NAME : T MANOJ

EMPID : LYAKE2KHS

DATE : 18-03-2025

Task 1 :

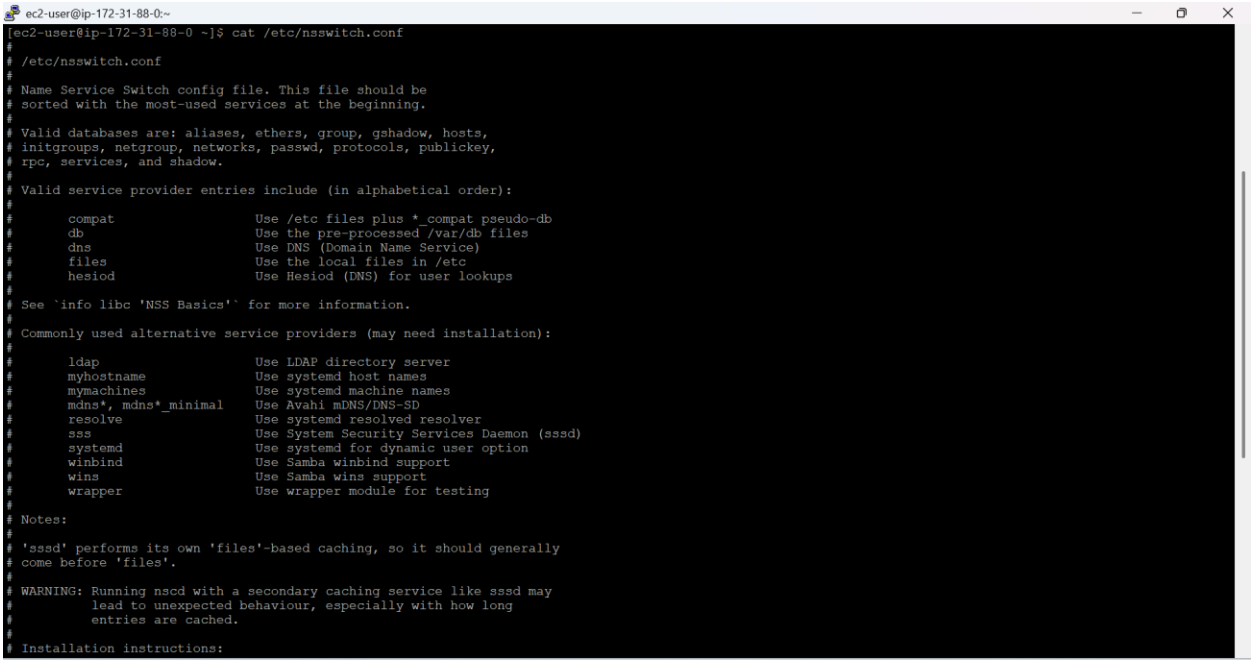
Interface detection : displays all **network interfaces** and their **IP configurations** on your **EC2 instance**.

Ip a

```
[ec2-user@ip-172-31-88-0 ~]$ ip a
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group defaul
t qlen 1000
    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 noprefixroute
       valid_lft forever preferred_lft forever
2: enX0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 9001 qdisc fq_codel state UP grou
p default qlen 1000
    link/ether 12:21:08:16:5a:f7 brd ff:ff:ff:ff:ff:ff
    altname eni-0fcc6d6fcd697811c
    altname device-number-0.0
    inet 172.31.88.0/20 metric 512 brd 172.31.95.255 scope global dynamic enX0
       valid_lft 2811sec preferred_lft 2811sec
    inet6 fe80::1021:8ff:fe16:5af7/64 scope link proto kernel_ll
       valid_lft forever preferred_lft forever
```

