter* |  |
| *ip access-list standard 20* | **Ping test from pc3 to server 0** |
| *deny 202.70.91.0 0.0.0.15* | PC>ping 202.70.91.2 |
| *deny 202.70.91.16 0.0.0.127* | Pinging 202.70.91.2 with 32 bytes of data: |
| *deny 202.70.91.128 0.0.0.63* | Reply from 202.70.91.2: bytes=32 time=10ms TTL=126 |
| *permit any* | Reply from 202.70.91.2: bytes=32 time=1ms TTL=126 |
| *exit* | Reply from 202.70.91.2: bytes=32 time=1ms TTL=126 |
| *ip access-list standard cimish* | Reply from 202.70.91.2: bytes=32 time=1ms TTL=12 |
| *deny host 202.70.91.146* |  |
| *deny 202.70.91.0 0.0.0.127* |  |

|  |  |
| --- | --- |
| *permit any ctrl+z*  *wr* |  |
| **Ping test from pc1 to server0**  PC>ping 202.70.91.2  Pinging 202.70.91.2 with 32 bytes of data:  Reply from 202.70.91.2: bytes=32 time=2ms TTL=126 Reply from 202.70.91.2: bytes=32 time=1ms TTL=126 Reply from 202.70.91.2: bytes=32 time=1ms TTL=126 Reply from 202.70.91.2: bytes=32 time=1ms TTL=126 | **Ping test from pc4 to server0**  PC>ping 202.70.91.2  Pinging 202.70.91.2 with 32 bytes of data:  Reply from 202.70.91.2: bytes=32 time=2ms TTL=126 Reply from 202.70.91.2: bytes=32 time=1ms TTL=126 Reply from 202.70.91.2: bytes=32 time=1ms TTL=126 Reply from 202.70.91.2: bytes=32 time=1ms TTL=126 |
| **Ping test from pc3 to pc4**  PC>ping 202.70.91.210  Pinging 202.70.91.210 with 32 bytes of data:  Reply from 202.70.91.210: bytes=32 time=1ms TTL=126 Reply from 202.70.91.210: bytes=32 time=1ms TTL=126 Reply from 202.70.91.210: bytes=32 time=1ms TTL=126 Reply from 202.70.91.210: bytes=32 time=1ms TTL=126 | **Ping test from pc1 to pc3**  PC>ping 202.70.91.18  Pinging 202.70.91.18 with 32 bytes of data:  Reply from 202.70.91.18: bytes=32 time=2ms TTL=126 Reply from 202.70.91.18: bytes=32 time=1ms TTL=126 Reply from 202.70.91.18: bytes=32 time=2ms TTL=126 Reply from 202.70.91.18: bytes=32 time=1ms TTL=126 |

Now apply the ACL in to the corresponding interfaces with inbound and outbound settings.

**On router 0, in configuration mode, issue the following acl**

int gig 0/0

ip access-group 10 out

**On router-1, in configuration mode, issue the following acl**

int gig 0/0

ip access-group complex out

**On router-2, in configuration mode, issue the following acl**

int gig 0/0

ip access-group cimish in exit

int gig 0/1

ip access-group 20 out ctrl+z

wr