NAME : ADITYA R SAWANT
CLASS : TE-4-D
ROLL NO. : 46
SUBJECT : COMPUTER NETWORK

Experiment 6

AIM:

a. Set up multipleIP addresses on a singleLAN.
b. Using Netstat And route commands of Linux, do the following:
   - View current routing table
   - Add and delete routes
   - Change default gateway
c. Perform packet filtering by enabling IP forwarding using IPtables in Linux.

Theory:

First, let us find the IP address of the network card. In my Ubuntu 15.10 server, I use only one network card.

Run the following command to find out the IP address:

Sudo ipaddr
Sample output:

1: lo: <LOOPBACK,UP,LOWER_UP>mtu 65536 qdiscnoqueue state UNKNOWN group default

link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00

inet 127.0.0.1/8 scope host lo

valid_lft forever preferred_lft forever

inet6 ::1/128 scope host

valid_lft forever preferred_lft forever

2: **enp0s3**: <BROADCAST,MULTICAST,UP,LOWER_UP>mtu 1500 qdiscpfifo_fast state UP group default qlen 1000

link/ether 08:00:27:2a:03:4b brdff:ff:ff:ff:ff:ff

inet**192.168.1.103/24**brd 192.168.1.255 scope global enp0s3

valid_lft forever preferred_lft forever

inet6 fe80::a00:27ff:fe2a:34e/64 scope link
valid_lft forever preferred_lft forever
Or

sudo ifconfig
Sample output:

enp0s3 Link encap:EthernetHWaddr 08:00:27:2a:03:4b

inet addr:**192.168.1.103** Bcast:192.168.1.255 Mask:255.255.255.0

inet6 addr: fe80::a00:27ff:fe2a:34e/64 Scope:Link

UP BROADCAST RUNNING MULTICAST MTU:1500 Metric:1

RX packets:186 errors:0 dropped:0 overruns:0 frame:0

TX packets:70 errors:0 dropped:0 overruns:0 carrier:0

collisions:0 txqueuelen:1000

RX bytes:21872 (21.8 KB) TX bytes:9666 (9.6 KB)

lo Link encap:Local Loopback

inet addr:127.0.0.1 Mask:255.0.0.0

inet6 addr: ::1/128 Scope:Host

UP LOOPBACK RUNNING MTU:65536 Metric:1

RX packets:217 errors:0 dropped:0 overruns:0 frame:0

TX packets:217 errors:0 dropped:0 overruns:0 carrier:0

collisions:0 txqueuelen:0

RX bytes:38793 (38.7 KB) TX bytes:38793 (38.7 KB)

As you see in the above output, my network card name is **enp0s3**, and its IP address is **192.168.1.103**.
Now let us add an additional IP address, for example **192.168.1.104**, to the Interface card.

Open your Terminal and run the following command to add additional IP. sudo ipaddr

add 192.168.1.104/24 dev enp0s3

or

#sudo ifconfig eth0:0 192.168.1.104 up

Now, let us check if the IP is added using command:

Sudo ipaddress show enp0s3
Sample output:

2: enp0s3: <BROADCAST,MULTICAST,UP,LOWER_UP>mtu 1500 qdiscpfifo_fast state UP group default qlen 1000

link/ether 08:00:27:2a:03:4e brdff:ff:ff:ff:ff:ff

inet 192.168.1.103/24 brd 192.168.1.255 scope global enp0s3

valid_lft forever preferred_lft forever

inet**192.168.1.104/24** scope global secondary enp0s3

valid_lft forever preferred_lft forever

inet6 fe80::a00:27ff:fe2a:34e/64 scope link

valid_lft forever preferred_lft forever

Similarly, you can add as many IP addresses as you want.

Let us ping the IP address to verify it.