2.cmd : cat /etc/nsswitch.conf

Displays the system's Name Service Switch (NSS) configuration



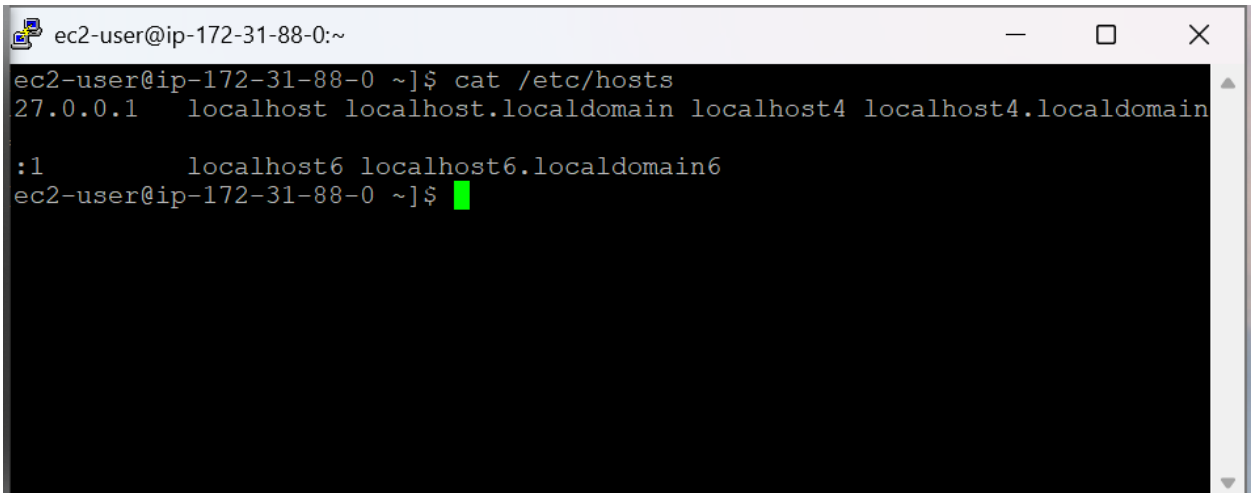

```
#
# Installation instructions:
#
# To use 'db', install the appropriate package(s) (provide 'makedb' and
# libnss_db.so.*), and place the 'db' in front of 'files' for entries
# you want to be looked up first in the databases, like this:
#
# passwd:    db files
# shadow:    db files
# group:     db files

# In order of likelihood of use to accelerate lookup.
passwd:     sss files
shadow:     files
group:      sss files
hosts:      files dns myhostname
services:   files sss
netgroup:   sss
automount:  files sss

aliases:    files
ethers:     files
gshadow:    files
# Allow initgroups to default to the setting for group.
# initgroups: files
networks:   files dns
protocols:  files
publickey:  files
rpc:        files
[ec2-user@ip-172-31-88-0 ~]$
```

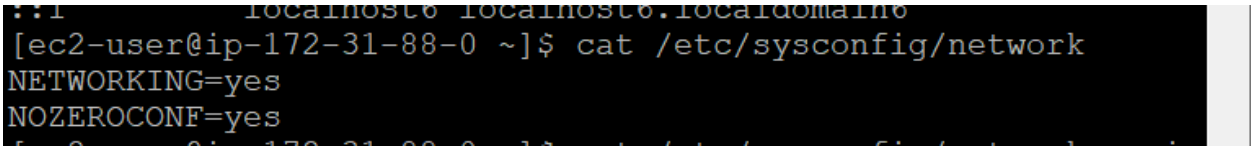
Cmd : cat /etc/hosts

>displays the host information



Cmd : cat /etc/sysconfig/network

system will automatically **apply these network settings every time it starts**.

