

**EX NO:1B**  
**NAME:E.MONIKA**  
**ROLL NO:231901032**

**DATE:27.07.2024**

## BASIC NETWORKING COMMANDS IN LINUX OPERATING SYSTEMS

### Linux Networking Commands

Every computer is connected to some other computer through a network whether internally or externally to exchange some information. This network can be small as some computers connected in your home or office, or can be large or complicated as in large University or the entire Internet.

Maintaining a system's network is a task of System/Network administrator. Their task includes network configuration and troubleshooting.

Here is a list of Networking and Troubleshooting commands:

<a href="#">ifconfig</a>	Display and manipulate route and network interfaces.
<a href="#">ip</a>	It is a replacement of ifconfig command.
<a href="#">traceroute</a>	Network troubleshooting utility.
<a href="#">tracepath</a>	Similar to traceroute but doesn't require root privileges.
<a href="#">ping</a>	To check connectivity between two nodes.
<a href="#">netstat</a>	Display connection information.
<a href="#">ss</a>	It is a replacement of netstat.
<a href="#">dig</a>	Query DNS related information.
<a href="#">nslookup</a>	Find DNS related query.
<a href="#">route</a>	Shows and manipulate IP routing table.
<a href="#">host</a>	Performs DNS lookups.

<a href="#">arp</a>	View or add contents of the kernel's ARP table.
<a href="#">iwconfig</a>	Used to configure wireless network interface.
<a href="#">hostname</a>	To identify a network name.
<a href="#">curl</a> or <a href="#">wget</a>	To download a file from internet.
<a href="#">mtr</a>	Combines ping and tracepath into a single command.
<a href="#">whois</a>	Will tell you about the website's whois.
<a href="#">ifplugstatus</a>	Tells whether a cable is plugged in or not.

Explanation of the above commands:

**1.ifconfig:** ifconfig is short for interface configurator. This command is utilized in network inspection, initializing the interface, enabling or disabling an IP address, and configuring an interface with an IP address. Also, it is used to show the network and route interface. The basic details shown with ifconfig are:

- MTU
- MAC address
- IP address

## Syntax:

### Ifconfig

```
root@ip-10-10-10-111:~# ifconfig
docker0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
    inet 172.17.0.1 netmask 255.255.0.0 broadcast 172.17.255.255
    inet6 fe80::42:1dff:fe3a:24d5 prefixlen 64 scopeid 0x20<link>
    ether 02:42:10:c4:24:d5 txqueuelen 0 (Ethernet)
    RX packets 0 bytes 0 (0.0 B)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 35 bytes 4761 (4.7 KB)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

ens3: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 9001
    inet 10.10.10.111 netmask 255.255.0.0 broadcast 10.10.255.255
    inet6 fe80::4a:89ff:fe31:79dd prefixlen 64 scopeid 0x20<link>
    ether 02:4a:89:31:79:dd txqueuelen 1000 (Ethernet)
    RX packets 8907 bytes 715504 (715.5 KB)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 6758 bytes 4159018 (4.1 MB)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 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 15768 bytes 4086788 (4.0 MB)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 15768 bytes 4086788 (4.0 MB)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

vethdadb7c: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
    inet6 fe80::540c:a4ff:fe12:a53b prefixlen 64 scopeid 0x20<link>
    ether 56:0c:a4:12:a5:3b txqueuelen 0 (Ethernet)
    RX packets 0 bytes 0 (0.0 B)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 57 bytes 7476 (7.4 KB)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

vethf09dcf2: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
    inet6 fe80::1040:daff:feb4:03f2 prefixlen 64 scopeid 0x20<link>
    ether 12:40:da:84:d9:f2 txqueuelen 0 (Ethernet)
    RX packets 0 bytes 0 (0.0 B)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 18 bytes 7546 (7.5 KB)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0
```

**2. ip:** It is the updated and latest edition of ifconfig command. The command provides the information of every network, such as ifconfig. Also, it can be used to get information about a particular interface. **Syntax:**

1. ip a
2. ip addr

[illegible]

3. **tracert:** The tracert command is one of the most helpful commands in the networking field. It's used to balance the network. It identifies the delay and decides the pathway to our target. Basically, it aids in the below ways:

- It determines the location of the network latency and informs it.
- It follows the path to the destination.
- It gives the names and recognizes all devices on the path.

### Syntax:

tracert <destination>

```
root@ip-10-10-10-111:~# traceroute www.google.com
traceroute to www.google.com (209.85.202.104), 30 hops max, 60 byte packets
 1 * * *
 2 * * *
 3 * * *
 4 * * *
 5 * * *
 6 * * *
 7 * * *
 8 * * *
 9 * * *
10 * * *
11 * * *
12 * * *
13 * * *
14 * * *
15 * * *
16 * * *
17 * * *
18 * * *
19 * * *
20 * * *
21 * * *
22 * * *
23 * * *
24 * * *
25 * * *
26 * * *
27 * * *
28 * * *
29 * * *
30 * * *
```

4. **tracpath:** The `tracpath` command is the same as the `traceroute` command, and it is used to find network delays. Besides, it does not need root privileges. By default, it comes preinstalled in Ubuntu. It traces the path to the destination and recognizes all hops in it. It identifies the point at which the network is weak if our network is not strong enough.

