

Computer Networks

Experiment 2

Name: Ayush Jain

SAP ID: 60004200132

Div: B2

Branch: Computer Engineering

Aim: To study & implement different networking devices.

Networking Commands:

ifconfig :

Description :- ifconfig stands for "interface configuration." It is used to view and change the configuration of the network interfaces on your system.

Running the ifconfig command with no arguments,

Syntax :- ifconfig

It displays information about all network interfaces currently in operation. The output resembles the following:

```

@student: ~
student@student:~$ ifconfig
eth0      Link encap:Ethernet  HWaddr ec:b1:d7:64:e8:23
          inet addr:10.120.63.71  Bcast:10.120.63.255  Mask:255.255.255.0
          inet6 addr: fe80::eeb1:d7ff:fe64:e823/64  Scope:Link
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:72469 errors:0 dropped:4 overruns:0 frame:0
          TX packets:23982 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:40787424 (40.7 MB)  TX bytes:6414289 (6.4 MB)

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:5727 errors:0 dropped:0 overruns:0 frame:0
          TX packets:5727 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:0
          RX bytes:534606 (534.6 KB)  TX bytes:534606 (534.6 KB)

virbr0    Link encap:Ethernet  HWaddr 92:78:49:79:6d:eb
          inet addr:192.168.122.1  Bcast:192.168.122.255  Mask:255.255.255.0
          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:0
          RX bytes:0 (0.0 B)  TX bytes:0 (0.0 B)

student@student:~$ █

```

Here, eth0, lo and wlan0 are the names of the active network interfaces on the system.

- eth0 is the first Ethernet interface. (Additional Ethernet interfaces would be named eth1, eth2, etc.) This type of interface is usually a NIC connected to the network by a category 5 cable.
- lo is the loopback interface. This is a special network interface that the system uses to communicate with itself.
- wlan0 is the name of the first wireless network interface on the system.

Additional wireless interfaces would be named wlan1, wlan2, etc.

These are the traditional naming conventions for network interfaces under Linux; other operating systems may have different names. For instance, under many BSD operating systems, Ethernet interfaces are named em0, em1, etc. Check your configuration, or consult your documentation, to determine the exact names of your interfaces.

- With argument -a :-

```
student@student:~$ ifconfig -a
eth0      Link encap:Ethernet  HWaddr ec:b1:d7:64:e8:23
          inet addr:10.120.63.71  Bcast:10.120.63.255  Mask:255.255.255.0
          inet6 addr: fe80::eeb1:d7ff:fe64:e823/64 Scope:Link
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:74747 errors:0 dropped:4 overruns:0 frame:0
          TX packets:25648 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:41490792 (41.4 MB)  TX bytes:7264523 (7.2 MB)

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:6303 errors:0 dropped:0 overruns:0 frame:0
          TX packets:6303 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:0
          RX bytes:581363 (581.3 KB)  TX bytes:581363 (581.3 KB)

virbr0    Link encap:Ethernet  HWaddr 92:78:49:79:6d:eb
          inet addr:192.168.122.1  Bcast:192.168.122.255  Mask:255.255.255.0
          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:0
          RX bytes:0 (0.0 B)  TX bytes:0 (0.0 B)

student@student:~$
```

- With argument -v :-

```
student@student:~$ ifconfig -v
eth0      Link encap:Ethernet  HWaddr ec:b1:d7:64:e8:23
          inet addr:10.120.63.71  Bcast:10.120.63.255  Mask:255.255.255.0
          inet6 addr: fe80::eeb1:d7ff:fe64:e823/64 Scope:Link
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:75506 errors:0 dropped:4 overruns:0 frame:0
          TX packets:26388 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:41602189 (41.6 MB)  TX bytes:7772365 (7.7 MB)

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:6503 errors:0 dropped:0 overruns:0 frame:0
          TX packets:6503 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:0
          RX bytes:596659 (596.6 KB)  TX bytes:596659 (596.6 KB)

virbr0    Link encap:Ethernet  HWaddr 92:78:49:79:6d:eb
          inet addr:192.168.122.1  Bcast:192.168.122.255  Mask:255.255.255.0
          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:0
          RX bytes:0 (0.0 B)  TX bytes:0 (0.0 B)

student@student:~$
```