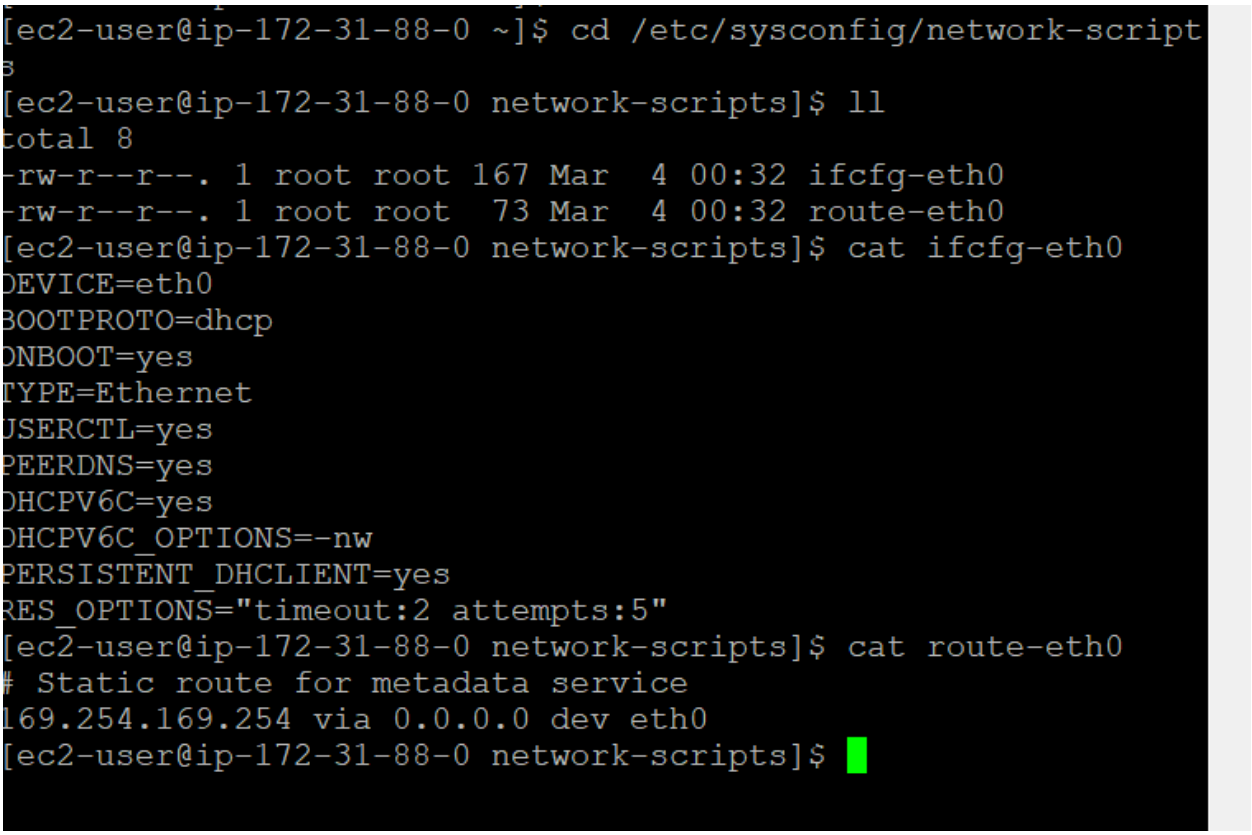


Cmd :

cd /etc/sysconfig/network-scripts

cat ifcfg-eth0

cat route-eth0



`ifcfg-eth0` (Interface Configuration File)

This file contains network settings for the **eth0** interface:

- **DEVICE=eth0** → Specifies the network interface name.
- **BOOTPROTO=dhcp** → The interface obtains an IP address dynamically using **DHCP**.
- **ONBOOT=yes** → The interface will start automatically on boot.
- **TYPE=Ethernet** → Defines the interface type as Ethernet.
- **USERCTL=yes** → Allows non-root users to control the interface.
- **PEERDNS=yes** → Allows the interface to modify **`/etc/resolv.conf`** with DNS settings from DHCP.
- **DHCPV6C=yes** → Enables IPv6 DHCP client.
- **PERSISTENT_DHCLIENT=yes** → Keeps the DHCP client persistent.
- **RES_OPTIONS="timeout:2 attempts:5"** → Specifies DNS resolver options.