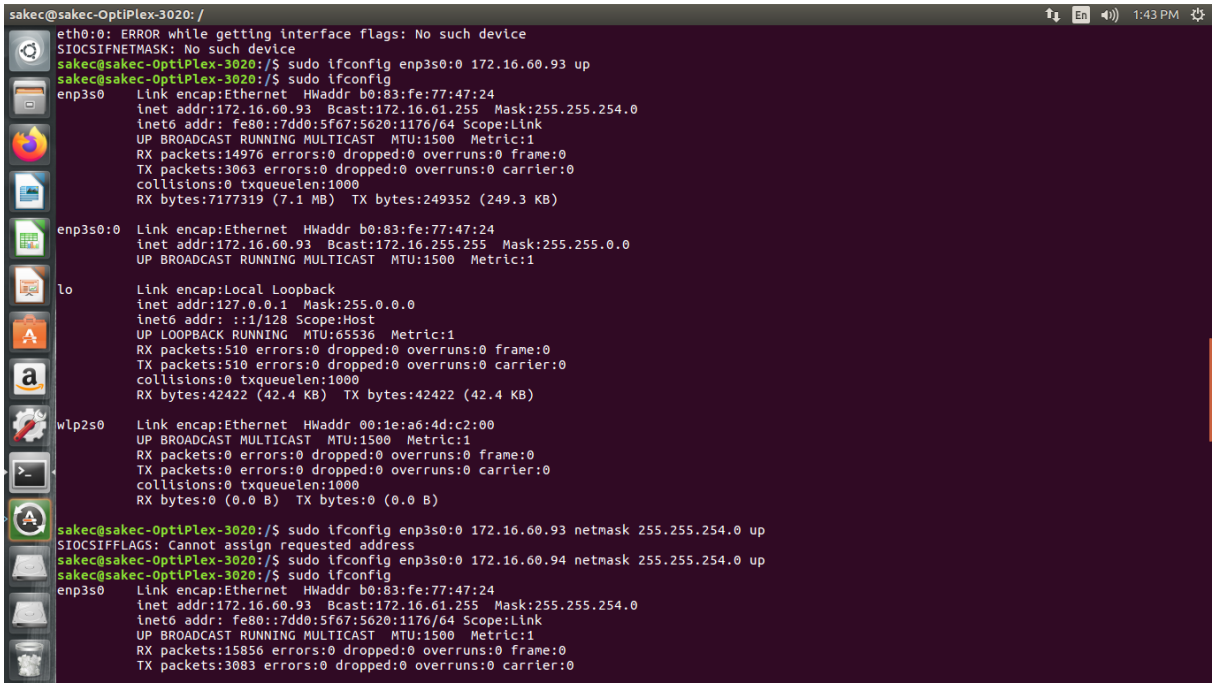
```
sudo ping 192.168.1.104
```

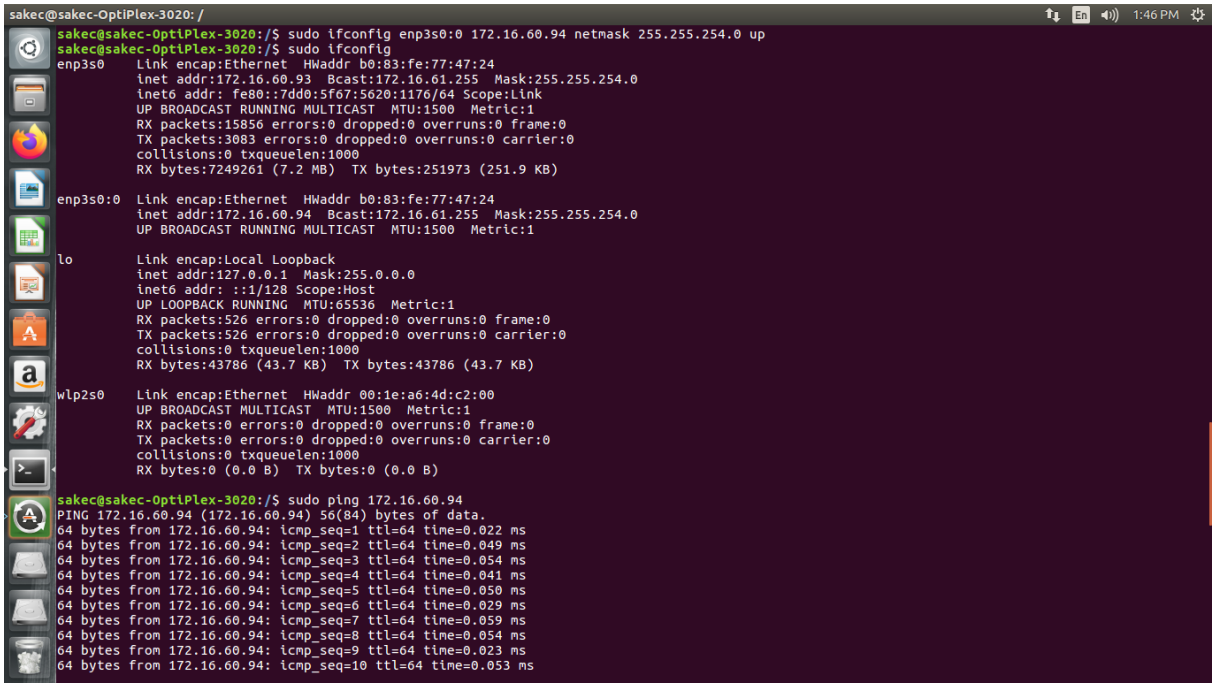
Sample output:

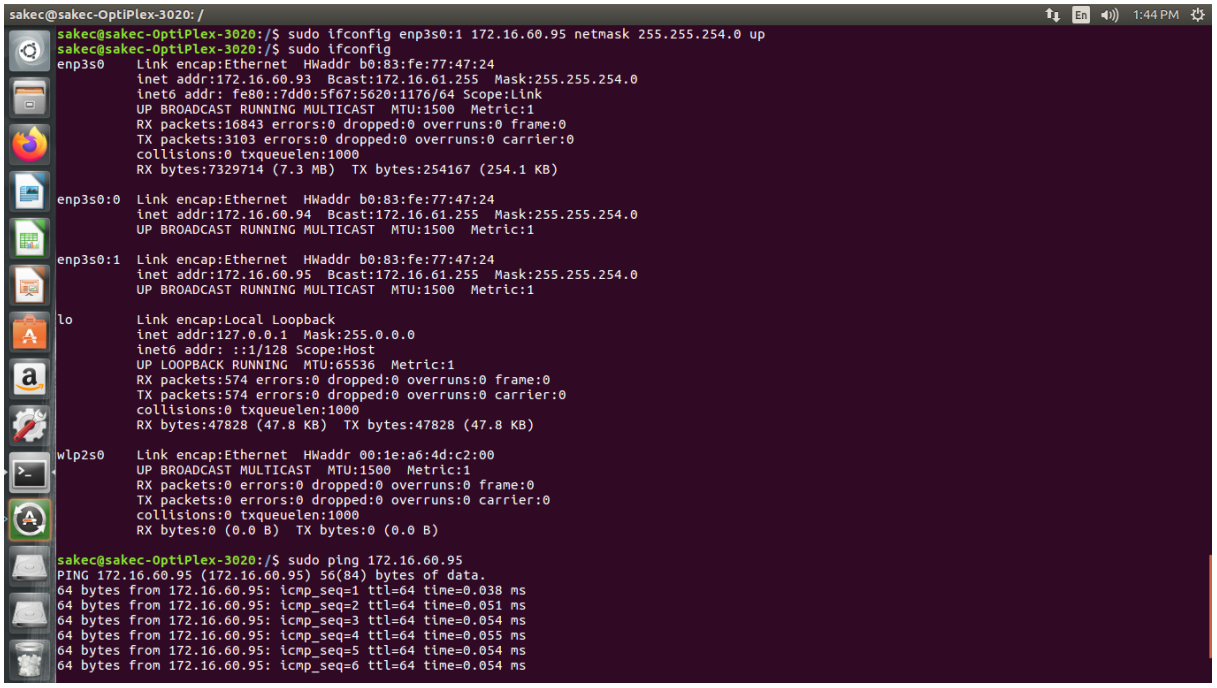
```
PING 192.168.1.104 (192.168.1.104) 56(84) bytes of data.

64 bytes from 192.168.1.104: icmp_seq=1 ttl=64 time=0.901 ms

64 bytes from 192.168.1.104: icmp_seq=2 ttl=64 time=0.571 ms

64 bytes from 192.168.1.104: icmp_seq=3 ttl=64 time=0.521 ms

64 bytes from 192.168.1.104: icmp_seq=4 ttl=64 time=0.524 ms
```

The advantage of using this IP aliasing is, you don't need to have a physical adapter attached to each IP, but instead you can create multiple or many virtual interfaces (aliases) to a single physical card.