**Syntax:** tracepath

**<destination>**

```

root@ip-10-10-38-111:~# tracert www.google.com
17: [LOCALHOST]                                pmtu 1500
1: no reply
2: no reply
3: no reply
4: no reply
5: no reply
6: no reply
7: no reply
8: no reply
9: no reply
10: no reply
11: no reply
12: no reply
13: no reply
14: no reply
15: no reply

```

5. **ping:** It is short for Packet Internet Groper. The ping command is one of the widely used commands for network troubleshooting. Basically, it inspects the network connectivity between two different nodes.

**Syntax:** ping

<destination>

```

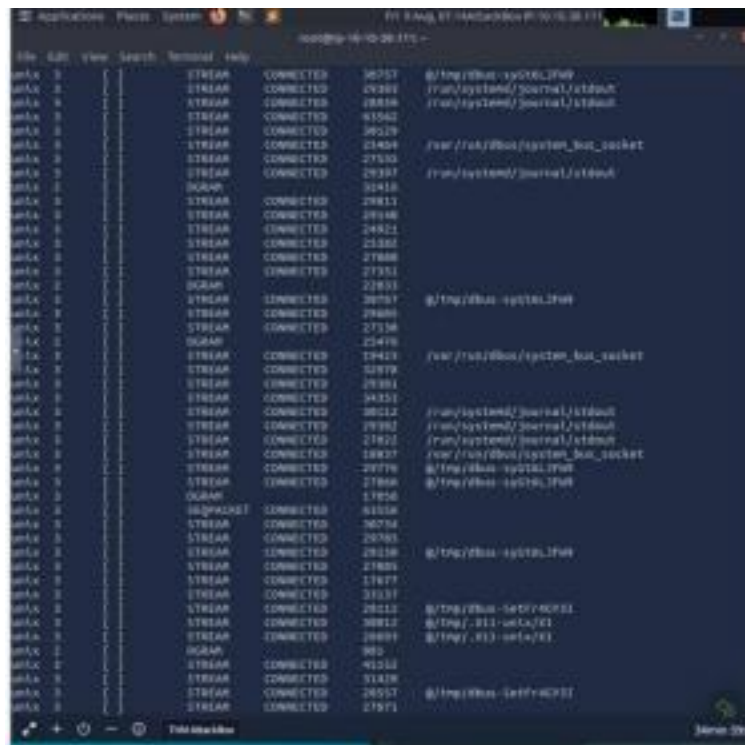
root@ip-10-10-38-111:~# ping 10.10.38.111
ping: 10.10.38.1: Name or service not known
root@ip-10-10-38-111:~# ping 10.10.38.111
PING 10.10.38.111 (10.10.38.111): 56(84) bytes of data.
64 bytes from 10.10.38.111: icmp_seq=1 ttl=64 time=0.042 ms
64 bytes from 10.10.38.111: icmp_seq=2 ttl=64 time=0.037 ms
64 bytes from 10.10.38.111: icmp_seq=3 ttl=64 time=0.027 ms
64 bytes from 10.10.38.111: icmp_seq=4 ttl=64 time=0.026 ms
64 bytes from 10.10.38.111: icmp_seq=5 ttl=64 time=0.031 ms
64 bytes from 10.10.38.111: icmp_seq=6 ttl=64 time=0.044 ms
64 bytes from 10.10.38.111: icmp_seq=7 ttl=64 time=0.025 ms
64 bytes from 10.10.38.111: icmp_seq=8 ttl=64 time=0.027 ms
64 bytes from 10.10.38.111: icmp_seq=9 ttl=64 time=0.032 ms
64 bytes from 10.10.38.111: icmp_seq=10 ttl=64 time=0.025 ms
64 bytes from 10.10.38.111: icmp_seq=11 ttl=64 time=0.035 ms
64 bytes from 10.10.38.111: icmp_seq=12 ttl=64 time=0.029 ms
64 bytes from 10.10.38.111: icmp_seq=13 ttl=64 time=0.041 ms
64 bytes from 10.10.38.111: icmp_seq=14 ttl=64 time=0.044 ms
64 bytes from 10.10.38.111: icmp_seq=15 ttl=64 time=0.039 ms
64 bytes from 10.10.38.111: icmp_seq=16 ttl=64 time=0.030 ms
64 bytes from 10.10.38.111: icmp_seq=17 ttl=64 time=0.048 ms
64 bytes from 10.10.38.111: icmp_seq=18 ttl=64 time=0.027 ms
64 bytes from 10.10.38.111: icmp_seq=19 ttl=64 time=0.032 ms
64 bytes from 10.10.38.111: icmp_seq=20 ttl=64 time=0.042 ms
64 bytes from 10.10.38.111: icmp_seq=21 ttl=64 time=0.027 ms
64 bytes from 10.10.38.111: icmp_seq=22 ttl=64 time=0.025 ms
64 bytes from 10.10.38.111: icmp_seq=23 ttl=64 time=0.037 ms
64 bytes from 10.10.38.111: icmp_seq=24 ttl=64 time=0.042 ms
64 bytes from 10.10.38.111: icmp_seq=25 ttl=64 time=0.181 ms
64 bytes from 10.10.38.111: icmp_seq=26 ttl=64 time=0.029 ms
64 bytes from 10.10.38.111: icmp_seq=27 ttl=64 time=0.043 ms
64 bytes from 10.10.38.111: icmp_seq=28 ttl=64 time=0.025 ms
64 bytes from 10.10.38.111: icmp_seq=29 ttl=64 time=0.032 ms
64 bytes from 10.10.38.111: icmp_seq=30 ttl=64 time=0.031 ms
64 bytes from 10.10.38.111: icmp_seq=31 ttl=64 time=0.047 ms
^C
--- 10.10.38.111 ping statistics ---
31 packets transmitted, 31 received, 0% packet loss, time 3872ms
rtt min/avg/max/mdev = 0.022/0.038/0.181/0.027 ms
root@ip-10-10-38-111:~#

