

Roll Number:- 2105048

Name:-Manav Jain

Date:- 13/09/2023

Lab Assignment No:-10

Aim:- To study and configure firewalls using IP tables

Lab Outcome Attained :- LO6

Theory:-

1. What is firewall ?

A firewall is a security system designed to prevent unauthorized access to or from a private network. It can be implemented as hardware, software, or a combination of both. Typically, a firewall has two network interfaces: one facing the external network and one facing the internal network. Its primary function is to regulate and control the traffic that can pass from one side to the other.

At its most fundamental level, firewalls can block network traffic intended for specific IP addresses or server ports. In a TCP network, data is transmitted in packets, each comprising a packet header that contains essential control information, including source and destination addresses and packet sequence details, along with the actual data payload. This control information within each packet not only ensures the proper delivery of associated data but also provides firewalls with various criteria for matching packets against predefined firewall rules.

2. List different types of firewall .

Three basic types of network firewalls: packet filtering (stateless), stateful, and application layer.

Packet filtering, or stateless, firewalls work by inspecting individual packets in isolation. As such, they are unaware of connection state and can only allow or deny packets based on individual packet headers.

Stateful firewalls are able to determine the connection state of packets, which makes them much more flexible than stateless firewalls. They work by collecting related packets until the connection state can be determined before any firewall rules are applied to the traffic.

Application firewalls go one step further by analyzing the data being transmitted, which allows network traffic to be matched against firewall rules that are specific to individual services or applications. These are also known as proxy-based firewalls.

3. Write different options that can be used in configuring firewall.

Configuring a firewall using IPtables involves a series of operations to define rules and policies that control network traffic. IPtables is a popular firewall management tool on Linux systems. Different options involved in configuring a firewall using IPtables include:

Listing Existing Rules:

You can start by listing existing rules to understand the current configuration. Use the iptables -L or iptables -S command to view the existing rules in place.

Setting Default Policies:

Define the default policies for incoming and outgoing traffic. This sets the default action (ACCEPT, DROP, or REJECT) for packets that don't match any specific rules.

Creating Rules:

You can create rules to allow or block traffic based on various criteria such as source IP, destination IP, ports, and protocols. For example, you can use the iptables -A command to append a rule.

Specifying Chains:

Iptables uses chains (e.g., INPUT, OUTPUT, FORWARD) to organize rules. You can specify which chain a rule applies to when creating it.

Defining Match Criteria:

Define match criteria for each rule. You can use parameters like -s for source IP, -d for destination IP, -p for protocol, and -m for matching modules.

Defining Target Actions:

Specify the target action for matched packets. Common targets include ACCEPT, DROP (silently discard), REJECT (discard with an error message), and LOG (log the packet).

Modifying Rules:

You can modify existing rules using commands like iptables -R (replace), iptables -D (delete), or iptables -I (insert) to make changes to specific rules.

Saving and Restoring Rules:

After configuring rules, use the iptables-save command to save the rules to a file. You can later restore the rules using the iptables-restore command.

Network Address Translation (NAT):

If you need to set up port forwarding, masquerading, or other NAT-related tasks, you can configure NAT rules using Iptables.

Packet Filtering for Specific Applications:

You can configure rules that are specific to certain applications or services, such as web servers (HTTP and HTTPS), DNS, or SSH.

4. Write the commands used for configuring firewall using IPTABLES.

a) List Existing Rules:

`iptables -L` or `iptables --list`: List current rules. `iptables -S` or `iptables --list-rules`: Show rules in a more detailed format.

Set Default Policies:

`iptables -P [chain] [target]`: Set the default policy for a chain (e.g., INPUT, OUTPUT, FORWARD).

Create Rules:

`iptables -A [chain] [options]`: Append a rule to a chain.

`iptables -I [chain] [rule number] [options]`: Insert a rule at a specific position. `iptables`

`-N [chain]`: Create a new user-defined chain.

Specify Chains:

Use the `-A`, `-I`, or `-N` options with the desired chain name.

Define Match Criteria:

`-s [source]`: Match packets from a specific source.

`-d [destination]`: Match packets going to a specific destination.

`-p [protocol]`: Match packets of a specific protocol.

`-m [module] [options]`: Use matching modules for more complex criteria.

Define Target Actions:

`-j [target]`: Specify the target action (e.g., ACCEPT, DROP, REJECT, LOG) for matched packets.

