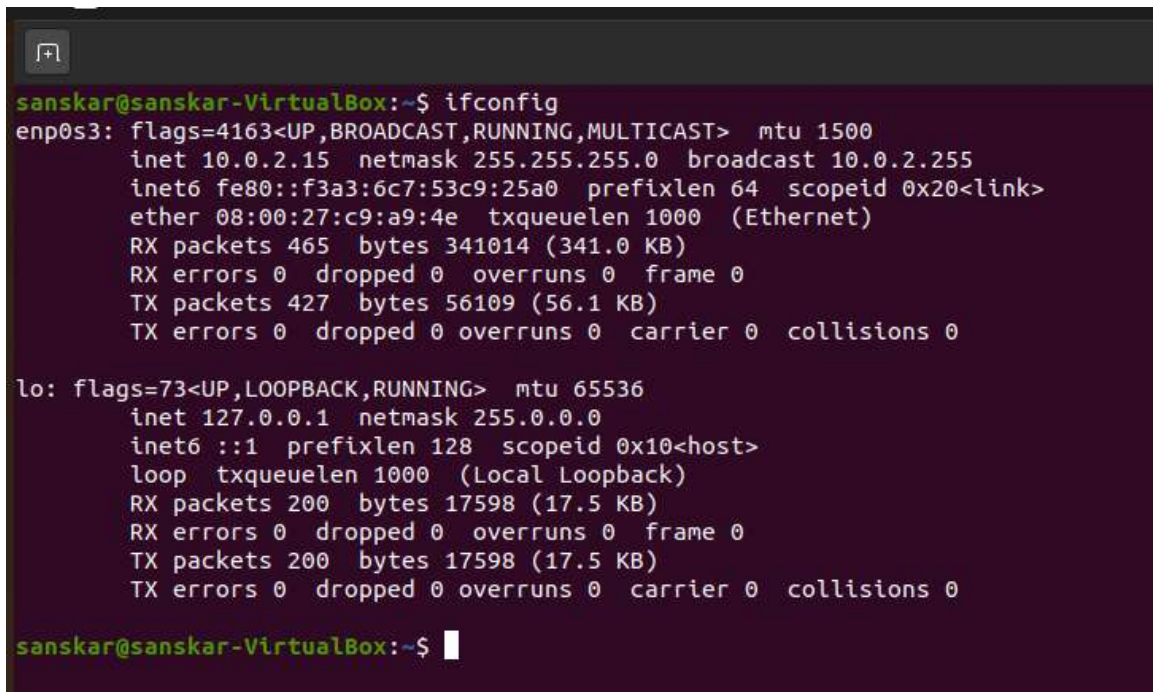


1. **ifconfig**:(interface configuration) command is used to configure the kernel-resident network interfaces. It is used at the boot time to set up the interfaces as necessary. After that, it is usually used when needed during debugging or when you need system tuning. Also, this command is used to assign the IP address and netmask to an interface or to enable or disable a given interface.

A terminal window with a dark purple background and a terminal icon in the top-left corner. The prompt is 'sanskar@sanskar-VirtualBox:~\$'. The command 'ifconfig' has been executed, showing details for two interfaces: 'enp0s3' and 'lo'. The output for 'enp0s3' includes flags, mtu, inet and inet6 addresses, broadcast address, ether address, txqueuelen, and RX/TX statistics. The output for 'lo' includes flags, mtu, inet and inet6 addresses, loopback address, txqueuelen, and RX/TX statistics. The prompt 'sanskar@sanskar-VirtualBox:~\$' is shown again at the bottom.

```
sanskar@sanskar-VirtualBox:~$ ifconfig
enp0s3: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 1500
        inet 10.0.2.15  netmask 255.255.255.0  broadcast 10.0.2.255
        inet6 fe80::f3a3:6c7:53c9:25a0  prefixlen 64  scopeid 0x20<link>
        ether 08:00:27:c9:a9:4e  txqueuelen 1000  (Ethernet)
        RX packets 465  bytes 341014 (341.0 KB)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 427  bytes 56109 (56.1 KB)
        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 200  bytes 17598 (17.5 KB)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 200  bytes 17598 (17.5 KB)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0

sanskar@sanskar-VirtualBox:~$
```