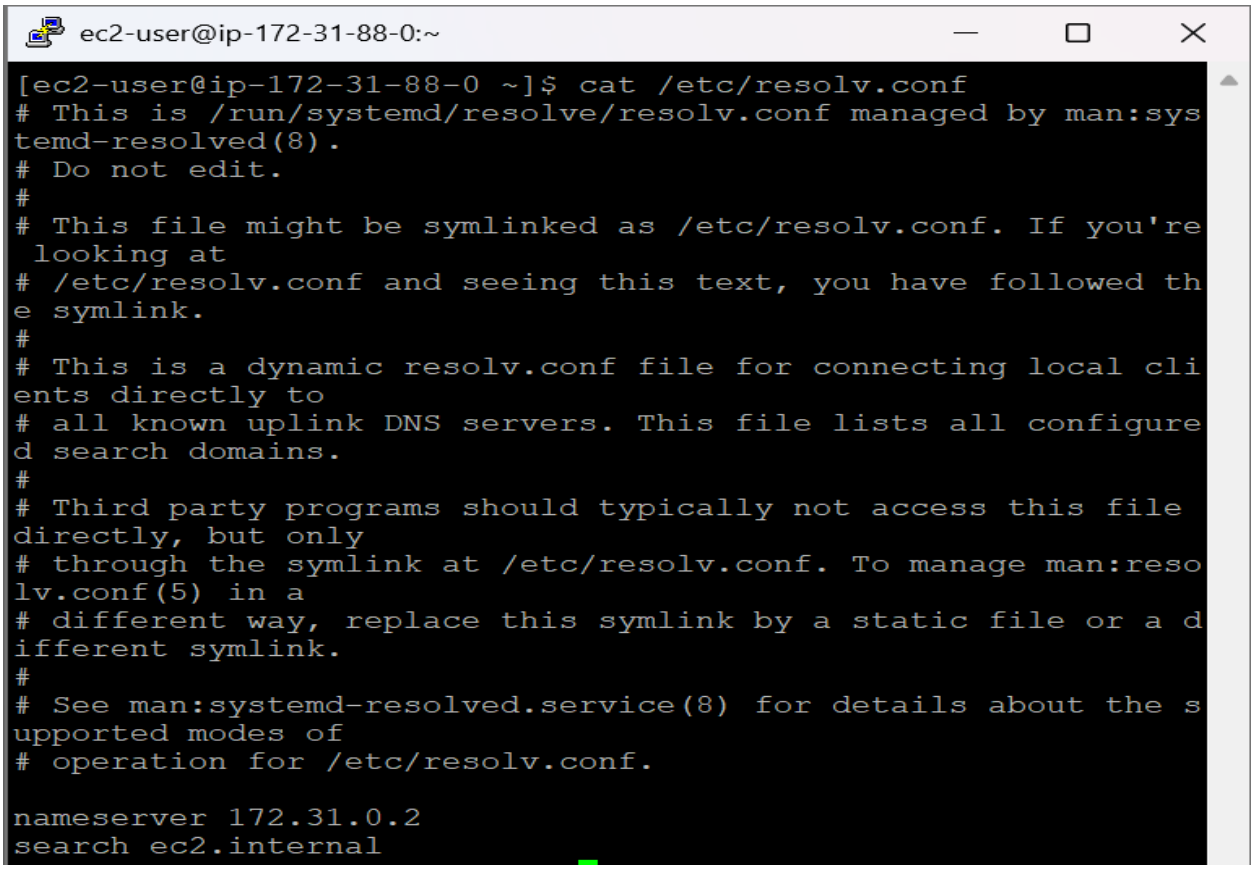
2. `route-eth0` (Static Route Configuration File)

- **169.254.169.254 via 0.0.0.0 dev eth0**
    - This is a **static route for AWS EC2 metadata service**.

- **169.254.169.254** is the AWS instance metadata endpoint.
- The instance routes metadata requests via **eth0**.