- With argument -s :-

```

student@student:~$ ifconfig -s
Iface  MTU Met  RX-OK RX-ERR RX-DRP RX-OVR    TX-OK TX-ERR TX-DRP TX-OVR Flg
eth0    1500  0     76592    0      4  0     27368    0      0      0 BMRU
lo      65536  0     6785     0      0  0     6785     0      0      0 LRU
virbr0  1500  0      0      0      0  0      0      0      0      0 BMU
student@student:~$ █

```

Ping :

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. Fast ping low latency means faster connection. Ping uses ICMP(Internet Control Message Protocol) to send an ICMP echo message to the specified host if that host is available then it sends ICMP reply message. Ping is generally measured in millisecond

Controlling the number of pings:

Earlier we did not define the number of packets to send to the server/host by using c option we can do so.

Controlling the size of packets send:

Earlier a default sized packets were sent to a host but we can send light and heavy packet by using
-s option.

```

student@student:~$ ping google.com
PING google.com (216.58.203.14) 56(84) bytes of data.
^C
--- google.com ping statistics ---
85 packets transmitted, 0 received, 100% packet loss, time 84670ms

student@student:~$ ping -c 10 google.com
PING google.com (142.250.77.78) 56(84) bytes of data.

--- google.com ping statistics ---
10 packets transmitted, 0 received, 100% packet loss, time 9070ms

student@student:~$ ping -s 10 google.com
PING google.com (142.250.77.46) 10(38) bytes of data.
^C
--- google.com ping statistics ---
8 packets transmitted, 0 received, 100% packet loss, time 6999ms

student@student:~$ █

```

ip :

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. 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. Syntax:

```
ip [ OPTIONS ] OBJECT { COMMAND | help }
```

- ip address

This will show the information related to all interfaces available on our system, but if we want to view the information of any particular interface, add the options show followed by the name of the particular network interface.

```
student@student:~$ ip address
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue 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: eth0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc pfifo_fast state UP group default qlen 1000
   link/ether ec:b1:d7:64:e8:23 brd ff:ff:ff:ff:ff:ff
   inet 10.120.63.71/24 brd 10.120.63.255 scope global eth0
       valid_lft forever preferred_lft forever
   inet6 fe80::eeb1:d7ff:fe64:e823/64 scope link
       valid_lft forever preferred_lft forever
3: virbr0: <NO-CARRIER,BROADCAST,MULTICAST,UP> mtu 1500 qdisc noqueue state DOWN group default
   link/ether 92:78:49:79:6d:eb brd ff:ff:ff:ff:ff:ff
   inet 192.168.122.1/24 brd 192.168.122.255 scope global virbr0
       valid_lft forever preferred_lft forever
student@student:~$
```


-link: It is used to display link layer information, it will fetch characteristics of the link layer devices currently available. Any networking device which has a driver loaded can be classified as an available device.

This link option when used with -s option is used to show the statistics of the various network interfaces.

```
student@student:~$ ip -s link
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN mode DEFAULT group default
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    RX: bytes    packets  errors  dropped overrun mcast
         888853    10085      0       0       0       0
    TX: bytes    packets  errors  dropped carrier collsns
         888853    10085      0       0       0       0
2: eth0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc pfifo_fast state UP mode DEFAULT group default qlen 1000
    link/ether ec:b1:d7:64:e8:23 brd ff:ff:ff:ff:ff:ff
    RX: bytes    packets  errors  dropped overrun mcast
        49551610    96590      0        4       0       0
    TX: bytes    packets  errors  dropped carrier collsns
        14121924    38975      0        0       0       0
3: virbr0: <NO-CARRIER,BROADCAST,MULTICAST,UP> mtu 1500 qdisc noqueue state DOWN mode DEFAULT group default
    link/ether 92:78:49:79:6d:eb brd ff:ff:ff:ff:ff:ff
    RX: bytes    packets  errors  dropped overrun mcast
           0         0        0       0       0       0
    TX: bytes    packets  errors  dropped carrier collsns
           0         0        0       0       0       0
student@student:~$
```

Traceroute

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. Below image depicts how traceroute command is used to reach the Google(172.217.26.206) host from the local machine and it also prints detail about all the hops that it visits in between.

```

C:\Users\AYUSH JAIN>tracert google.com

Tracing route to google.com [142.250.199.174]
over a maximum of 30 hops:

  1    26 ms    1 ms    2 ms  192.168.0.1
  2     5 ms   15 ms    2 ms  103.80.112.30
  3     4 ms    3 ms    1 ms  103.80.112.29
  4      *      *      *    Request timed out.
  5     6 ms    4 ms    4 ms  72.14.220.80
  6     6 ms    5 ms    5 ms  108.170.248.209
  7     3 ms    3 ms    5 ms  209.85.240.55
  8     5 ms    4 ms    4 ms  bom07s37-in-f14.1e100.net [142.250.199.174]

Trace complete.