2. **ip**: command in Linux is present in the net-tools which is used for performing several network administration tasks. IP stands for Internet Protocol. This command is used to show or manipulate routing, devices, and tunnels. It is similar to ifconfig command but it is much more powerful with more functions and facilities attached to it. *ifconfig* is one of the deprecated commands in the net-tools of Linux that has not been maintained for many years. ip command is used to perform several tasks like assigning an address to a network interface or configuring network interface parameters. It can perform several other tasks like configuring and modifying the default and static routing, setting up tunnel over

IP, listing IP addresses and property information, modifying the status of the interface, assigning, deleting and setting up IP addresses and routes.

```
es  Terminal ▾

sanskar@sanskar-VirtualBox:~$ ip
Usage: ip [ OPTIONS ] OBJECT { COMMAND | help }
       ip [ -force ] -batch filename
where  OBJECT := { link | address | addrlabel | route | rule | neigh | ntable |
                  tunnel | tuntap | maddress | mroute | mrule | monitor | xfrm |
                  netns | lztp | fou | macsec | tcp_metrics | token | netconf | illa |
                  vrf | sr | nexthop }
       OPTIONS := { -V[ersion] | -s[tatistics] | -d[etails] | -r[esolve] |
                    -h[uman-readable] | -l[eac] | -j[son] | -p[retty] |
                    -f[amily] { inet | inet6 | mpls | bridge | link } |
                    -4 | -6 | -I | -D | -M | -B | -O |
                    -l[oops] { maximum-addr-flush-attempts } | -br[ief] |
                    -o[neline] | -t[imestamp] | -ts[hort] | -b[atch] [filename] |
                    -rc[vbuf] [size] | -n[etns] name | -N[umeric] | -a[ll] |
                    -c[olor]}
sanskar@sanskar-VirtualBox:~$
```

```
sanskar@sanskar-VirtualBox:~$ ip address
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default 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
        valid_lft forever preferred_lft forever
2: enp0s3: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP group default qlen 1000
    link/ether 08:00:27:c9:a9:4e brd ff:ff:ff:ff:ff:ff
    inet 10.0.2.15/24 brd 10.0.2.255 scope global dynamic noprefixroute enp0s3
        valid_lft 85924sec preferred_lft 85924sec
    inet6 fe80::f3a3:6c7:53c9:25a0/64 scope link noprefixroute
        valid_lft forever preferred_lft forever
sanskar@sanskar-VirtualBox:~$
```

3. **traceroute**: command in Linux prints the route that a packet takes to reach the host. This command is useful when you want to know about the route and about all the hops that a packet takes.

```
sanskar@sanskar-VirtualBox:~$ traceroute google.com
traceroute to google.com (142.250.192.110), 30 hops max, 60 byte packets
 1  _gateway (10.0.2.2)  1.224 ms  12.174 ms  12.001 ms
 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  * * *
sanskar@sanskar-VirtualBox:~$
```

4. **tracepath**: command in Linux is used to traces path to destination discovering MTU along this path. It uses UDP port or some random port. It is similar to traceroute, but it does not require superuser privileges and has no fancy options. *tracepath6* is a good replacement for *traceroute6* and classic example of the application of Linux error queues.

```
sanskar@sanskar-VirtualBox:~$ tracepath www.google.com
1?: [LOCALHOST] pmtu 1500
1: _gateway 0.627ms
1: _gateway 0.879ms
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
16: no reply
17: no reply
18: no reply
19: no reply
20: no reply
21: no reply
22: no reply
23: no reply
24: no reply
25: no reply
26: no reply
27: no reply
28: no reply
29: no reply
30: no reply
Too many hops: pmtu 1500
Resume: pmtu 1500
sanskar@sanskar-VirtualBox:~$
```