Output :


sakec@sakec-OptiPlex-3020: /
1:43 PM
eth0:0: ERROR while getting interface flags: No such device
SIOCSIFNETMASK: No such device
sakec@sakec-OptiPlex-3020:/$ sudo ifconfig enp3s0:0 172.16.60.93 up
sakec@sakec-OptiPlex-3020:/$ sudo ifconfig
enp3s0 Link encap:Ethernet HWaddr b0:83:fe:77:47:24
inet addr:172.16.60.93 Bcast:172.16.61.255 Mask:255.255.254.0
inet6 addr: fe80::7dd0:5f67:5620:1176/64 Scope:Link
UP BROADCAST RUNNING MULTICAST MTU:1500 Metric:1
RX packets:14976 errors:0 dropped:0 overruns:0 frame:0
TX packets:3063 errors:0 dropped:0 overruns:0 carrier:0
collisions:0 txqueuelen:1000
RX bytes:7177319 (7.1 MB) TX bytes:249352 (249.3 KB)
enp3s0:0 Link encap:Ethernet HWaddr b0:83:fe:77:47:24
inet addr:172.16.60.93 Bcast:172.16.255.255 Mask:255.255.0.0
UP BROADCAST RUNNING MULTICAST MTU:1500 Metric:1
lo Link encap:Local Loopback
inet addr:127.0.0.1 Mask:255.0.0.0
inet6 addr: ::1/128 Scope:Host
UP LOOPBACK RUNNING MTU:65536 Metric:1
RX packets:510 errors:0 dropped:0 overruns:0 frame:0
TX packets:510 errors:0 dropped:0 overruns:0 carrier:0
collisions:0 txqueuelen:1000
RX bytes:42422 (42.4 KB) TX bytes:42422 (42.4 KB)
wlp2s0 Link encap:Ethernet HWaddr 00:1e:a6:4d:c2:00
UP BROADCAST MULTICAST MTU:1500 Metric:1
RX packets:0 errors:0 dropped:0 overruns:0 frame:0
TX packets:0 errors:0 dropped:0 overruns:0 carrier:0
collisions:0 txqueuelen:1000
RX bytes:0 (0.0 B) TX bytes:0 (0.0 B)
sakec@sakec-OptiPlex-3020:/$ sudo ifconfig enp3s0:0 172.16.60.93 netmask 255.255.254.0 up
SIOCSIFFLAGS: Cannot assign requested address
sakec@sakec-OptiPlex-3020:/$ sudo ifconfig enp3s0:0 172.16.60.94 netmask 255.255.254.0 up
sakec@sakec-OptiPlex-3020:/$ sudo ifconfig
enp3s0 Link encap:Ethernet HWaddr b0:83:fe:77:47:24
inet addr:172.16.60.93 Bcast:172.16.61.255 Mask:255.255.254.0
inet6 addr: fe80::7dd0:5f67:5620:1176/64 Scope:Link
UP BROADCAST RUNNING MULTICAST MTU:1500 Metric:1
RX packets:15856 errors:0 dropped:0 overruns:0 frame:0
TX packets:3083 errors:0 dropped:0 overruns:0 carrier:0