```

6. **netstat:** It is short for network statistics. It gives statistical figures of many interfaces, which contain open sockets, connection information, and routing tables.

**Syntax:**

Netstat



7. **ss:** This command is the substitution for the netstat command. The ss command is more informative and much faster than netstat. The ss command's faster response is possible because it fetches every information from inside the kernel userspace.

**Syntax:**

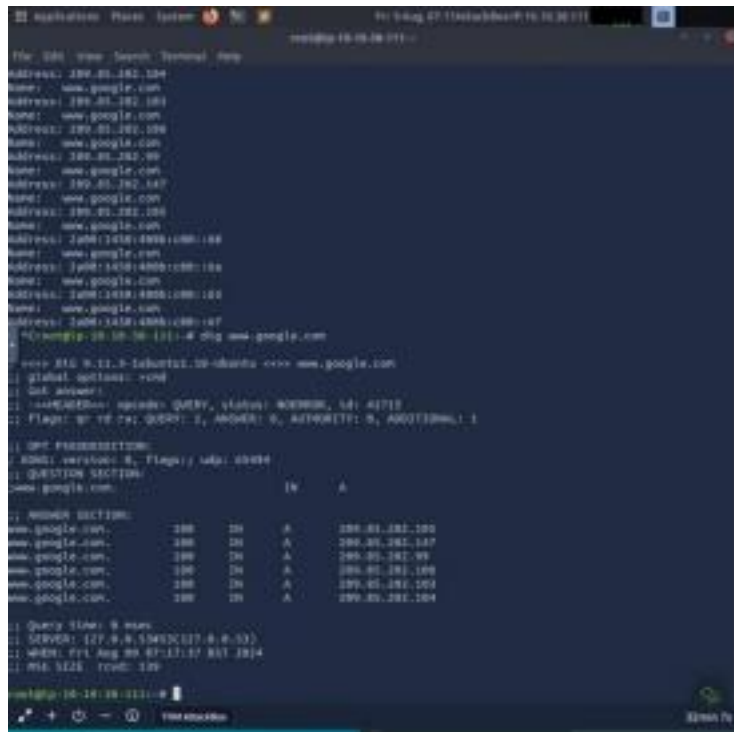
**Ss**



8. **nslookup:** The nslookup command is an older edition of the dig command. Also, it is







10. **route:** The route command shows and employs the routing table available for our system. Basically, a router is used to detect a better way to transfer the packets around a destination.

## Syntax: Route

11. **host:** The host command shows the IP address for a hostname and the domain name for an IP address. Also, it is used to get DNS lookup for DNS related issues.

**Syntax:**

```
host -t <resourceName>
```

12. **arp:** The arp command is short for Address Resolution Protocol. This command is used to see and include content in the ARP table of the kernel.

## Arp

13. **iwconfig:** It is a simple command which is used to see and set the system's hostname.

### Syntax:

## Hostname

14. **curl and wget:** These commands are used to download files from CLI from the internet. curl must be specified with the "O" option to get the file, while wget is directly used.

## curl Syntax:

```
curl -O <fileLink>
```

## 15. wget

### Syntax:

```
wget <fileLink>
```

16. **mtr:** The mtr command is a mix of the traceroute and ping commands. It regularly shows information related to the packets transferred using the ping time of all hops. Also, it is used to see network problems.

**Syntax:**

mtr <path>

17. **whois:** The whois command fetches every website related information. We can get every information of a website, such as an owner and the registration information.

**Syntax:**

mtr <websiteName>

18. **ifplugstatus:** The ifplugstatus command checks whether a cable is currently plugged into a network interface. It is not available in Ubuntu directly. We can install it with the help of the below command:

sudo apt-get install ifplugd **Syntax:**

**Ifplugstatus iftop:** The iftop command is utilized in traffic monitoring. **tcpdump:** The tcpdump command is widely used in network analysis with other commands of the Linux network. It analyses the traffic passing from the network interface and shows it.

When balancing the network, this type of packet access will be crucial.

**Syntax:** \$ tcpdump -i <network\_device>

**RESULT:**

Hence, Linux commands are executed successfully.