5. **PING** :(Packet Internet Groper) command is used to check the network connectivity between host and server/host. This command takes as input the IP address or the URL and sends a data packet to the specified address with the message “PING” and get a response from the server/host this time is recorded which is called latency.



```
sanskar@sanskar-VirtualBox:~$ ping javatpoint.com
PING javatpoint.com (104.21.23.99) 56(84) bytes of data.
64 bytes from 104.21.23.99 (104.21.23.99): icmp_seq=1 ttl=45 time=141 ms
64 bytes from 104.21.23.99 (104.21.23.99): icmp_seq=2 ttl=45 time=214 ms
64 bytes from 104.21.23.99 (104.21.23.99): icmp_seq=3 ttl=45 time=124 ms
64 bytes from 104.21.23.99 (104.21.23.99): icmp_seq=4 ttl=45 time=145 ms
64 bytes from 104.21.23.99 (104.21.23.99): icmp_seq=5 ttl=45 time=309 ms
64 bytes from 104.21.23.99 (104.21.23.99): icmp_seq=6 ttl=45 time=128 ms
64 bytes from 104.21.23.99 (104.21.23.99): icmp_seq=7 ttl=45 time=150 ms
64 bytes from 104.21.23.99 (104.21.23.99): icmp_seq=8 ttl=45 time=350 ms
64 bytes from 104.21.23.99 (104.21.23.99): icmp_seq=9 ttl=45 time=157 ms
64 bytes from 104.21.23.99 (104.21.23.99): icmp_seq=10 ttl=45 time=193 ms
64 bytes from 104.21.23.99 (104.21.23.99): icmp_seq=11 ttl=45 time=137 ms
64 bytes from 104.21.23.99 (104.21.23.99): icmp_seq=12 ttl=45 time=131 ms
64 bytes from 104.21.23.99 (104.21.23.99): icmp_seq=13 ttl=45 time=126 ms
64 bytes from 104.21.23.99 (104.21.23.99): icmp_seq=14 ttl=45 time=258 ms
64 bytes from 104.21.23.99 (104.21.23.99): icmp_seq=15 ttl=45 time=151 ms
64 bytes from 104.21.23.99 (104.21.23.99): icmp_seq=16 ttl=45 time=136 ms
64 bytes from 104.21.23.99 (104.21.23.99): icmp_seq=17 ttl=45 time=174 ms
64 bytes from 104.21.23.99 (104.21.23.99): icmp_seq=18 ttl=45 time=155 ms
64 bytes from 104.21.23.99 (104.21.23.99): icmp_seq=19 ttl=45 time=172 ms
64 bytes from 104.21.23.99 (104.21.23.99): icmp_seq=20 ttl=45 time=687 ms
64 bytes from 104.21.23.99 (104.21.23.99): icmp_seq=21 ttl=45 time=623 ms