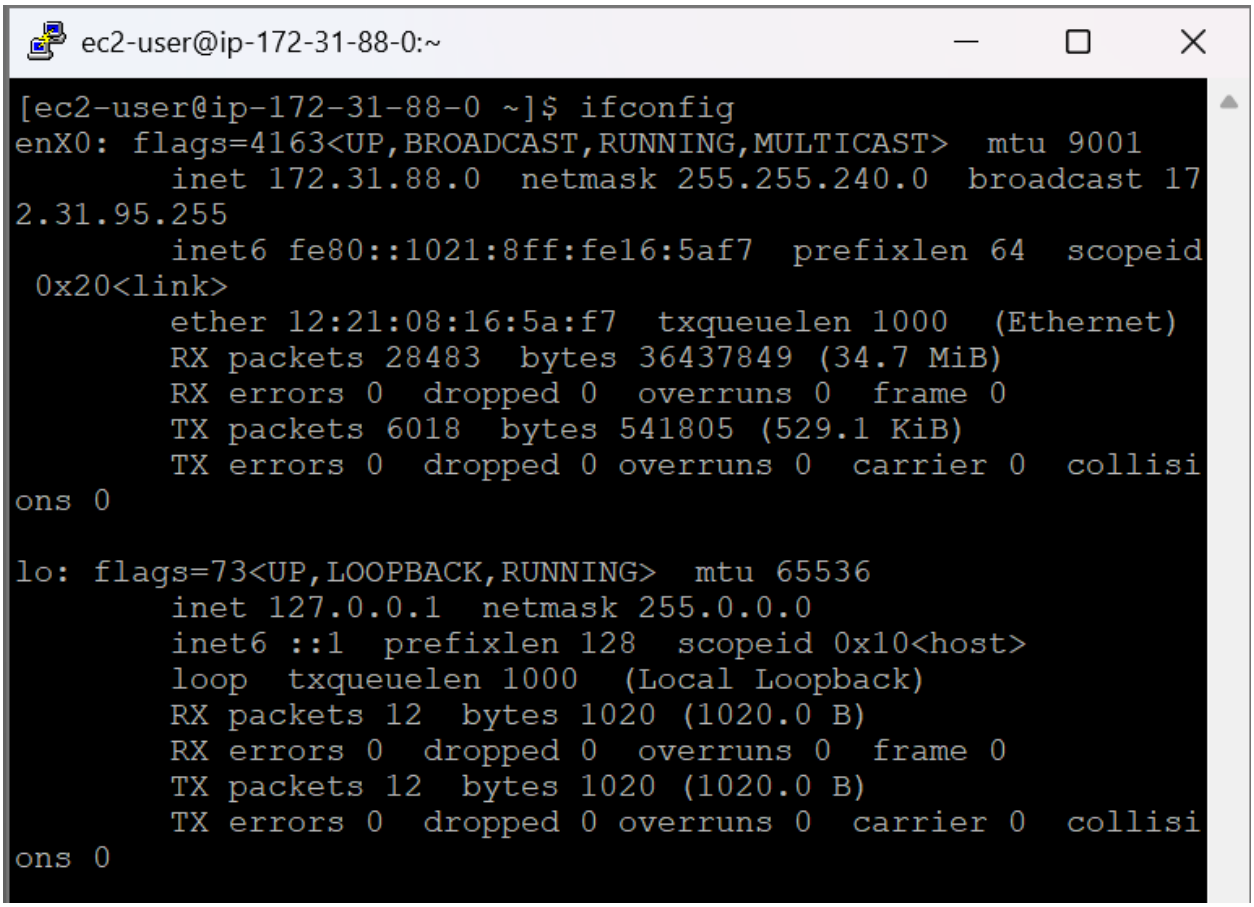
Cmd : cat /etc/resolv.conf

The **`/etc/resolv.conf`** file is used to configure the **Domain Name System (DNS) resolution** on a Linux system. It tells the system which **DNS servers** to use when resolving domain names into IP addresses.