Modify Rules:

iptables -R [chain] [rule number] [options]: Replace a rule.

iptables -D [chain] [options]: Delete a rule. iptables -I

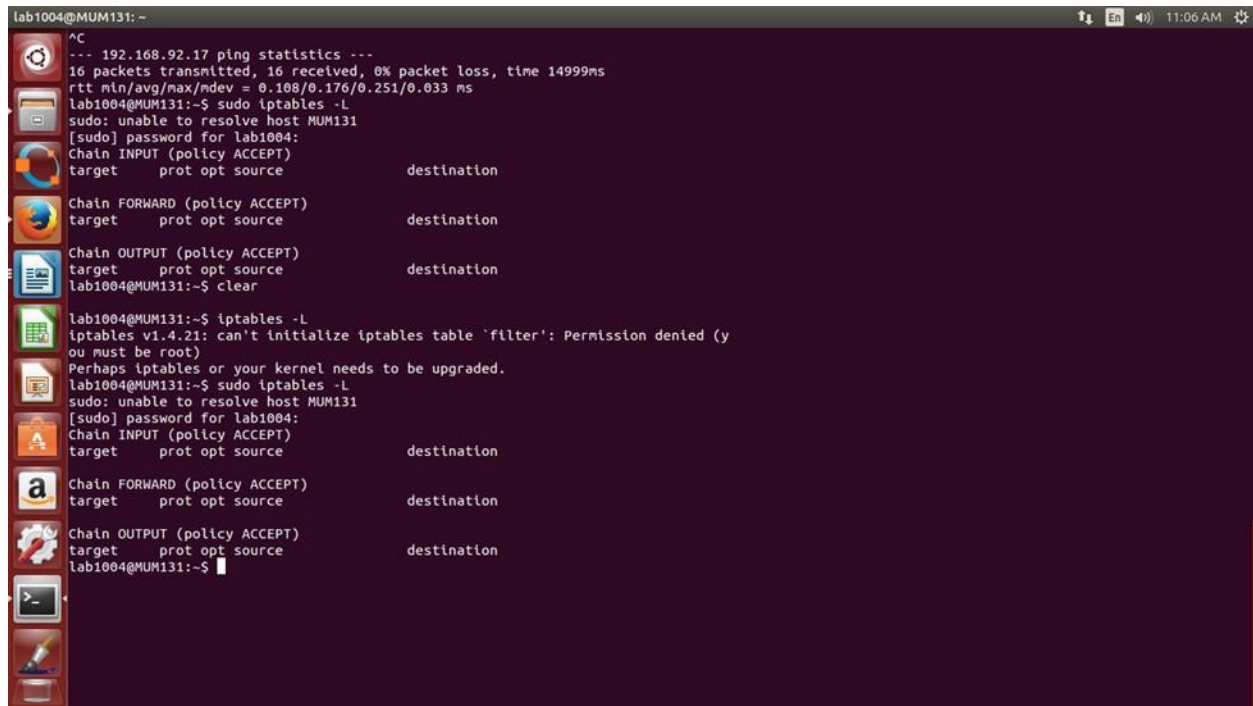
[chain] [rule number] [options]: Insert a rule.

Save and Restore Rules:

iptables-save: Save rules to a file (typically redirected to a file).

iptables-restore: Restore rules from a file.

