

Name: Prathamesh Agawane

Batch: COMP - DIV2 – T4

MIS: 142203001

CN LAB Assignment – 2

Task: Type the following commands on the terminal and take a snapshot of the output, paste it into a word file and write a short description of the command.

- **arp:** **arp** (Address Resolution Protocol) is a command used to display and manipulate the ARP cache, which maps IP addresses to physical MAC addresses on a local network. It helps in troubleshooting network connectivity and resolving IP-to-MAC address mappings.

```
prathamesh@alexa:~$ arp
Address                  HWtype  HWaddress      Flags Mask    Iface
_gateway                ether    1c:61:b4:b7:e4:f2 C              wlo1
prathamesh@alexa:~$
```

- **curl:** **curl** is a command-line tool for transferring data with URLs. It supports a wide range of network protocols and is often used for downloading files, making HTTP requests, and interacting with various web services and APIs.

```
prathamesh@alexa:~$ curl -help
Usage: curl [options...] <url>
-d, --data <data>          HTTP POST data
-f, --fail                  Fail silently (no output at all) on HTTP errors
-h, --help <category>     Get help for commands
-i, --include                Include protocol response headers in the output
-o, --output <file>        Write to file instead of stdout
-O, --remote-name            Write output to a file named as the remote file
-s, --silent                Silent mode
-T, --upload-file <file>   Transfer local FILE to destination
-u, --user <user:password> Server user and password
-A, --user-agent <name>    Send User-Agent <name> to server
-v, --verbose                Make the operation more talkative
-V, --version                Show version number and quit

This is not the full help, this menu is stripped into categories.
Use "--help category" to get an overview of all categories.
For all options use the manual or "--help all".
prathamesh@alexa:~$ curl https://www.youtube.com -A
```

- **dig:** Similar to **nslookup**, **dig** (domain information groper) is used to perform DNS queries and retrieve detailed information about DNS records, name servers, and domain configurations. It's a powerful tool for network administrators and developers.

```
prathamesh@alexa:~$ dig
; <<> DIG 9.18.12-Ubuntu0.22.04.2-Ubuntu <<>
;; global options: +cmd
;; Got answer:
;; -->HEADER<-- opcode: QUERY, status: NOERROR, id: 22103
;; flags: qr rd ra; QUERY: 1, ANSWER: 13, AUTHORITY: 0, ADDITIONAL: 27

;; OPT PSEUDOSECTION:
; EDNS: version: 0, flags:; udp: 65494
;; QUESTION SECTION:
;:                               IN      NS

;; ANSWER SECTION:
; 305087 IN      NS      m.root-servers.net.
; 305087 IN      NS      d.root-servers.net.
; 305087 IN      NS      i.root-servers.net.
; 305087 IN      NS      c.root-servers.net.
; 305087 IN      NS      j.root-servers.net.
; 305087 IN      NS      h.root-servers.net.
; 305087 IN      NS      g.root-servers.net.
; 305087 IN      NS      b.root-servers.net.
; 305087 IN      NS      f.root-servers.net.
; 305087 IN      NS      l.root-servers.net.
; 305087 IN      NS      k.root-servers.net.
; 305087 IN      NS      a.root-servers.net.
; 305087 IN      NS      e.root-servers.net.

;; ADDITIONAL SECTION:
a.root-servers.net. 248586 IN      A       198.41.0.4
a.root-servers.net. 261085 IN      AAAA    2001:503:ba3e::2:30
b.root-servers.net. 257070 IN      A       199.9.14.201
b.root-servers.net. 350858 IN      AAAA    2001:500:200::b
c.root-servers.net. 257070 IN      A       192.33.4.12
c.root-servers.net. 350860 IN      AAAA    2001:500:2::c
d.root-servers.net. 257070 IN      A       199.7.91.13
d.root-servers.net. 350860 IN      AAAA    2001:500:2d::d
e.root-servers.net. 257070 IN      A       192.203.230.10
e.root-servers.net. 493435 IN      AAAA    2001:500:a8::e
f.root-servers.net. 257070 IN      A       192.5.5.241
f.root-servers.net. 350860 IN      AAAA    2001:500:2f::f
g.root-servers.net. 257070 IN      A       192.112.36.4
g.root-servers.net. 493435 IN      AAAA    2001:500:12::dbd
h.root-servers.net. 257070 IN      A       198.97.190.53
h.root-servers.net. 532766 IN      AAAA    2001:500:1::53
i.root-servers.net. 257070 IN      A       192.36.148.17
i.root-servers.net. 493435 IN      AAAA    2001:7fe::53
j.root-servers.net. 257070 IN      A       192.58.128.30
j.root-servers.net. 493435 IN      AAAA    2001:503:c27::2:30
k.root-servers.net. 257070 IN      A       193.0.14.129
```

