**Name & ID**: Syed Asghar Abbas Zaidi **Date:** 7th November 2023

| **EE-424L Data Communication & Networking**  **Fall 2023**  **Habib University**  **Dhanani School of Science & Engineering** |
| --- |



**LAB 11: Configuration of Standard, Extended Access Control List, NAT and PAT**

| **Lab #11 Marks distribution:**   |  |  | **LR2=30** | **LR4=30** | **LR5=30** | **AR4=10** | | | --- | --- | --- | --- | --- | --- | --- | | **In-Lab Tasks** | **Task 1** | 10 | 10 | 10 | 10 | | | **Task 2** | 10 | 10 | 10 | | **Task 3** | 10 | 10 | 10 | | **Total Marks** | **100** | | | | |   **Lab #11 Marks Obtained:**   |  |  | **LR2=30** | **LR4=30** | **LR5=30** | **AR4=10** | | | --- | --- | --- | --- | --- | --- | --- | | **In-Lab Tasks** | **Task 1** |  |  |  |  | | | **Task 2** |  |  |  | | **Task 3** |  |  |  | | **Marks Obt.** |  | | | | | |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |

| o**bjectives** | **The objective of this lab is to configure and verify Standard, Extended Access Control List (ACL) , Network Address Translation (NAT) and Port Address Translation (PAT).** |
| --- | --- |

**Introduction**

Access Control List (ACL) is a security feature that allows you to filter the network traffic based on configured statements. An ACL can be used to filter either inbound or outbound traffic on an interface. Once you applied an access list on a router, the router examines every packet moving from interface to another interface in the specified direction and takes the appropriate action.

**Types of ACL**

An ACL can be either of the following two types.

1. **Standard access lists**

A Standard access list can use only the source IP address in an IP packet to filter the network traffic. Standard access lists are typically used permit or deny an entire system or network. They cannot be used to filter individual protocol or services such as FTP and Telnet.

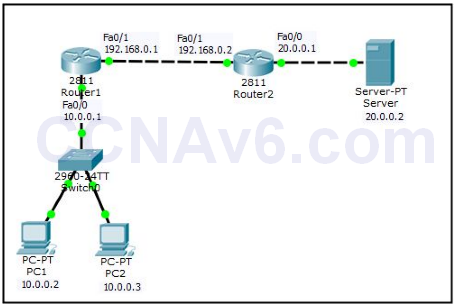
1. **Extended access lists**

Extended access lists use the source and destination IP addresses. They can be used to filter specific protocol or service. An ACL can be configured using either a number or a name. If you decide to use a name to configure an ACL it is referred as Named ACL.

**Network Address Translation (NAT)** and **Port Address Translation (PAT)** are the two protocols via which we can map the unregistered private (inside local address of an internal network to a registered public (inside global) address of an external network before moving the packet. The primary distinction is that **NAT** is used to map public IP addresses to private IP addresses in a one-to-one or many-to-one relationships. On the other hand, **PAT** is a sort of **NAT** in which numerous private IP addresses (many-to-one) are mapped into a single public IP address via ports.

| **Task 1: Configuration of Standard ACL** | **[15]** | |
| --- | --- | --- |
|  | |  |

Configure and create the below topology in packet tracer and complete the IP configuration and interface configuration in Table 1 according to your Network Topology. Attach your network topology with labelled IPs, subnet mask and interfaces as shown in below figure.



| **Device Name** | **Interface** | **IPs Assigned to devices** |
| --- | --- | --- |
| PC1 | Fa0/1 (of switch0) | 10.0.0.2 |
| PC2 | Fa0/2 (of switch0) | 10.0.0.3 |
| Router1 | Fa0/0  Fa0/1 | 10.0.0.1  192.168.0.1 |
| Router2 | Fa0/0  Fa0/1 | 192.168.0.2  20.0.0.1 |
| Server | Fa0/1 (of router2) | 20.0.0.2 |

Once you have created the above topology, configure the appropriate IP addresses as mentioned in the topology. To do so, execute the following commands on Router1.

Router1( config)# int fa0/ 0

Router1( config-if)# ip add 10.0.0.1 255.0.0.0

Router1( config-if)# no shut

Router1( config-if)# exit

Router1( config)# int fa0/ 1

Router1( config-if)# ip add 192.168.0.1 255.255.255.0

Router1( config-if)# no shut

Router1( config-if)# exit

Once you have configured appropriate IP addresses, use a routing method such as RIP. Write the commands below for RIP on Router 1.

| **Configuration of Router1:**    **Routing :** |
| --- |

Next, move on to Router2 and configure IP addresses and the RIP routing protocol. Attach the screenshots below.

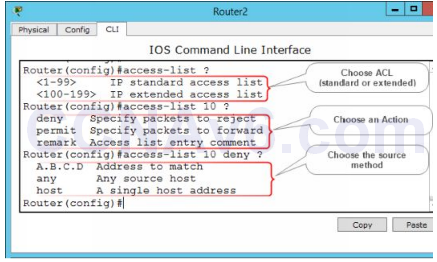
After configuring IP addresses on routers, configure IP addresses on PC1, PC2, and Server.