Output Screenshots:-



```
lab1004@MUM131: ~  
^C  
--- 192.168.92.17 ping statistics ---  
16 packets transmitted, 16 received, 0% packet loss, time 14999ms  
rtt min/avg/max/mdev = 0.108/0.176/0.251/0.033 ms  
lab1004@MUM131:~$ sudo iptables -L  
sudo: unable to resolve host MUM131  
[sudo] password for lab1004:  
Chain INPUT (policy ACCEPT)  
target    prot opt source                destination  
  
Chain FORWARD (policy ACCEPT)  
target    prot opt source                destination  
  
Chain OUTPUT (policy ACCEPT)  
target    prot opt source                destination  
lab1004@MUM131:~$ clear  
  
lab1004@MUM131:~$ iptables -L  
iptables v1.4.21: can't initialize iptables table 'filter': Permission denied (you must be root)  
Perhaps iptables or your kernel needs to be upgraded.  
lab1004@MUM131:~$ sudo iptables -L  
sudo: unable to resolve host MUM131  
[sudo] password for lab1004:  
Chain INPUT (policy ACCEPT)  
target    prot opt source                destination  
  
Chain FORWARD (policy ACCEPT)  
target    prot opt source                destination  
  
Chain OUTPUT (policy ACCEPT)  
target    prot opt source                destination  
lab1004@MUM131:~$
```

```
lab1004@MUM131:~$ sudo iptables -L
[sudo] password for lab1004:
Chain INPUT (policy ACCEPT)
target prot opt source destination

Chain FORWARD (policy ACCEPT)
target prot opt source destination

Chain OUTPUT (policy ACCEPT)
target prot opt source destination
lab1004@MUM131:~$ clear

lab1004@MUM131:~$ sudo iptables -L
iptables v1.4.21: can't initialize iptables table 'filter': Permission denied (you must be root)
Perhaps iptables or your kernel needs to be upgraded.
lab1004@MUM131:~$ sudo iptables -L
[sudo] password for lab1004:
Chain INPUT (policy ACCEPT)
target prot opt source destination

Chain FORWARD (policy ACCEPT)
target prot opt source destination

Chain OUTPUT (policy ACCEPT)
target prot opt source destination
lab1004@MUM131:~$ sudo iptables -A INPUT -p tcp --dport ssh -j ACCEPT
[sudo] password for lab1004:
lab1004@MUM131:~$ sudo iptables -L
Chain INPUT (policy ACCEPT)
target prot opt source destination
ACCEPT tcp -- anywhere anywhere tcp dpt:ssh

Chain FORWARD (policy ACCEPT)
target prot opt source destination

Chain OUTPUT (policy ACCEPT)
target prot opt source destination
lab1004@MUM131:~$
```

```
lab1004@MUM131:~$ sudo iptables -L
[sudo] password for lab1004:
Chain INPUT (policy ACCEPT)
target prot opt source destination

Chain FORWARD (policy ACCEPT)
target prot opt source destination

Chain OUTPUT (policy ACCEPT)
target prot opt source destination
lab1004@MUM131:~$ sudo iptables -A INPUT -p tcp --dport ssh -j ACCEPT
[sudo] password for lab1004:
lab1004@MUM131:~$ sudo iptables -L
Chain INPUT (policy ACCEPT)
target prot opt source destination
ACCEPT tcp -- anywhere anywhere tcp dpt:ssh

Chain FORWARD (policy ACCEPT)
target prot opt source destination

Chain OUTPUT (policy ACCEPT)
target prot opt source destination
lab1004@MUM131:~$ sudo iptables -A INPUT -p tcp --dport 80 -j ACCEPT
[sudo] password for lab1004:
lab1004@MUM131:~$ sudo iptables -L
Chain INPUT (policy ACCEPT)
target prot opt source destination
ACCEPT tcp -- anywhere anywhere tcp dpt:ssh
ACCEPT tcp -- anywhere anywhere tcp dpt:http

Chain FORWARD (policy ACCEPT)
target prot opt source destination

Chain OUTPUT (policy ACCEPT)
target prot opt source destination
lab1004@MUM131:~$
```

```
lab1004@MUM131: ~  
target      prot opt source      destination  
  
Chain OUTPUT (policy ACCEPT)  
target      prot opt source      destination  
lab1004@MUM131:~$  
lab1004@MUM131:~$  
lab1004@MUM131:~$  
lab1004@MUM131:~$ sudo iptables -A INPUT -j DROP  
sudo: unable to resolve host MUM131  
lab1004@MUM131:~$ sudo iptables -L  
sudo: unable to resolve host MUM131  
Chain INPUT (policy ACCEPT)  
target      prot opt source      destination  
ACCEPT      tcp  --  anywhere    anywhere      tcp dpt:ssh  
ACCEPT      tcp  --  anywhere    anywhere      tcp dpt:http  
DROP        all  --  anywhere    anywhere  
DROP        all  --  anywhere    anywhere  
  
Chain FORWARD (policy ACCEPT)  
target      prot opt source      destination  
  
Chain OUTPUT (policy ACCEPT)  
target      prot opt source      destination  
lab1004@MUM131:~$
```

```
lab1004@MUM131: ~  
target      prot opt source      destination  
  
Chain OUTPUT (policy ACCEPT)  
target      prot opt source      destination  
lab1004@MUM131:~$  
lab1004@MUM131:~$  
lab1004@MUM131:~$  
lab1004@MUM131:~$ sudo iptables -A INPUT -j DROP  
sudo: unable to resolve host MUM131  
lab1004@MUM131:~$ sudo iptables -L  
sudo: unable to resolve host MUM131  
Chain INPUT (policy ACCEPT)  
target      prot opt source      destination  
ACCEPT      tcp  --  anywhere    anywhere      tcp dpt:ssh  
ACCEPT      tcp  --  anywhere    anywhere      tcp dpt:http  
DROP        all  --  anywhere    anywhere  
DROP        all  --  anywhere    anywhere  
  
Chain FORWARD (policy ACCEPT)  
target      prot opt source      destination  
  
Chain OUTPUT (policy ACCEPT)  
target      prot opt source      destination  
lab1004@MUM131:~$ sudo iptables -I INPUT 1 -i lo -j ACCEPT  
sudo: unable to resolve host MUM131  
lab1004@MUM131:~$ sudo iptables -L  
sudo: unable to resolve host MUM131  
Chain INPUT (policy ACCEPT)  
target      prot opt source      destination  
ACCEPT      all  --  anywhere    anywhere  
ACCEPT      tcp  --  anywhere    anywhere      tcp dpt:ssh  
ACCEPT      tcp  --  anywhere    anywhere      tcp dpt:http  
DROP        all  --  anywhere    anywhere  
DROP        all  --  anywhere    anywhere  
  
Chain FORWARD (policy ACCEPT)  
target      prot opt source      destination  
  
Chain OUTPUT (policy ACCEPT)  
target      prot opt source      destination  
lab1004@MUM131:~$
```

Conclusion:-

Learnt about firewall , types of firewall , configuration of firewall using rules and also executed basic options for firewall like allowing incoming traffic on specific port , blocking traffic , editing IP tables.