64 bytes from 104.21.23.99 (104.21.23.99): icmp_seq=22 ttl=45 time=356 ms
64 bytes from 104.21.23.99 (104.21.23.99): icmp_seq=23 ttl=45 time=136 ms
64 bytes from 104.21.23.99 (104.21.23.99): icmp_seq=24 ttl=45 time=192 ms
64 bytes from 104.21.23.99 (104.21.23.99): icmp_seq=25 ttl=45 time=264 ms
64 bytes from 104.21.23.99 (104.21.23.99): icmp_seq=26 ttl=45 time=263 ms
64 bytes from 104.21.23.99 (104.21.23.99): icmp_seq=27 ttl=45 time=167 ms
64 bytes from 104.21.23.99 (104.21.23.99): icmp_seq=28 ttl=45 time=124 ms
64 bytes from 104.21.23.99 (104.21.23.99): icmp_seq=29 ttl=45 time=125 ms
64 bytes from 104.21.23.99 (104.21.23.99): icmp_seq=30 ttl=45 time=254 ms
64 bytes from 104.21.23.99 (104.21.23.99): icmp_seq=31 ttl=45 time=351 ms
64 bytes from 104.21.23.99 (104.21.23.99): icmp_seq=32 ttl=45 time=130 ms
64 bytes from 104.21.23.99 (104.21.23.99): icmp_seq=33 ttl=45 time=335 ms
64 bytes from 104.21.23.99 (104.21.23.99): icmp_seq=34 ttl=45 time=234 ms
64 bytes from 104.21.23.99 (104.21.23.99): icmp_seq=35 ttl=45 time=141 ms
64 bytes from 104.21.23.99 (104.21.23.99): icmp_seq=36 ttl=45 time=319 ms
64 bytes from 104.21.23.99 (104.21.23.99): icmp_seq=37 ttl=45 time=148 ms
64 bytes from 104.21.23.99 (104.21.23.99): icmp_seq=38 ttl=45 time=178 ms
64 bytes from 104.21.23.99 (104.21.23.99): icmp_seq=39 ttl=45 time=212 ms
64 bytes from 104.21.23.99 (104.21.23.99): icmp_seq=40 ttl=45 time=156 ms
64 bytes from 104.21.23.99 (104.21.23.99): icmp_seq=41 ttl=45 time=129 ms
64 bytes from 104.21.23.99 (104.21.23.99): icmp_seq=42 ttl=45 time=228 ms
64 bytes from 104.21.23.99 (104.21.23.99): icmp_seq=43 ttl=45 time=161 ms
64 bytes from 104.21.23.99 (104.21.23.99): icmp_seq=44 ttl=45 time=128 ms
64 bytes from 104.21.23.99 (104.21.23.99): icmp_seq=45 ttl=45 time=299 ms
64 bytes from 104.21.23.99 (104.21.23.99): icmp_seq=46 ttl=45 time=247 ms
64 bytes from 104.21.23.99 (104.21.23.99): icmp_seq=47 ttl=45 time=367 ms
```