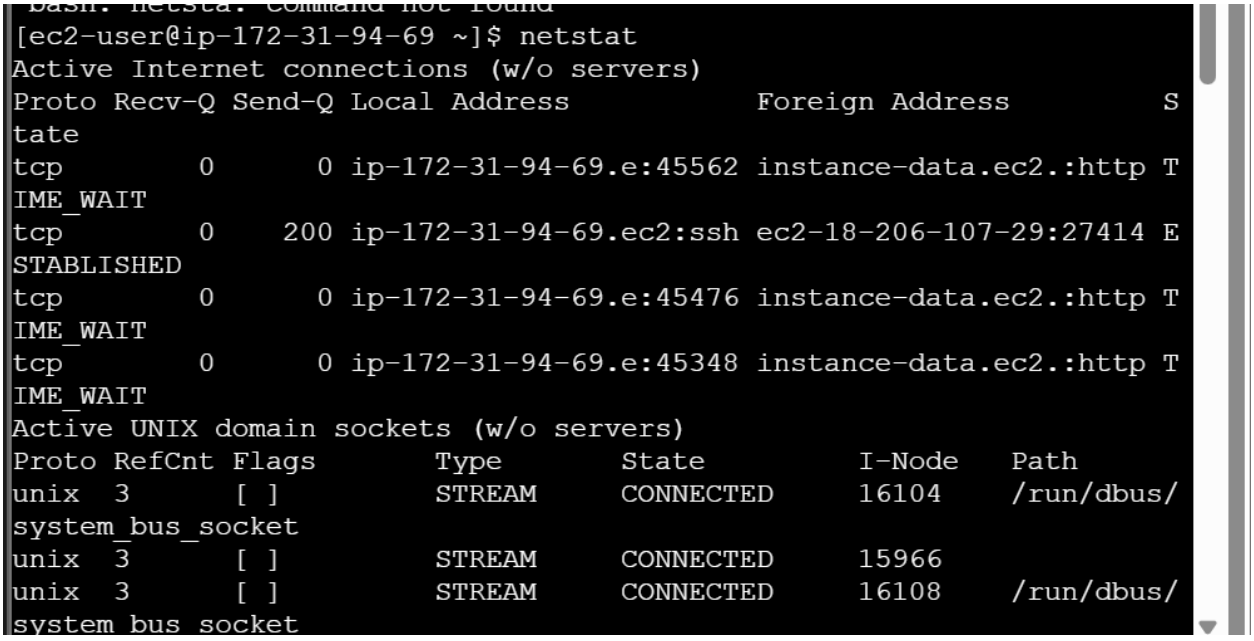


Cmd : ifconfig

>shows the ipaddress


ec2-user@ip-172-31-88-0:~
[ec2-user@ip-172-31-88-0 ~]$ ifconfig
enX0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 9001
inet 172.31.88.0 netmask 255.255.240.0 broadcast 17
2.31.95.255
inet6 fe80::1021:8ff:fe16:5af7 prefixlen 64 scopeid
0x20<link>
ether 12:21:08:16:5a:f7 txqueuelen 1000 (Ethernet)
RX packets 28483 bytes 36437849 (34.7 MiB)
RX errors 0 dropped 0 overruns 0 frame 0
TX packets 6018 bytes 541805 (529.1 KiB)
TX errors 0 dropped 0 overruns 0 carrier 0 collisi
ons 0
lo: flags=73<UP,LOOPBACK,RUNNING> mtu 65536
inet 127.0.0.1 netmask 255.0.0.0
inet6 ::1 prefixlen 128 scopeid 0x10<host>
loop txqueuelen 1000 (Local Loopback)
RX packets 12 bytes 1020 (1020.0 B)
RX errors 0 dropped 0 overruns 0 frame 0
TX packets 12 bytes 1020 (1020.0 B)
TX errors 0 dropped 0 overruns 0 carrier 0 collisi
ons 0