sakec@sakec-OptiPlex-3020: /
1:46 PM
sakec@sakec-OptiPlex-3020:/$ sudo ifconfig enp3s0:0 172.16.60.94 netmask 255.255.254.0 up
sakec@sakec-OptiPlex-3020:/$ sudo ifconfig
enp3s0 Link encap:Ethernet HWaddr b0:83:fe:77:47:24
inet addr:172.16.60.93 Bcast:172.16.61.255 Mask:255.255.254.0
inet6 addr: fe80::7dd0:5f67:5620:1176/64 Scope:Link
UP BROADCAST RUNNING MULTICAST MTU:1500 Metric:1
RX packets:15856 errors:0 dropped:0 overruns:0 frame:0
TX packets:3083 errors:0 dropped:0 overruns:0 carrier:0
collisions:0 txqueuelen:1000
RX bytes:7249261 (7.2 MB) TX bytes:251973 (251.9 KB)
enp3s0:0 Link encap:Ethernet HWaddr b0:83:fe:77:47:24
inet addr:172.16.60.94 Bcast:172.16.61.255 Mask:255.255.254.0
UP BROADCAST RUNNING MULTICAST MTU:1500 Metric:1
lo Link encap:Local Loopback
inet addr:127.0.0.1 Mask:255.0.0.0
inet6 addr: ::1/128 Scope:Host
UP LOOPBACK RUNNING MTU:65536 Metric:1
RX packets:526 errors:0 dropped:0 overruns:0 frame:0
TX packets:526 errors:0 dropped:0 overruns:0 carrier:0
collisions:0 txqueuelen:1000
RX bytes:43786 (43.7 KB) TX bytes:43786 (43.7 KB)
wlp2s0 Link encap:Ethernet HWaddr 00:1e:a6:4d:c2:00
UP BROADCAST MULTICAST MTU:1500 Metric:1
RX packets:0 errors:0 dropped:0 overruns:0 frame:0
TX packets:0 errors:0 dropped:0 overruns:0 carrier:0
collisions:0 txqueuelen:1000
RX bytes:0 (0.0 B) TX bytes:0 (0.0 B)
sakec@sakec-OptiPlex-3020:/$ sudo ping 172.16.60.94
PING 172.16.60.94 (172.16.60.94) 56(84) bytes of data.
64 bytes from 172.16.60.94: icmp_seq=1 ttl=64 time=0.022 ms
64 bytes from 172.16.60.94: icmp_seq=2 ttl=64 time=0.049 ms
64 bytes from 172.16.60.94: icmp_seq=3 ttl=64 time=0.054 ms
64 bytes from 172.16.60.94: icmp_seq=4 ttl=64 time=0.041 ms
64 bytes from 172.16.60.94: icmp_seq=5 ttl=64 time=0.050 ms
64 bytes from 172.16.60.94: icmp_seq=6 ttl=64 time=0.029 ms
64 bytes from 172.16.60.94: icmp_seq=7 ttl=64 time=0.059 ms
64 bytes from 172.16.60.94: icmp_seq=8 ttl=64 time=0.054 ms
64 bytes from 172.16.60.94: icmp_seq=9 ttl=64 time=0.023 ms
64 bytes from 172.16.60.94: icmp_seq=10 ttl=64 time=0.053 ms