6. **Netstat:** command displays various network related information such as network connections, routing tables, interface statistics, masquerade connections, multicast memberships etc.,



```
sanskar@sanskar-VirtualBox:~$ netstat
```

```
Active Internet connections (w/o servers)
```

Proto	Recv-Q	Send-Q	Local Address	Foreign Address	State
udp	0	0	sanskar-VirtualB:bootpc	_gateway:bootps	ESTABLISHED

```
Active UNIX domain sockets (w/o servers)
```

Proto	RefCnt	Flags	Type	State	I-Node	Path
unix	3	[ ]	DGRAM		17437	/run/systemd/notify
unix	2	[ ]	DGRAM		33752	/run/user/1000/systemd/notify
unix	2	[ ]	DGRAM		17451	/run/systemd/journal/syslog
unix	16	[ ]	DGRAM		17461	/run/systemd/journal/dev-log
unix	8	[ ]	DGRAM		17465	/run/systemd/journal/socket
unix	3	[ ]	STREAM	CONNECTED	33513	
unix	3	[ ]	STREAM	CONNECTED	23869	/run/systemd/journal/stdout
unix	3	[ ]	STREAM	CONNECTED	37445	
unix	3	[ ]	STREAM	CONNECTED	34794	/run/user/1000/bus
unix	3	[ ]	STREAM	CONNECTED	36278	
unix	3	[ ]	STREAM	CONNECTED	36259	
unix	3	[ ]	STREAM	CONNECTED	34825	
unix	3	[ ]	STREAM	CONNECTED	32397	/run/systemd/journal/stdout
unix	3	[ ]	STREAM	CONNECTED	39154	
unix	3	[ ]	STREAM	CONNECTED	37611	/run/dbus/system_bus_socket
unix	3	[ ]	STREAM	CONNECTED	40002	/run/user/1000/bus
unix	3	[ ]	STREAM	CONNECTED	38121	
unix	3	[ ]	STREAM	CONNECTED	37353	/run/systemd/journal/stdout
unix	3	[ ]	STREAM	CONNECTED	32675	/run/systemd/journal/stdout
unix	3	[ ]	STREAM	CONNECTED	40053	/run/user/1000/bus
unix	3	[ ]	STREAM	CONNECTED	38953	
unix	3	[ ]	STREAM	CONNECTED	37031	/run/systemd/journal/stdout
unix	3	[ ]	STREAM	CONNECTED	37120	/run/systemd/journal/stdout
unix	3	[ ]	STREAM	CONNECTED	36276	@/home/sanskar/.cache/ibus/dbus-732V2Psr
unix	3	[ ]	STREAM	CONNECTED	36313	@/home/sanskar/.cache/ibus/dbus-732V2Psr
unix	3	[ ]	STREAM	CONNECTED	32694	/run/systemd/journal/stdout
unix	2	[ ]	DGRAM		24590	
unix	3	[ ]	STREAM	CONNECTED	37393	/run/systemd/journal/stdout
unix	3	[ ]	STREAM	CONNECTED	34598	
unix	3	[ ]	STREAM	CONNECTED	34136	/run/user/1000/bus
unix	3	[ ]	STREAM	CONNECTED	37952	
unix	3	[ ]	STREAM	CONNECTED	30691	
unix	3	[ ]	STREAM	CONNECTED	29491	
unix	3	[ ]	STREAM	CONNECTED	22956	
unix	3	[ ]	STREAM	CONNECTED	38327	@/tmp/.X11-unix/X0
unix	3	[ ]	STREAM	CONNECTED	38232	@/tmp/dbus-Vt8tyOHCjB
unix	3	[ ]	STREAM	CONNECTED	25054	/run/dbus/system_bus_socket
unix	3	[ ]	STREAM	CONNECTED	24744	/run/dbus/system_bus_socket
unix	3	[ ]	STREAM	CONNECTED	37121	
unix	3	[ ]	STREAM	CONNECTED	37076	/run/user/1000/bus
unix	3	[ ]	STREAM	CONNECTED	35404	
unix	3	[ ]	STREAM	CONNECTED	22198	
unix	3	[ ]	STREAM	CONNECTED	21782	

7. **Nslookup:** (stands for “Name Server Lookup”) is a useful command for getting information from the DNS server. It is a network administration tool for querying the Domain Name System (DNS) to obtain domain name or IP address mapping or any other specific DNS record. It is also used to troubleshoot DNS-related problems.

```
sanskar@sanskar-VirtualBox:~$ nslookup yahoo.com
Server:          127.0.0.53
Address:         127.0.0.53#53

Non-authoritative answer:
Name:   yahoo.com
Address: 74.6.143.25
Name:   yahoo.com
Address: 74.6.143.26
Name:   yahoo.com
Address: 74.6.231.20
Name:   yahoo.com
Address: 74.6.231.21
Name:   yahoo.com
Address: 98.137.11.163
Name:   yahoo.com
Address: 98.137.11.164
Name:   yahoo.com
Address: 2001:4998:24:120d::1:1
Name:   yahoo.com
Address: 2001:4998:44:3507::8000
Name:   yahoo.com
Address: 2001:4998:44:3507::8001
Name:   yahoo.com
Address: 2001:4998:124:1507::f000
Name:   yahoo.com
Address: 2001:4998:124:1507::f001
Name:   yahoo.com
Address: 2001:4998:24:120d::1:0

sanskar@sanskar-VirtualBox:~$
```

```
sanskar@sanskar-VirtualBox:~$ nslookup 98.137.11.163
163.11.137.98.in-addr.arpa      name = media-router-fp74.prod.media.vip.gq1.yahoo.com.

Authoritative answers can be found from:

sanskar@sanskar-VirtualBox:~$
```