- **host:** The **host** command is used for DNS lookups and resolving IP addresses from domain names. It can also display various DNS-related information, such as name server records, mail exchanger records, and more.

```
prathamesh@alexa:~$ host
Usage: host [-aCdIlrTVw] [-c class] [-N ndots] [-t type] [-W time]
        [-R number] [-m flag] [-p port] 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
-p specifies the port on the server to query
-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

prathamesh@alexa:~$
```

- **ifconfig:** The **ifconfig** command, short for "interface configuration," is a tool used in Unix-based operating systems to display and configure network interfaces on a system. It provides detailed information about network interfaces, including their IP addresses, MAC addresses, and other related network configuration settings. It is commonly used for troubleshooting and managing network connections.

●

```
prathamesh@alexa:~$ ifconfig
eno1: flags=4099<UP,BROADCAST,MULTICAST> mtu 1500
    ether 14:cb:19:66:cc:9d txqueuelen 1000 (Ethernet)
    RX packets 0 bytes 0 (0.0 B)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 0 bytes 0 (0.0 B)
    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 174 bytes 15479 (15.4 KB)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 174 bytes 15479 (15.4 KB)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

wlo1: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
    inet 192.168.0.102 netmask 255.255.255.0 broadcast 192.168.0.255
    inet6 fe80::1c98:693:7567:4a74 prefixlen 64 scopeid 0x20<link>
    ether 00:e9:3a:a2:f4:0b txqueuelen 1000 (Ethernet)
    RX packets 414 bytes 500351 (500.3 KB)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 266 bytes 30836 (30.8 KB)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

prathamesh@alexa:~$ SSS
```

- **ifplugstatus:** **ifplugstatus** is a utility used to check the status of Ethernet interfaces to determine if they are plugged in or disconnected. This can be helpful in automating network configurations and managing network connectivity.

```
prathamesh@alexa:~$ ifplugstatus
lo: link beat detected
eno1: unplugged
wlo1: link beat detected
prathamesh@alexa:~$
```

- **ip:** The **ip** command is a versatile networking utility used for managing and configuring various aspects of network interfaces in Unix-based systems. It can be used to display information about network interfaces, routes, and addresses, as well as to modify network settings. With **ip**, you can set up IP addresses, manage routing tables, and manipulate network devices.

```
prathamesh@alexa:~$ ip
Usage: ip [ OPTIONS ] OBJECT { COMMAND | help }
       ip [ -force ] -batch filename
where  OBJECT := { address | addrlabel | fou | help | ila | ioam | l2tp | link |
                  macsec | maddress | monitor | mptcp | mroute | mrule |
                  neighbor | neighbour | netconf | netns | nexthop | ntable |
                  ntbl | route | rule | sr | tap | tcpmetrics |
                  token | tunnel | tuntap | vrf | xfrm }
      OPTIONS := { -V[ersion] | -s[tatistics] | -d[etails] | -r[esolve] |
                  -h[uman-readable] | -iec | -j[son] | -p[retty] |
                  -f[amily] { inet | inet6 | mpls | bridge | link } |
                  -4 | -6 | -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]}
```

- **iwconfig:** **iwconfig** is a command used to configure and display information about wireless network interfaces on Linux systems. It allows users to manage wireless settings, such as SSID, encryption, and signal strength.

```
prathamesh@alexa:~$ iwconfig
lo        no wireless extensions.

eno1      no wireless extensions.

wlo1      IEEE 802.11  ESSID:"TP-Link_7337"
          Mode:Managed  Frequency:2.422 GHz  Access Point: 1C:61:B4:B7:E4:F2
          Bit Rate=300 Mb/s   Tx-Power=20 dBm
          Retry short limit:7   RTS thr:off   Fragment thr:off
          Power Management:on
          Link Quality=69/70  Signal level=-41 dBm
          Rx invalid nwid:0  Rx invalid crypt:0  Rx invalid frag:0
          Tx excessive retries:0  Invalid misc:20  Missed beacon:0

prathamesh@alexa:~$
```