Now, open the Command Prompt on PC1 and type ping 20.0.0.2 what is the response?

| **Configuration and routing of Router 2:**  **PC1 should be able to ping Server, as the screenshot for the pinging wasn’t required, it is not attached.** |
| --- |

Now, you have configured the appropriate IP addresses and routing on your network topology.

Before configuring an ACL, we would like to explain the command syntaxes used to configure it. The following figure shows the command syntax used to configure an ACL.



**Configure Standard ACL**

In this task, we will restrict host PC 1(10.0.0.2) from accessing Server (20.0.0.2). To do so, perform the following steps:

For our Standard Access-List, we can use the **ACL Number** 1 to 99. These numbers can be **100 to 199**, if you use extended ACLs.

1. First, execute the following command on **Router 2** to deny host 10.0.0.2.

Router2(config)# access-list 10 deny host 10.0.0.2

2. Once you deny a host on a router, the router will deny all the hosts until you explicitly define the permitted hosts. In the following command we will permit all the hosts.

Router2(config)# access-list 10 permit any

3. Next, switch to the interface on which you want to apply the ACL, in this case Fa0/ 1, and define the direction (inbound or outbound) of traffic that you want to filter. In this case, we will filter the incoming packets towards Router2. To do so execute the following commands.

Router2(config)# int fa0/1

Router2(config-if)# ip access-group 10 in

Router2(config-if)# exit

4.  Once you applied an ACL on a router, execute the following command to view the applied ACLs.

Run **show ip access-lists** on Router 2 and write your observations below.

| **Configuration and then running “show ip access-lists” on Router 2:**    **I accidentally did it outwards instead of inwards, so i do further configuration to remove that and then do it correctly of it being inwards.**    **Observations:**  First we deny 10.0.0.2 host to route, however, it needs a comparison so that it can know what it can allow and what it can’t. Once we “permit any”, the router is successfully able to know that it needs to allow every other host except “10.0.0.2”.  But deny 10.0.0.2 where!?  This is where we access the port/interface, and then tell that, if 10.0.0.2 tries to “enter-in” through that port, you deny the host the “access”. |
| --- |

5.  Next, open the Command Prompt of PC1 and PC2 and try to ping server, what is the response?

| **Pinging from PC1 and PC2 to Server:**    **Response:**  As can be seen, it is denying PC1 (10.0.0.2) and that’s why we are getting “destination route unreachable”. |
| --- |

6.  Now, you have tested your ACL configuration. Now, remove the ACL configuration. To remove the configured ACL, execute the following command on Router2.

Router2( config)# no access-list 10 deny host 10.0.0.2

| **Configuration of Router 2:** |
| --- |

7.  Try to ping again from PC0 to Router2, this time you should be able ping successfully, because you have removed the applied ACL.

| **Proof that PC1 is now pinging:**    **As can be seen, it wasn’t previously able to reach it, but after successfully removal of the ASL, it is able to ping the server successfully** |
| --- |

| **Task 2: Configuration of Extended ACL** | **[15]** |
| --- | --- |

Extended ACL is more precise than standard ACL. Even we can block a particular IP or range of IP address or network address using extended ACL. We can also allow certain hosts and block few as per our requirement using extended ACL. Here in this task we will learn to configure and use Extended access-list. **Extended ACLs** are a little complex if we compare it with **Standard ACLs**. With **Extended ACLs**, we can restrict or allow specific things like **destination, protocol** or **port**.

**Configure ACL list on Router 2:**

First, we’ll create a statement that will permit the PC1 workstation access to Server:

R2(config)#access-list 100 permit ip 10.0.0.2 0.0.0.0 20.0.0.2 0.0.0.0

Next, we need to create a statement that will deny the PC2 workstation access to Server:

R2(config)#access-list 100 deny ip 10.0.0.3 0.0.0.0 20.0.0.2 0.0.0.0

Lastly, we need to apply the access list to the **Fa0/0** interface on R2:

R2(config)#int f0/1  
R2(config-if)#ip access-group 100 in

**Ping PC1 and PC2 to Server and note down the response below.**

| **Configuration of my Router 2:**    **Pinging of PC1 and PC2 to server:**    **As expected, it is denying access of the host 10.0.0.3 to 20.0.0.2** |
| --- |

**Verify the ACL**

Router# show access-lists and write down the response below:

| **Router 2:** |
| --- |

| **Task 3: Static NAT and PAT Configuration** | **[20]** |
| --- | --- |

Use the above network topology to configure the commands below.

First delete R1 and R2 and attach new R1 and R2 to avoid any issues encountered previously.

Assign IPs to R1 and R2 interfaces.

**Configure PAT (NAT Overload) on R1:**

R1>enable

R1#configure terminal

Enter configuration commands, one per line. End with CNTL/Z.