```

Tracepath

Tracepath command in Linux is used to trace 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. The situation with IPv4 is worse because commercial IP routers do not return enough information in ICMP error messages. Probably, it will change, when they will be updated.

For now, it uses Van Jacobson's trick, sweeping a range of UDP ports to maintain trace history.

Syntax:

```
tracepath [-n] [-b] [-l pktlen] [-m max_hops] [-p port] destination
```

```
student@student-HP-ProOne-400-G1-A10: $ tracepath www.google.com
1?: [LOCALHOST] pmtu 1500
1: _gateway 1.102ms asymm 2
1: _gateway 0.801ms asymm 2
2: 10.120.138.18 0.798ms
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
```

Netstat

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


```

Microsoft Windows [Version 10.0.22000.556]
(c) Microsoft Corporation. All rights reserved.

C:\Users\AYUSH JAIN>netstat

Active Connections

Proto Local Address           Foreign Address         State
TCP    127.0.0.1:49671          Ayush:49672            ESTABLISHED
TCP    127.0.0.1:49672          Ayush:49671            ESTABLISHED
TCP    127.0.0.1:49673          Ayush:49674            ESTABLISHED
TCP    127.0.0.1:49674          Ayush:49673            ESTABLISHED
TCP    192.168.0.107:49715      13.76.153.29:https     ESTABLISHED
TCP    192.168.0.107:49738      20.197.71.89:https     ESTABLISHED
TCP    192.168.0.107:49806      52.114.14.198:https     ESTABLISHED
TCP    192.168.0.107:49835      20.198.162.76:https     ESTABLISHED
TCP    192.168.0.107:49857      20.197.71.89:https     ESTABLISHED
TCP    192.168.0.107:58361      whatsapp-cdn-shv-02-bom1:https ESTABLISHED
TCP    192.168.0.107:58364      si-in-f188:5228        ESTABLISHED
TCP    192.168.0.107:58369      ec2-52-201-9-47:https  ESTABLISHED
TCP    192.168.0.107:58370      199.232.253.229:https  ESTABLISHED
TCP    192.168.0.107:58451      52.109.124.33:https    ESTABLISHED
TCP    192.168.0.107:58465      bom07s27-in-f10:https  TIME_WAIT
TCP    192.168.0.107:58466      20.42.65.89:https      TIME_WAIT
TCP    192.168.0.107:58470      40.90.130.197:https    TIME_WAIT
TCP    192.168.0.107:58471      51.132.193.104:https   ESTABLISHED

```

- The following command to display active connections showing numeric IP address and port number instead of trying to determine the names
- Syntax :- netstat -n

```

C:\Users\AYUSH JAIN>netstat -n

Active Connections

Proto Local Address           Foreign Address         State
TCP    127.0.0.1:49671          127.0.0.1:49672        ESTABLISHED
TCP    127.0.0.1:49672          127.0.0.1:49671        ESTABLISHED
TCP    127.0.0.1:49673          127.0.0.1:49674        ESTABLISHED
TCP    127.0.0.1:49674          127.0.0.1:49673        ESTABLISHED
TCP    192.168.0.107:49715      13.76.153.29:443       ESTABLISHED
TCP    192.168.0.107:49738      20.197.71.89:443       ESTABLISHED
TCP    192.168.0.107:49806      52.114.14.198:443       ESTABLISHED
TCP    192.168.0.107:49835      20.198.162.76:443       ESTABLISHED
TCP    192.168.0.107:49857      20.197.71.89:443       ESTABLISHED
TCP    192.168.0.107:58361      31.13.79.53:443        ESTABLISHED
TCP    192.168.0.107:58364      172.217.194.188:5228   ESTABLISHED
TCP    192.168.0.107:58369      52.201.9.47:443        ESTABLISHED
TCP    192.168.0.107:58451      52.109.124.33:443       ESTABLISHED
TCP    192.168.0.107:58471      51.132.193.104:443     TIME_WAIT
TCP    192.168.0.107:58486      40.90.130.197:443       ESTABLISHED
TCP    192.168.0.107:58487      51.132.193.104:443     ESTABLISHED
TCP    192.168.0.107:58488      40.90.130.203:443       ESTABLISHED
TCP    192.168.0.107:58489      40.90.130.203:443       ESTABLISHED
TCP    192.168.0.107:58490      40.90.130.203:443       ESTABLISHED
TCP    192.168.0.107:58491      40.90.130.203:443       ESTABLISHED
TCP    192.168.0.107:58492      40.90.130.203:443       ESTABLISHED
TCP    192.168.0.107:58493      40.90.130.203:443       ESTABLISHED
TCP    192.168.0.107:58494      40.90.130.203:443       ESTABLISHED
TCP    192.168.0.107:58495      40.90.130.203:443       ESTABLISHED
TCP    192.168.0.107:58496      40.90.130.203:443       ESTABLISHED
TCP    192.168.0.107:58497      52.114.15.110:443      ESTABLISHED
TCP    192.168.0.107:58498      52.109.124.51:443      TIME_WAIT