Cmd : netstat


[ec2-user@ip-172-31-94-69 ~]$ netstat
Active Internet connections (w/o servers)
Proto Recv-Q Send-Q Local Address Foreign Address S
tate
tcp 0 0 ip-172-31-94-69.e:45562 instance-data.ec2.:http T
IME_WAIT
tcp 0 200 ip-172-31-94-69.ec2:ssh ec2-18-206-107-29:27414 E
STABLISHED
tcp 0 0 ip-172-31-94-69.e:45476 instance-data.ec2.:http T
IME_WAIT
tcp 0 0 ip-172-31-94-69.e:45348 instance-data.ec2.:http T
IME_WAIT
Active UNIX domain sockets (w/o servers)
Proto RefCnt Flags Type State I-Node Path
unix 3 [ ] STREAM CONNECTED 16104 /run/dbus/
system_bus_socket
unix 3 [ ] STREAM CONNECTED 15966
unix 3 [ ] STREAM CONNECTED 16108 /run/dbus/
system_bus_socket

Cmd : netstat –tulnp

```
[ec2-user@ip-172-31-94-69 ~]$ netstat -tulnp
(Not all processes could be identified, non-owned process info
 will not be shown, you would have to be root to see it all.)
Active Internet connections (only servers)
Proto Recv-Q Send-Q Local Address           Foreign Address         S
tate       PID/Program name
tcp        0      0 0.0.0.0:22              0.0.0.0:*               L
ISTEN      -
tcp6       0      0 :::22                   :::*                    L
ISTEN      -
udp        0      0 172.31.94.69:68         0.0.0.0:*
           -
udp        0      0 127.0.0.1:323           0.0.0.0:*
           -
udp6       0      0 fe80::108a:beff:fe4:546 :::*
           -
udp6       0      0 ::1:323                 :::*
           -
[ec2-user@ip-172-31-94-69 ~]$
```

Cmd :

netstat –u

```
[ec2-user@ip-172-31-94-69 ~]$ netstat -u
Active Internet connections (w/o servers)
Proto Recv-Q Send-Q Local Address           Foreign Address         S
tate
```

Cmd :

Netstat –t

```
[ec2-user@ip-172-31-94-69 ~]$ netstat -t
Active Internet connections (w/o servers)
Proto Recv-Q Send-Q Local Address           Foreign Address         S
tate
tcp        0      0 ip-172-31-94-69.ec2:ssh 58.149.253.83:42812     S
YN_RECV
tcp        0      0 ip-172-31-94-69.ec2:ssh ec2-18-206-107-29:27414 E
STABLISHED
tcp        0      0 ip-172-31-94-69.ec2:ssh 58.149.253.83:52245     E
STABLISHED
```

Cmd :

netstat –an | grep ESTABLISHED

```
[ec2-user@ip-172-31-94-69 ~]$ netstat -an | grep ESTABLISHED
tcp        0      0 172.31.94.69:22         18.206.107.29:27414     E
STABLISHED
[ec2-user@ip-172-31-94-69 ~]$
```

Cmd :

netstat –r

```
[ec2-user@ip-172-31-94-69 ~]$ netstat -r
Kernel IP routing table
Destination     Gateway         Genmask         Flags   MSS Window  i
rtt Iface
default         ip-172-31-80-1. 0.0.0.0         UG        0 0
  0 enX0
ip-172-31-0-2.e ip-172-31-80-1. 255.255.255.255 UGH       0 0
  0 enX0
ip-172-31-80-0. 0.0.0.0         255.255.240.0   U         0 0
  0 enX0
ip-172-31-80-1. 0.0.0.0         255.255.255.255 UH        0 0
  0 enX0
[ec2-user@ip-172-31-94-69 ~]$
```