To check the routing table

Command*: nestat -rn*

$ netstat -rn

Kernel IP routing table

Destination Gateway Genmask Flags MSS Window irttIface

0.0.0.0 192.168.0.1 0.0.0.0 UG 0 0 0 wlan0

192.168.0.0 0.0.0.0 255.255.255.0 U 0 0 0 wlan0

Adding route

sudo route add -net 192.168.3.0 gw 192.168.1.1 netmask 255.255.255.0 dev eth0

Deleting route

sudo route del -net 192.168.3.0 gw 192.168.1.1 netmask 255.255.255.0 dev eth0

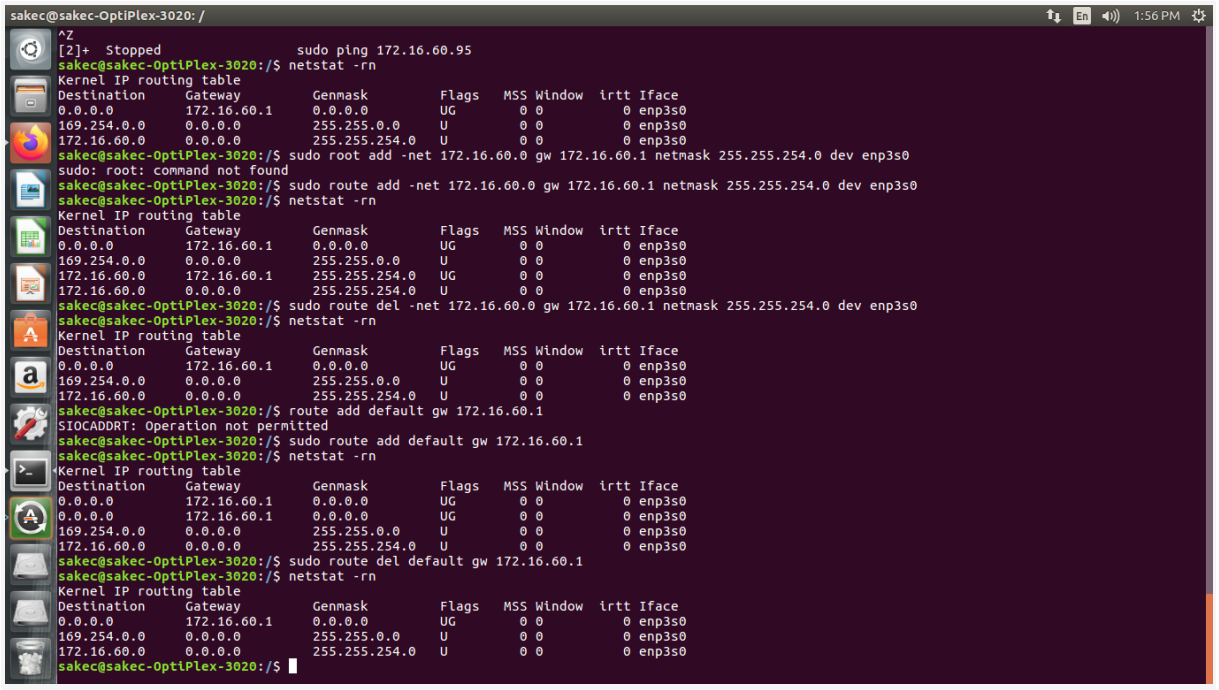
A quick way to add default route

route add default gw 192.168.1.1

A quick way to delete defualt route

route del default gw 192.168.1.1

Output :



Use of iptables in linux to create firewalls

iptables is a command-line firewall utility that uses policy chains to allow or block traffic.

When a connection tries to establish itself on your system, iptables looks for a rule in its listto match it to. If it doesn‘t find one, it resorts to the default action.

To install iptables:

sudo apt-get install iptables

Packet Filtering

The Linux kernel uses the Net filter facility to filter packets, allowing some of them to bereceived by or pass through the system while stopping others. This facility is built in to theLinux kernel, and has five built-in tables or rules lists, as follows:

- filter — The default table for handling network packets.

- NAT — Used to alter packets that create a new connection and used for Network Address Translation (NAT).
- mangle — Used for specific types of packet alteration.

Each table has a group of built-in chains, which correspond to the actions performed on the packet by netfilter.

The built-in chains for the filter table are as follows:

o **INPUT** — Applies to network packets that are targeted for the

host. o **OUTPUT** — Applies to locally-generated network packets.

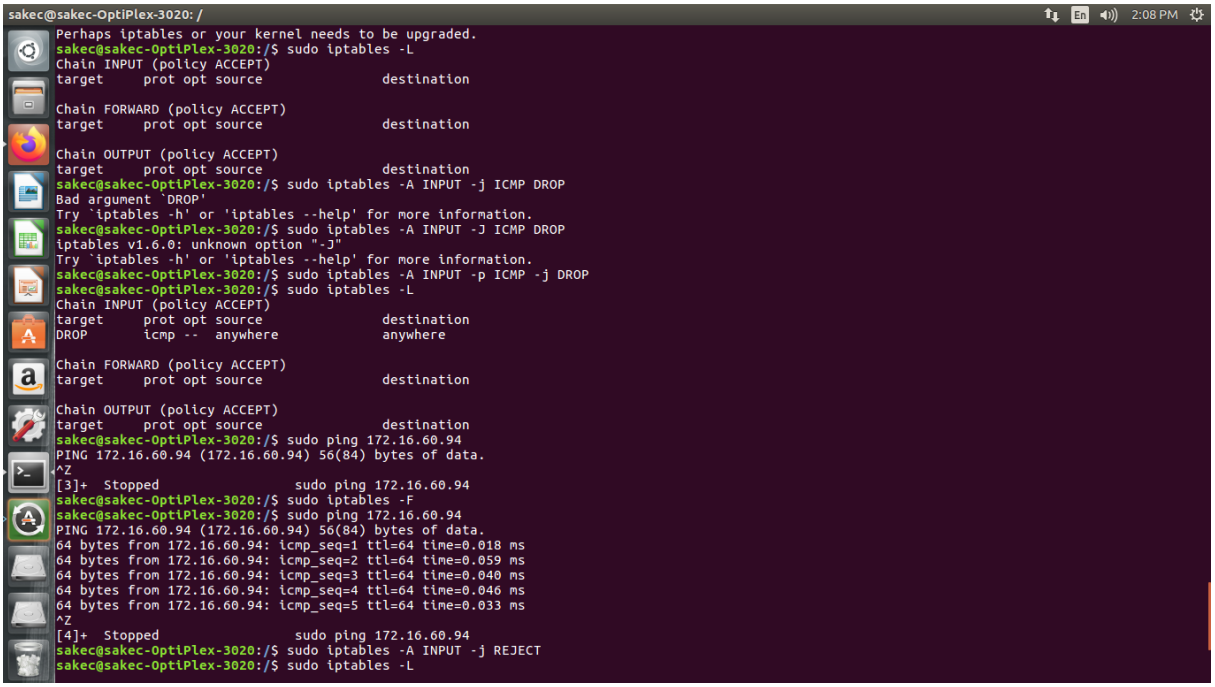
o **FORWARD** — Applies to network packets routed through the host.