8. **dig:** command stands for Domain Information Groper. It is used for retrieving information about DNS name servers. It is basically used by network administrators. It is used for verifying and troubleshooting DNS problems and to perform DNS lookups. Dig command replaces older tools such as nslookup and the host.

```
sanskar@sanskar-VirtualBox:~$ dig yahoo.com

; <<>> DiG 9.16.1-Ubuntu <<>> yahoo.com
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 37929
;; flags: qr rd ra; QUERY: 1, ANSWER: 6, AUTHORITY: 0, ADDITIONAL: 1

;; OPT PSEUDOSECTION:
; EDNS: version: 0, flags:;, udp: 65494
;; QUESTION SECTION:
;yahoo.com.                IN      A

;; ANSWER SECTION:
yahoo.com.                 318     IN      A       74.6.143.25
yahoo.com.                 318     IN      A       74.6.143.26
yahoo.com.                 318     IN      A       74.6.231.20
yahoo.com.                 318     IN      A       74.6.231.21
yahoo.com.                 318     IN      A       98.137.11.163
yahoo.com.                 318     IN      A       98.137.11.164

;; Query time: 0 msec
;; SERVER: 127.0.0.53#53(127.0.0.53)
;; WHEN: Sat Sep 03 01:00:09 IST 2022
;; MSG SIZE rcvd: 134

sanskar@sanskar-VirtualBox:~$
```



9. **route:** command in Linux is used when you want to work with the IP/kernel routing table. It is mainly used to set up static routes to specific hosts or networks via an interface. It is used for showing or update the IP/kernel routing table.

```
sanskar@sanskar-VirtualBox:~$ route
Kernel IP routing table
Destination    Gateway         Genmask         Flags Metric Ref    Use Iface
default        _gateway        0.0.0.0         UG    100    0      0 enp0s3
10.0.2.0       0.0.0.0         255.255.255.0   U    100    0      0 enp0s3
link-local     0.0.0.0         255.255.0.0     U    1000   0      0 enp0s3
sanskar@sanskar-VirtualBox:~$
```

10. **host:** command in Linux system is used for DNS (Domain Name System) lookup operations. In simple words, this command is used to find the IP address of a particular domain name or if you want to find out the domain name of a particular IP address the host command becomes handy.

```
sanskar@sanskar-VirtualBox:~$ host
Usage: host [-aCdilrTvVw] [-c class] [-N ndots] [-t type] [-W time]
        [-R number] [-m flag] hostname [server]
  -a is equivalent to -v -t ANY
  -A is like -a but omits RRSIG, NSEC, NSEC3
  -c specifies query class for non-IN data
  -C compares SOA records on authoritative nameservers
  -d is equivalent to -v
  -l lists all hosts in a domain, using AXFR
  -m set memory debugging flag (trace|record|usage)
  -N changes the number of dots allowed before root lookup is done
  -r disables recursive processing
  -R specifies number of retries for UDP packets
  -s a SERVFAIL response should stop query
  -t specifies the query type
  -T enables TCP/IP mode
  -U enables UDP mode
  -v enables verbose output
  -V print version number and exit
  -w specifies to wait forever for a reply
  -W specifies how long to wait for a reply
  -4 use IPv4 query transport only
  -6 use IPv6 query transport only
sanskar@sanskar-VirtualBox:~$
```

```
sanskar@sanskar-VirtualBox:~$ host google.com
google.com has address 142.250.192.110
google.com has IPv6 address 2404:6800:4009:82a::200e
google.com mail is handled by 10 smtp.google.com.
sanskar@sanskar-VirtualBox:~$
```

11. **arp command:** manipulates the System's ARP cache. It also allows a complete dump of the ARP cache. ARP stands for Address Resolution Protocol. The primary function of this protocol is to resolve the IP address of a system to its mac address, and hence it works between level 2(Data link layer) and level 3(Network layer).

```
sanskar@sanskar-VirtualBox:~$ arp
Address          HWtype  HWaddress      Flags Mask    Iface
_gateway         ether    52:54:00:12:35:02  C             enp0s3
sanskar@sanskar-VirtualBox:~$
```

12. **iwconfig:** command in Linux is like **ifconfig** command, in the sense it works with kernel-resident network interface but it is dedicated to wireless networking interfaces only. It is used to set the parameters of the network interface that are particular to the wireless operation like SSID, frequency etc. *iwconfig* may also be used to display the parameters, and the wireless statistics which are extracted from */proc/net/wireless*.

```
sanskar@sanskar-VirtualBox:~$ iwconfig
lo                no wireless extensions.

enp0s3            no wireless extensions.

sanskar@sanskar-VirtualBox:~$
```