- **Mail:** The "mail" command is a basic email utility in Unix-like operating systems used for sending and receiving emails from the command line. Users can compose and send emails by specifying the recipient's email address, subject, and content

directly within the terminal. Additionally, it offers the ability to read, manage, and delete received emails, making it a simple but useful tool for basic email tasks. While "mail" provides essential email functionality, it lacks the advanced features and user-friendly interface found in dedicated email clients, making it suitable for quick email tasks in a terminal environment.

```
prathamesh@alexa:~$ mail
Cannot open mailbox /var/mail/prathamesh: No such file or directory
No mail for prathamesh
prathamesh@alexa:~$
```

- **netstat:** The **netstat** command provides information about network connections, routing tables, interface statistics, masquerade connections, and more. It's a valuable tool for monitoring network activities, identifying open ports, and diagnosing network-related issues.

```
prathamesh@alexa:~$ netstat
Active Internet connections (w/o servers)
Proto Recv-Q Send-Q Local Address           Foreign Address         State
udp        0      0 alexa:bootpc            _gateway:bootps        ESTABLISHED

Active UNIX domain sockets (w/o servers)
Proto RefCnt Flags   Type       State       I-Node  Path
unix  3      [ ]     STREAM    CONNECTED  32597     /run/snapd-snap.socket
unix  3      [ ]     STREAM    CONNECTED  30617
unix  3      [ ]     STREAM    CONNECTED  22345
unix  3      [ ]     STREAM    CONNECTED  36202
unix  3      [ ]     STREAM    CONNECTED  25450     /run/systemd/journal/stdout
unix  3      [ ]     STREAM    CONNECTED  32658     /run/systemd/journal/stdout
unix  3      [ ]     STREAM    CONNECTED  36365     /run/user/1000/bus
unix  3      [ ]     STREAM    CONNECTED  29619
unix  3      [ ]     STREAM    CONNECTED  32464     /run/dbus/system_bus_socket
unix  3      [ ]     STREAM    CONNECTED  31343
unix  3      [ ]     STREAM    CONNECTED  20431
unix  3      [ ]     STREAM    CONNECTED  33633     /run/user/1000/wayland-0
unix  3      [ ]     STREAM    CONNECTED  36168
unix  3      [ ]     STREAM    CONNECTED  36131
unix  3      [ ]     STREAM    CONNECTED  22517
unix  3      [ ]     STREAM    CONNECTED  31097
unix  3      [ ]     STREAM    CONNECTED  28232     /run/systemd/journal/stdout
unix  3      [ ]     STREAM    CONNECTED  27856
unix  3      [ ]     STREAM    CONNECTED  36496
unix  3      [ ]     STREAM    CONNECTED  37030
unix  3      [ ]     STREAM    CONNECTED  28361
unix  3      [ ]     STREAM    CONNECTED  31136     /run/systemd/journal/stdout
unix  3      [ ]     STREAM    CONNECTED  31849
```


- **nload**: **nload** is a command-line utility that displays real-time network traffic and bandwidth usage on a specific network interface. It provides a visual representation of network activity, making it useful for monitoring network performance and resource usage.

```
Device eno1 (1/3):
=====
Incoming:

Outgoing:

Curr: 0.00 Bit/s
Avg: 0.00 Bit/s
Min: 0.00 Bit/s
Max: 0.00 Bit/s
Ttl: 0.00 Byte

Curr: 0.00 Bit/s
Avg: 0.00 Bit/s
Min: 0.00 Bit/s
Max: 0.00 Bit/s
Ttl: 0.00 Byte
```

- **nslookup**: **nslookup** is a command-line tool used to query DNS (Domain Name System) servers for information about domain names, IP addresses, and other DNS-related data. It helps in troubleshooting DNS issues and verifying domain name resolution.

```
prathamesh@alexa:~$ nslookup google.com
Server:      127.0.0.53
Address:     127.0.0.53#53

Non-authoritative answer:
Name:   google.com
Address: 172.217.166.46
Name:   google.com
Address: 2404:6800:4009:80c::200e

prathamesh@alexa:~$
```