```

Nslookup :

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.

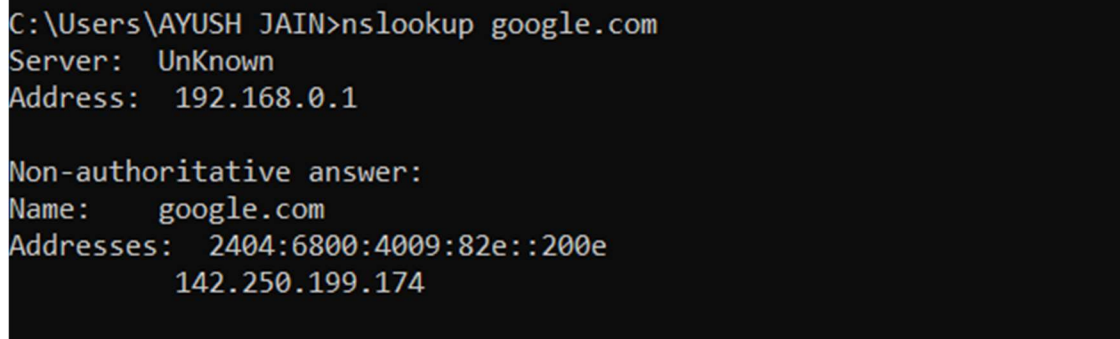
Syntax:

nslookup [option]

- Options of nslookup command:

nslookup google.com :

nslookup followed by the domain name will display the “A Record” (IP Address) of the domain. Use this command to find the address record for a domain. It queries to domain name servers and gets the details.



```
C:\Users\AYUSH JAIN>nslookup google.com
Server:  UnKnown
Address:  192.168.0.1

Non-authoritative answer:
Name:     google.com
Addresses: 2404:6800:4009:82e::200e
          142.250.199.174
```

Dig :

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.

Syntax:

dig [server] [name] [type]

```

student@student-HP-ProOne-400-G1-Ai0:~$ dig google.com

; <<>> DiG 9.18.0-2ubuntu2-Ubuntu <<>> google.com
;; global options: +cmd
;; Got answer:
;; ->HEADER<- opcode: QUERY, status: NOERROR, id: 24931
;; flags: qr rd ra; QUERY: 1, ANSWER: 1, AUTHORITY: 0, ADDITIONAL: 1

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

;; ANSWER SECTION:
google.com.                278     IN      A      142.250.192.14

;; Query time: 4 msec
;; SERVER: 127.0.0.53#53(127.0.0.53) (UDP)
;; WHEN: Sat Apr 09 17:20:28 IST 2022
;; MSG SIZE rcvd: 55

student@student-HP-ProOne-400-G1-Ai0:~$

```

- To remove comment lines.

dig geeksforgeeks.org +nocomments

This command makes a request and excludes the comment lines.

```

student@student-HP-ProOne-400-G1-Ai0:~$ dig google.com +nocomments

; <<>> DiG 9.18.0-2ubuntu2-Ubuntu <<>> google.com +nocomments
;; global options: +cmd
;google.com.                IN      A
google.com.                204     IN      A      142.250.192.14
;; Query time: 0 msec
;; SERVER: 127.0.0.53#53(127.0.0.53) (UDP)
;; WHEN: Sat Apr 09 17:21:41 IST 2022
;; MSG SIZE rcvd: 55

student@student-HP-ProOne-400-G1-Ai0:~$ dig google.com +noall
student@student-HP-ProOne-400-G1-Ai0:~$

```

Arp

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).