13. **curl:** is a command-line tool to transfer data to or from a server, using any of the supported protocols (HTTP, FTP, IMAP, POP3, SCP, SFTP, SMTP, TFTP, TELNET, LDAP, or FILE). curl is powered by Libcurl. This tool is preferred for automation since it is designed to work without user interaction. curl can transfer multiple files at once.

```
sanskar@sanskar-VirtualBox:~$ curl -o outputtxt.html https://ubuntu.com/download/desktop.html
% Total    % Received % Xferd  Average Speed   Time    Time     Time  Current
           % Dload    % Upload   Dload   Upload   Total   Spent    Left   Speed
100  241  100  241    0    0   315      0 --:--:-- --:--:-- --:--:--  315
sanskar@sanskar-VirtualBox:~$
```

14. **wget:** is the non-interactive network downloader which is used to download files from the server even when the user has not logged on to the system and it can work in the background without hindering the current process.



```
sanskar@sanskar-VirtualBox:~$ wget https://en.wikipedia.org/wiki/Wget
--2022-09-03 01:24:16-- https://en.wikipedia.org/wiki/Wget
Resolving en.wikipedia.org (en.wikipedia.org)... 103.102.166.224, 2001:df2:e500:edia::1
Connecting to en.wikipedia.org (en.wikipedia.org)|103.102.166.224|:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: 131662 (129K) [text/html]
Saving to: 'Wget'

Wget                                     100%[=====] 128.50K  425KB/s   in 0.3s

2022-09-03 01:24:17 (425 KB/s) - 'Wget' saved [131662/131662]

sanskar@sanskar-VirtualBox:~$
```

15. **TELNET**: stands for **TERminal NETwork**. It is a type of protocol that enables one computer to connect to local computer. It is used as a standard **TCP/IP protocol** for virtual terminal service which is given by **ISO**. Computer which starts connection known as the **local computer**. Computer which is being connected to i.e. which accepts the connection known as **remote computer**. When the connection is established between local and remote computer. During telnet operation whatever that is being performed on the remote computer will be displayed by local computer. Telnet operates on client/server principle. Local computer uses telnet client program and the remote computers use telnet server program.

```
sanskar@sanskar-VirtualBox:~$ telnet localhost
Trying 127.0.0.1...
Connected to localhost.
Escape character is '^]'.
Ubuntu 20.04.1 LTS
sanskar-VirtualBox login: sanskar
Password:
Welcome to Ubuntu 20.04.1 LTS (GNU/Linux 5.13.0-40-generic x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:    https://landscape.canonical.com
 * Support:       https://ubuntu.com/advantage

 * Super-optimized for small spaces - read how we shrank the memory
   footprint of MicroK8s to make it the smallest full K8s around.

   https://ubuntu.com/blog/microk8s-memory-optimisation

405 updates can be installed immediately.
140 of these updates are security updates.
To see these additional updates run: apt list --upgradable

Your Hardware Enablement Stack (HWE) is supported until April 2025.

The programs included with the Ubuntu system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Ubuntu comes with ABSOLUTELY NO WARRANTY, to the extent permitted by
applicable law.

sanskar@sanskar-VirtualBox:~$ logout
Connection closed by foreign host.
sanskar@sanskar-VirtualBox:~$
```