R1(config)#access-list 1 permit 10.0.0.2 0.0.0.0

R1(config)#access-list 1 permit 10.0.0.3 0.0.0.0

R1(config)#access-list 1 deny any

R1(config)#ip nat pool lab 50.0.0.1 50.0.0.1 netmask 255.0.0.0

R1(config)#ip nat inside source list 1 pool lab overload

R1(config)#interface FastEthernet 0/0

R1(config-if)#ip nat inside

R1(config-if)#exit

R1(config)#interface FastEthernet 0/1

R1(config-if)#ip nat outside

R1(config-if)#exit

**Configure static NAT on R2:**

R2>enable

R2#configure terminal

Enter configuration commands, one per line. End with CNTL/Z.

R2(config)#ip nat inside source static 20.0.0.2 30.0.0.2

R2(config)#interface FastEthernet 0/1

R2(config-if)#ip nat outside

R2(config-if)#exit

R2(config)#interface FastEthernet 0/0

R2(config-if)#ip nat inside

R2(config-if)#exit

**Configure static routing in R1**

R1(config)#ip route 30.0.0.0 255.0.0.0 192.168.0.2

**Configure static routing in R2**

R2(config)#ip route 50.0.0.0 255.0.0.0 192.168.0.1

| **Configuration of Router 1:**      **Configuration of Router 2:**    **Static Routing of Router 1:**    **Static Routing of Router 2:** |
| --- |

In this lab we configured PAT on R1 for 10.0.0.2 and 10.0.0.3 and static NAT on R2 for 20.0.0.2

| **Device** | **Inside Local IP address** | **Inside Global IP address** |
| --- | --- | --- |
| **PC1** | 10.0.0.2 | 50.0.0.1 |
| **PC2** | 10.0.0.3 | 50.0.0.1 |
| **Server** | 20.0.0.2 | 30.0.0.2 |

To test this setup click **PC1** and **Desktop** and click **Command Prompt**.

* Run **ipconfig** command.
* Run **ping 30.0.0.2** command.
* Run **ping 20.0.0.2** command.

| **Result:** |
| --- |

**Discuss above results.**

| We are doing PAT on router 1, while we are doing NAT on router 2.  NAT (Network Address Translation) is a one-to-one mapping, and when we map 20.0.0.2 as 30.0.0.2, we, from PC1 fail to ping 20.0.0.2 cause it exists in the form of 30.0.0.2 now. This can be verified when I ping 20.0.0.1 which it succeeds in doing.    This is cause, I never configured and translated 20.0.0.1 to be any other address like 20.0.0.2 |
| --- |

**You can verify these by running these two commands on R2. Discuss the results and attach the screenshot below.**

Router#show ip nat translations

Router#show ip nat statistics

| **Router 2:** |
| --- |

Network Topology:

|  |
| --- |

**Lab Evaluation Assessment Rubric**

**EE-424 Lab 11**

| **#** | **Assessment Elements** | **Level 1: Unsatisfactory**  **Points 0-1** | **Level 2: Developing**  **Points 2** | **Level 3: Good**  **Points 3** | **Level 4: Exemplary**  **Points 4** |
| --- | --- | --- | --- | --- | --- |
| **LR2** | **Program/Code/ Simulation Model/ Network Model** | Program/code/simulation model/network model does not implement the required functionality and has several errors. The student is not able to utilize even the basic tools of the software. | Program/code/simulation model/network model has some errors and does not produce completely accurate results. Student has limited command on the basic tools of the software. | Program/code/simulation model/network model gives correct output but not efficiently implemented or implemented by computationally complex routine. | Program/code/simulation /network model is efficiently implemented and gives correct output. Student has full command on the basic tools of the software. |
| **LR4** | **Data Collection** | Measurements are incomplete, inaccurate and imprecise. Observations are incomplete or not included. Symbols, units and significant figures are not included. | Measurements are somewhat inaccurate and imprecise. Observations are incomplete or vague. Major errors are there in using symbols, units and significant digits. | Measurements are mostly accurate. Observations are generally complete. Minor errors are present in using symbols, units and significant digits. | Measurements are both accurate and precise. Data collection is systematic. Observations are very thorough and include appropriate symbols, units and significant digits and task completed in due time. |
| **LR5** | **Results & Plots** | Figures/ graphs / tables are not developed or are poorly constructed with erroneous results. Titles, captions, units are not mentioned. Data is presented in an obscure manner. | Figures, graphs and tables are drawn but contain errors. Titles, captions, units are not accurate. Data presentation is not too clear. | All figures, graphs, tables are correctly drawn but contain minor errors or some of the details are missing. | Figures / graphs / tables are correctly drawn and appropriate titles/captions and proper units are mentioned. Data presentation is systematic. |
| **AR4** | **\*Report Submission** | Late submission after 1 week and in between 2 weeks. | Late submission after 2 days and within a week. | Late submission after the lab timing and within 2 days of the due date. | Timely submission of the report and in the lab time. |

**\*Report:** Report will not be accepted after due date