Syntax:

```
arp [-v] [-i if] [-H type] -a [hostname]
```

```
C:\Users\AYUSH JAIN>arp -a
```

```
Interface: 192.168.0.107 --- 0x5
```

Internet Address	Physical Address	Type
192.168.0.1	10-27-f5-da-b6-0d	dynamic
192.168.0.255	ff-ff-ff-ff-ff-ff	static
224.0.0.22	01-00-5e-00-00-16	static
224.0.0.251	01-00-5e-00-00-fb	static
224.0.0.252	01-00-5e-00-00-fc	static
239.255.255.250	01-00-5e-7f-ff-fa	static
255.255.255.255	ff-ff-ff-ff-ff-ff	static

Whois :

You can use the whois command in Linux to find out information about a domain, such as the owner of the domain, the owner's contact information, and the nameservers that the domain is using.

For example, to find out domain information of linux-bible.com, we can use the following command:

```
susel:~ # whois -H linux-bible.com

Whois Server Version 2.0

Domain names in the .com and .net domains can now be registered
with many different competing registrars. Go to http://www.internic.net
for detailed information.

Domain Name: LINUX-BIBLE.COM
Registrar: LAUNCHPAD.COM, INC.
Whois Server: whois.launchpad.com
Referral URL: http://www.launchpad.com
Name Server: NS6175.HOSTGATOR.COM
Name Server: NS6176.HOSTGATOR.COM
Status: clientTransferProhibited
Updated Date: 16-may-2014
Creation Date: 16-may-2014
Expiration Date: 16-may-2015
Registrant Name: Antun Peicevic
Registrant Organization: 1
Registrant Street: Nova cesta 1
Registrant City: Zagreb
Registrant State/Province: Zagreb
Registrant Postal Code: 10000
Registrant Country: HR
Registrant Phone: +385.921021346
Registrant Phone Ext:
Registrant Fax:
Registrant Fax Ext:
Registrant Email: antunpeicevic@gmail.com
```

Host :

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. You can also find more specific details of a domain by specifying the corresponding option along with the domain name.

Syntax:

```
host [-aCdIriTWV] [-c class] [-N ndots] [-t type] [-W time]
```

```
[-R number] [-m flag] hostname [server]
```

- Host domain_name: This will print the IP address details of the specified domain
- Host IP_Address: This will display the domain details of the specified IP Address.

```
student@student-HP-ProOne-400-G1-A10:~$ host google.com
google.com has address 142.250.192.14
google.com has IPv6 address 2404:6800:4009:821::200e
google.com mail is handled by 30 alt2.aspmx.l.google.com.
google.com mail is handled by 10 aspmx.l.google.com.
google.com mail is handled by 20 alt1.aspmx.l.google.com.
google.com mail is handled by 50 alt4.aspmx.l.google.com.
google.com mail is handled by 40 alt3.aspmx.l.google.com.
student@student-HP-ProOne-400-G1-A10:~$ host 127.0.0.1
1.0.0.127.in-addr.arpa domain name pointer localhost.
student@student-HP-ProOne-400-G1-A10:~$ host -t txt google.com
google.com descriptive text "google-site-verification=wD8N7i1JTNTkezJ49swvWw48f8_9xveREV4oB-0Hf5o"
google.com descriptive text "docuSign=1b0a6754-49b1-4db5-8540-d2c12664b289"
google.com descriptive text "apple-domain-verification=30afIBcvSuDV2PLX"
google.com descriptive text "MS=E4A68B9AB2BB9670BCE15412F62916164C0B20BB"
google.com descriptive text "globalsign-smime-dv=CDYX+XFHUw2wml6/Gb8+59BSH31KzUr6c1l2BPvqKX8="
google.com descriptive text "facebook-domain-verification=22rm551cu4k0ab0bxsw536tlds4h95"
google.com descriptive text "docuSign=05958488-4752-4ef2-95eb-aa7ba8a3bd0e"
google.com descriptive text "v=spf1 include:_spf.google.com ~all"
google.com descriptive text "google-site-verification=TV9-DBe4R80X4v0M4U_bd_J9cp0JM0nikft0jAgjmsQ"
student@student-HP-ProOne-400-G1-A10:~$
```