- **route**: The **route** command is used for viewing and managing the IP routing table on Unix-based systems. It allows you to add, delete, or manipulate routing entries to control how network traffic is directed within a local network or across the internet.

```
prathamesh@alexa:~$ route
Kernel IP routing table
Destination Gateway Genmask Flags Metric Ref Use Iface
default _gateway 0.0.0.0 UG 600 0 0 wlo1
link-local 0.0.0.0 255.255.0.0 U 1000 0 0 wlo1
192.168.0.0 0.0.0.0 255.255.255.0 U 600 0 0 wlo1
prathamesh@alexa:~$
```

- **tracpath**: Similar to **traceroute**, **tracpath** is used for path analysis and network troubleshooting. It offers a simpler output than **traceroute** and can be useful in situations where you need a quick overview of the network path taken by packets without the detailed information provided by **traceroute**.

```
prathamesh@alexa:~$ tracpath

Usage
  tracpath [options] <destination>

Options:
  -4          use IPv4
  -6          use IPv6
  -b          print both name and ip
  -l <length> use packet <length>
  -m <hops>   use maximum <hops>
  -n          no dns name resolution
  -p <port>   use destination <port>
  -V          print version and exit
  <destination> dns name or ip address

For more details see tracpath(8).
```

- **traceroute:** The **traceroute** command is used to trace the route that data packets take from the source to a destination host. It displays the path and latency of each hop (router or server) the packets encounter, helping diagnose network connectivity issues. This tool is essential for diagnosing network problems and optimizing network performance.

```
prathamesh@alexa:~$ traceroute
Usage:
  traceroute [ -46dFItnreAUDV ] [ -f first_ttl ] [ -g gate,... ] [ -i device ] [ -m max_ttl ] [ -N queries ] [ -p port ] [ -t tos ] [ -l flow_label ] [ -w MAX,HERE,NEAR ] [ -q nqueries ] [ -s src_addr ] [ -z sendwait ] [ --fwmark-num ] host [ packetlen ]
Options:
  -4                      Use IPv4
  -6                      Use IPv6
  -d --debug              Enable socket level debugging
  -F --dont-fragment      Do not fragment packets
  -f first_ttl --first=first_ttl
                          Start from the first_ttl hop (instead from 1)
  -g gate,... --gateway=gate,...
                          Route packets through the specified gateway
                          (maximum 8 for IPv4 and 127 for IPv6)
  -I --icmp               Use ICMP ECHO for tracerouting
  -T --tcp                Use TCP SYN for tracerouting (default port is 80)
  -i device --interface=device
                          Specify a network interface to operate with
  -m max_ttl --max-hops=max_ttl
                          Set the max number of hops (max TTL to be
                          reached). Default is 30
  -N queries --sin-queries=queries
                          Set the number of probes to be tried
                          simultaneously (default is 16)
  -n                      Do not resolve IP addresses to their domain names
  -p port --port=port     Set the destination port to use. It is either
                          initial udp port value for "default" method
                          (incremented by each probe, default is 33434), or
                          initial seq for "icmp" (incremented as well,
                          default from 1), or some constant destination
                          port for other methods (with default of 80 for
                          "tcp", 53 for "udp", etc.)
  -t tos --tos=tos        Set the TOS (IPv4 type of service) or TC (IPv6
                          traffic class) value for outgoing packets
  -l flow_label --flowlabel=flow_label
                          Use specified flow_label for IPv6 packets
  -w MAX,HERE,NEAR --wait=MAX,HERE,NEAR
                          Wait for a probe no more than HERE (default 3)
                          times longer than a response from the same hop,
                          or no more than NEAR (default 10) times than some
                          next hop, or MAX (default 5.0) seconds (float
                          point values allowed too)
  -q nqueries --queries=nqueries
                          Set the number of probes per each hop. Default is
                          3
  -r                      Bypass the normal routing and send directly to a
                          host on an attached network
  -s src_addr --source=src_addr
                          Use source src_addr for outgoing packets
  -z sendwait --sendwait=sendwait
```

- **wget:** **wget** is another command-line utility for downloading files from the internet. It supports HTTP, HTTPS, and FTP protocols, making it useful for retrieving web content and files from remote servers.

```
prathamesh@alexa:~$ wget https://www.youtube.com
--2023-09-10 22:39:38-- https://www.youtube.com/
Resolving www.youtube.com (www.youtube.com)... 142.250.183.46, 142.251.42.78, 172.217.166.174, ...
Connecting to www.youtube.com (www.youtube.com)|142.250.183.46|:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: unspecified [text/html]
Saving to: 'index.html.2'

index.html.2          [ <=> ] 827.16K  3.48MB/s   in 0.2s

2023-09-10 22:39:39 (3.48 MB/s) - 'index.html.2' saved [847012]

prathamesh@alexa:~$
```