16. **whois:** searches for an object in a WHOIS database. WHOIS is a query and response protocol that is widely used for querying databases that store the registered users of an Internet resource, such as a domain name or an IP address block, but is also used for a wider range of other information.

```
sanskar@sanskar-VirtualBox:~$ whois google.com
Domain Name: GOOGLE.COM
Registry Domain ID: 2138514_DOMAIN_COM-VRSN
Registrar WHOIS Server: whois.markmonitor.com
Registrar URL: http://www.markmonitor.com
Updated Date: 2019-09-09T15:39:04Z
Creation Date: 1997-09-15T04:00:00Z
Registry Expiry Date: 2028-09-14T04:00:00Z
Registrar: MarkMonitor Inc.
Registrar IANA ID: 292
Registrar Abuse Contact Email: abusecomplaints@markmonitor.com
Registrar Abuse Contact Phone: +1.2086851750
Domain Status: clientDeleteProhibited https://icann.org/epp#clientDeleteProhibited
Domain Status: clientTransferProhibited https://icann.org/epp#clientTransferProhibited
Domain Status: clientUpdateProhibited https://icann.org/epp#clientUpdateProhibited
Domain Status: serverDeleteProhibited https://icann.org/epp#serverDeleteProhibited
Domain Status: serverTransferProhibited https://icann.org/epp#serverTransferProhibited
Domain Status: serverUpdateProhibited https://icann.org/epp#serverUpdateProhibited
Name Server: NS1.GOOGLE.COM
Name Server: NS2.GOOGLE.COM
Name Server: NS3.GOOGLE.COM
Name Server: NS4.GOOGLE.COM
DNSSEC: unsigned
URL of the ICANN Whois Inaccuracy Complaint Form: https://www.icann.org/wicf/
>>> Last update of whois database: 2022-09-02T20:09:21Z <<<

For more information on Whois status codes, please visit https://icann.org/epp

NOTICE: The expiration date displayed in this record is the date the
registrar's sponsorship of the domain name registration in the registry is
currently set to expire. This date does not necessarily reflect the expiration
date of the domain name registrant's agreement with the sponsoring
registrar. Users may consult the sponsoring registrar's Whois database to
view the registrar's reported date of expiration for this registration.

TERMS OF USE: You are not authorized to access or query our Whois
database through the use of electronic processes that are high-volume and
automated except as reasonably necessary to register domain names or
modify existing registrations; the Data in VeriSign Global Registry
Services' ("VeriSign") Whois database is provided by VeriSign for
information purposes only, and to assist persons in obtaining information
about or related to a domain name registration record. VeriSign does not
guarantee its accuracy. By submitting a Whois query, you agree to abide
by the following terms of use: You agree that you may use this Data only
for lawful purposes and that under no circumstances will you use this Data
to: (1) allow, enable, or otherwise support the transmission of mass
unsolicited, commercial advertising or solicitations via e-mail, telephone,
or facsimile; or (2) enable high volume, automated, electronic processes
that apply to VeriSign (or its computer systems). The compilation.
```

17. **ifplugstatus** :This command tells us whether a cable is plugged into our network interface or not. By default, it is not installed in Ubuntu, to install it use command **sudo apt-get install ifplugd**.



```
sanskar@sanskar-VirtualBox:~$ ifplugstatus
lo: link beat detected
enp0s3: link beat detected
sanskar@sanskar-VirtualBox:~$
```

18. **nload:** is a command-line tool to keep an eye on network traffic and bandwidth usage in real time. It helps you to monitor incoming and outgoing traffic using graphs and provides additional information such as the total amount of transferred data and min/max network usage.

```
Device enp0s3 [10.0.2.15] (1/2):
=====
Incoming:

                                     Curr: 0.00 Bit/s
                                     Avg: 0.00 Bit/s
                                     Min: 0.00 Bit/s
                                     Max: 0.00 Bit/s
                                     Ttl: 1.42 MByte

Outgoing:

                                     Curr: 0.00 Bit/s
                                     Avg: 0.00 Bit/s
                                     Min: 0.00 Bit/s
                                     Max: 0.00 Bit/s
                                     Ttl: 302.79 kByte
```

19. **mail:** command is a command-line utility that allows us to send emails from the command line. It will be quite useful to send emails from the command line if we want to generate emails programmatically from shell scripts or web applications. The mail command can be used directly by the terminal as well as the Shell script. However, many other commands are available in Linux that can be used to send the emails from the command line such as **sendmail**, **mutt**, **SSMTP**, **telnet**, and more. In this section, we will focus on the mail command.

```
sanskar@sanskar-VirtualBox:~$ mail -s "casual mail" sanskarjamadar2002gmail.com
Cc: Nothing
Hello!!!
sanskar@sanskar-VirtualBox:~$
```