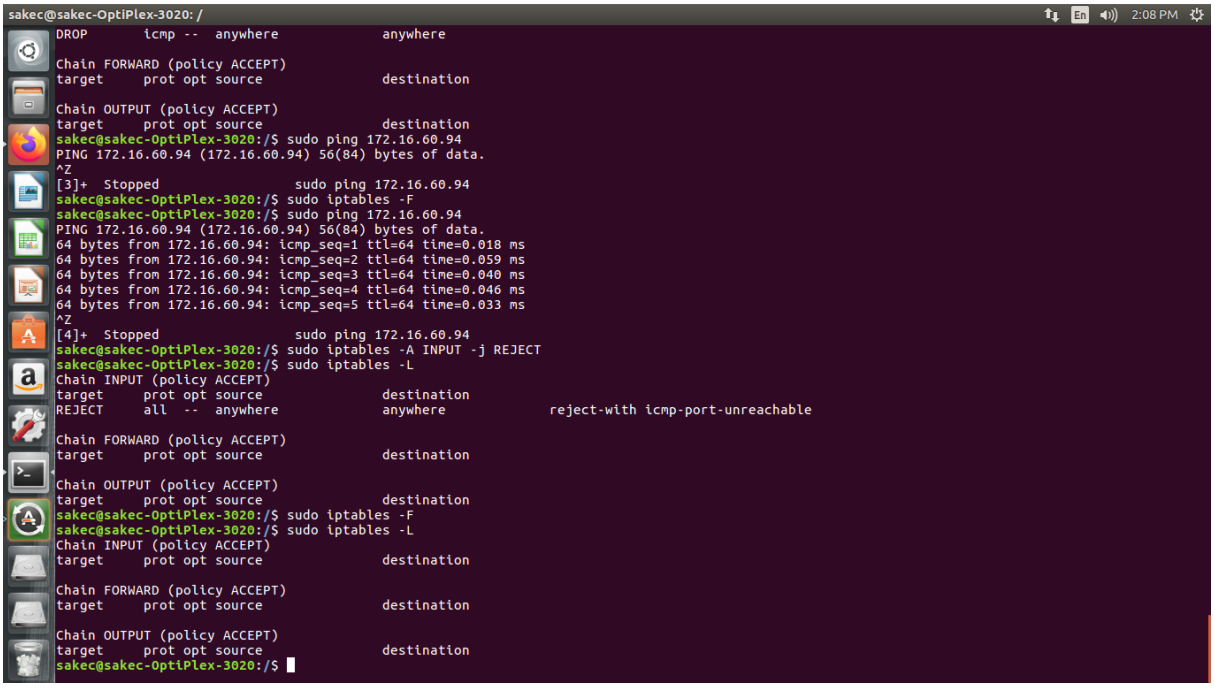
Every chain has a default policy to **ACCEPT**, **DROP** or **REJECT**. If none oftherules in the chain apply to the packet, then the packet is dealt with in accordance with thedefault policy.

Firewall Configuration:

1) #iptables –A INPUT –j DROP
2) Try ping from other machine
3) #iptables –L (list the table)
4) #iptables –F (Flush the table)
5) #iptables – A INPUT –j REJECT (Firewall drop the packet and also send error message)
6) #iptables – A INPUT –j ACCEPT
7) Allow ping but not allow telnet or any other input packet
#iptables –A –p icmp –j ACCEPT
#iptables –A INPUT –j DROP (Reverse this sequence then there is no meaning) 8) For the particular source IP you want to reject. i.e in firewall u identify attack from particular source and then u want to apply rule to that source
#iptables –A INPUT –s 192.168.0.2 –p TCP --dport 23 –j REJECT

Output :





CONCLUSION: Thus, we have studied and successfully add the multiple IP address and also perform actions in Linux