- **whois**: The **whois** command provides information about domain registrations and ownership. It retrieves data from public WHOIS databases, revealing details about domain names, IP addresses, and the entities that own or operate them.

```
prathamesh@alexa:~$ whois
Usage: whois [OPTION]... OBJECT...

-h HOST, --host HOST    connect to server HOST
-p PORT, --port PORT    connect to PORT
-I                      query whois.iana.org and follow its referral
-H                      hide legal disclaimers
--verbose               explain what is being done
--no-recursion          disable recursion from registry to registrar servers
--help                 display this help and exit
--version              output version information and exit

These flags are supported by whois.ripe.net and some RIPE-like servers:
-l                      find the one level less specific match
-L                      find all levels less specific matches
-m                      find all one level more specific matches
-M                      find all levels of more specific matches
-c                      find the smallest match containing a mnt-irt attribute
-x                      exact match
-b                      return brief IP address ranges with abuse contact
-B                      turn off object filtering (show email addresses)
-G                      turn off grouping of associated objects
-d                      return DNS reverse delegation objects too
-i ATTR[,ATTR]...      do an inverse look-up for specified ATTRibutes
-T TYPE[,TYPE]...      only look for objects of TYPE
-K                      only primary keys are returned
-r                      turn off recursive look-ups for contact information
-R                      force to show local copy of the domain object even
                        if it contains referral
-a                      also search all the mirrored databases
-s SOURCE[,SOURCE]...  search the database mirrored from SOURCE
-g SOURCE:FIRST-LAST   find updates from SOURCE from serial FIRST to LAST
-t TYPE                request template for object of TYPE
-v TYPE                request verbose template for object of TYPE
-q [version|sources|types] query specified server info
prathamesh@alexa:~$
```

- **telnet**: **telnet** is a network protocol and command-line tool used for remote terminal access to network devices, servers, and systems. While it has been largely replaced by more secure alternatives like SSH, it is still occasionally used for specific tasks.

```
telnet> h
Commands may be abbreviated.  Commands are:

close                close current connection
logout              forcibly logout remote user and close the connection
display             display operating parameters
mode                try to enter line or character mode ('mode ?' for more)
open                connect to a site
quit                exit telnet
send                transmit special characters ('send ?' for more)
set                 set operating parameters ('set ?' for more)
unset               unset operating parameters ('unset ?' for more)
status              print status information
toggle              toggle operating parameters ('toggle ?' for more)
slc                 set treatment of special characters

z                   suspend telnet
environ             change environment variables ('environ ?' for more)
telnet>
```

- **ping:** **ping** is a network utility used to test the reachability of a host on an IP network. It sends ICMP (Internet Control Message Protocol) echo requests to a target host and measures the response time, allowing you to check if a host is online and estimate network latency.

```
File Edit View Search Terminal Help
$ ping www.geeksforgeeks.org
PING d13vvqr7dxay1j.cloudfront.net (52.222.128.155) 56(84) bytes of data:
64 bytes from server-52-222-128-155.bom51.r.cloudfront.net (52.222.128.155): icmp_seq=1 ttl=244 time=97.5 ms
64 bytes from server-52-222-128-155.bom51.r.cloudfront.net (52.222.128.155): icmp_seq=2 ttl=244 time=1080 ms
64 bytes from server-52-222-128-155.bom51.r.cloudfront.net (52.222.128.155): icmp_seq=3 ttl=244 time=115 ms
64 bytes from server-52-222-128-155.bom51.r.cloudfront.net (52.222.128.155): icmp_seq=4 ttl=244 time=1179 ms
64 bytes from server-52-222-128-155.bom51.r.cloudfront.net (52.222.128.155): icmp_seq=5 ttl=244 time=1069 ms
^C
--- d13vvqr7dxay1j.cloudfront.net ping statistics ---
6 packets transmitted, 5 received, 16% packet loss, time 5049ms
rtt min/avg/max/mdev = 97.502/708.343/1179.015/492.982 ms, pipe 2
$
```