Firewall :

sudo amazon-linux-extras enable epel

This enables the **Extra Packages for Enterprise Linux (EPEL)** repository on Amazon Linux, allowing access to additional software packages.

```
Running transaction check
Running transaction test
Transaction test succeeded
Running transaction
  Installing : epel-release-7-11.noarch [                                                                                ] 1/
  Installing : epel-release-7-11.noarch [#####                                                                           ] 1/
  Installing : epel-release-7-11.noarch [#########                                                                       ] 1/
  Installing : epel-release-7-11.noarch [############                                                                    ] 1/
  Installing : epel-release-7-11.noarch [####################                                                            ] 1/
  Installing : epel-release-7-11.noarch [################################################################################] 1/
  Installing : epel-release-7-11.noarch                                                                                    1/
1
  Verifying  : epel-release-7-11.noarch                                                                                    1/
1

Installed:
  epel-release.noarch 0:7-11


Complete!
  2  httpd_modules           available     [ =1.0  =stable ]
```

sudo yum install -y epel-release

Installs the EPEL repository package, which provides access to a collection of useful software packages for Amazon Linux and CentOS.

```
230 packages excluded due to repository priority protections
Package epel-release-7-11.noarch already installed and latest version
Nothing to do
[ec2-user@ip-172-31-15-15 ~]$ sudo amazon-linux-extras enable  epel
  2  httpd_modules            available    [ =1.0  =stable ]
  3  memcached1.5             available    \
       [ =1.5.1  =1.5.16  =1.5.17 ]
  9  R3.4                     available    [ =3.4.3  =stable ]
 10  rust1                    available    \
       [ =1.22.1  =1.26.0  =1.26.1  =1.27.2  =1.31.0  =1.38.0
         =stable ]
 18  libreoffice              available    \
       [ =5.0.6.2_15  =5.3.6.1  =stable ]
 19  gimp                     available    [ =2.8.22 ]
 20 †docker=latest            enabled      \
       [ =17.12.1  =18.03.1  =18.06.1  =18.09.9  =stable ]
 21  mate-desktop1.x          available    \
       [ =1.19.0  =1.20.0  =stable ]
 22  GraphicsMagick1.3        available    \
       [ =1.3.29  =1.3.32  =1.3.34  =stable ]
 24  epel=latest              enabled      [ =7.11  =stable ]
```

sudo yum install -y ufw

Installs **UFW (Uncomplicated Firewall)**, a frontend for managing firewall rules.

```
  Installing : ufw-0.35-9.el7.noarch [###############################################################################           ] 1/
  Installing : ufw-0.35-9.el7.noarch [################################################################################          ] 1/
  Installing : ufw-0.35-9.el7.noarch [#################################################################################         ] 1/
  Installing : ufw-0.35-9.el7.noarch [##################################################################################        ] 1/
  Installing : ufw-0.35-9.el7.noarch [###################################################################################       ] 1/
  Installing : ufw-0.35-9.el7.noarch [#####################################################################################     ] 1/
  Installing : ufw-0.35-9.el7.noarch [#########################################################################################  ] 1/
  Installing : ufw-0.35-9.el7.noarch [###########################################################################################] 1/
  Installing : ufw-0.35-9.el7.noarch [############################################################################################  ] 1/
  Installing : ufw-0.35-9.el7.noarch [###############################################################################################] 1/
  Installing : ufw-0.35-9.el7.noarch                                                                                                1/
1
  Verifying  : ufw-0.35-9.el7.noarch                                                                                                1/
1

Installed:
  ufw.noarch 0:0.35-9.el7


Complete!
[ec2-user@ip-172-31-15-15 ~]$
```

sudo systemctl enable --now ufw

Enables UFW to start on boot and immediately starts the service.

```
[ec2-user@ip-172-31-15-15 ~]$ sudo systemctl enable --now ufw
Created symlink from /etc/systemd/system/basic.target.wants/ufw.service to /usr/lib/systemd/system/ufw.service.
[ec2-user@ip-172-31-15-15 ~]$
```

sudo ufw allow shh/22

Allows incoming SSH connections through the firewall (port **22/tcp**).

```
[ec2-user@ip-172-31-15-15 ~]$ sudo ufw allow 22/tcp
Rule added
Rule added (v6)
[ec2-user@ip-172-31-15-15 ~]$
```