

Aayush Shah
19BCE245
3 August 2021

Practical 1

Network Commands

- Perform basic network commands like **ping**, **ifconfig**, **netstat**, **traceroute**, **nslookup**, **finger**, **fping** and **arp**.

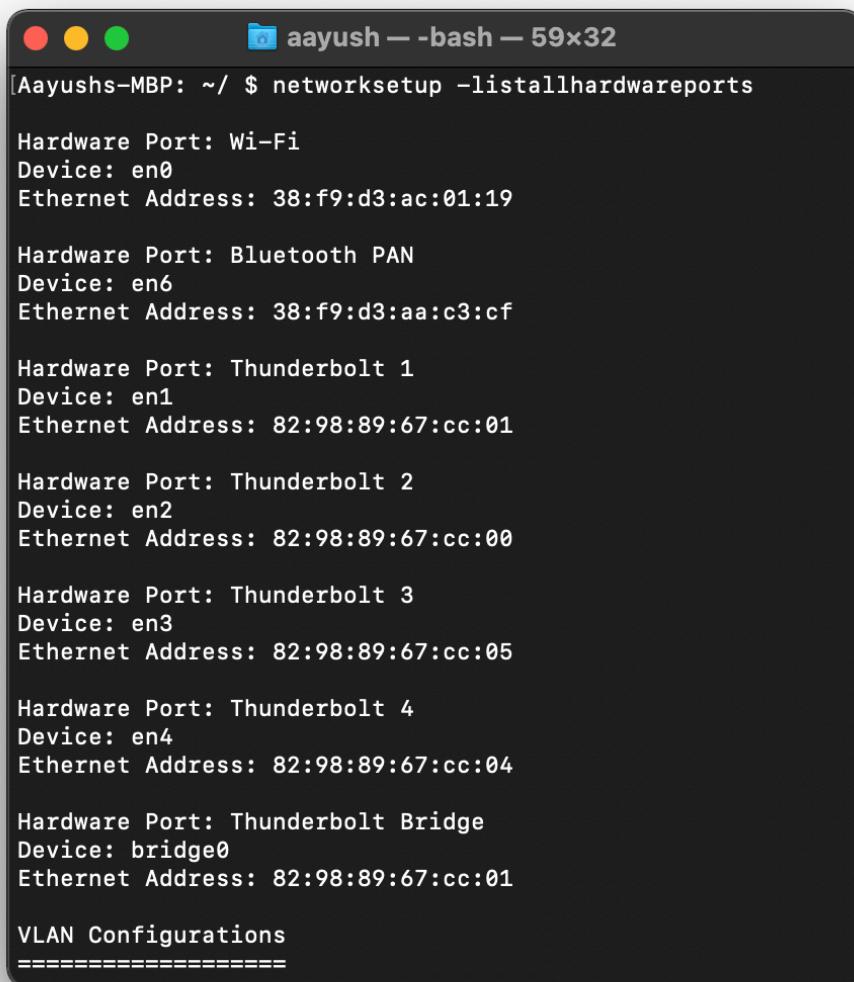
1. ipconfig (*ifconfig for MacOS*)

List all of your network settings, including the physical addresses of your wired and wireless hardware.

```
[Aayushs-MBP: ~ $ ifconfig
lo0: flags=8049<UP,LOOPBACK,RUNNING,MULTICAST> mtu 16384
    options=1203<RXCSUM,TXCSUM,TXSTATUS,SW_TIMESTAMP>
    inet 127.0.0.1 netmask 0fff000000
        inet6 ::1 prefixlen 128
        inet6 fe80::1%lo0 prefixlen 64 scopeid 0x1
            nd6 options=201<PERFORMNUD,DAD>
gif0: flags=8010<POINTOPOINT,MULTICAST> mtu 1280
stf0: flags=0< mtu 1280
en5: flags=8863<UP,BROADCAST,SMART,RUNNING,SIMPLEX,MULTICAST> mtu 1500
    ether ac:de:48:00:11:22
    inet6 fe80::aede:48ff:fe00:1122%en5 prefixlen 64 scopeid 0x4
        nd6 options=201<PERFORMNUD,DAD>
        media: autoselect (100baseTX <full-duplex>)
        status: active
ap1: flags=8802<BROADCAST,SIMPLEX,MULTICAST> mtu 1500
    options=400<CHANNEL_IO>
    ether 3a:f9:d3:ac:01:19
    media: autoselect
    status: inactive
en0: flags=8863<UP,BROADCAST,SMART,RUNNING,SIMPLEX,MULTICAST> mtu 1500
    options=400<CHANNEL_IO>
    ether 38:f9:d3:ac:01:19
    inet 192.168.0.181 netmask 0xffffffff broadcast 192.168.0.255
        nd6 options=201<PERFORMNUD,DAD>
        media: autoselect
        status: active
en1: flags=8963<UP,BROADCAST,SMART,RUNNING,PROMISC,SIMPLEX,MULTICAST> mtu 1500
    options=460<TS04,TS06,CHANNEL_IO>
    ether 82:98:89:67:cc:01
    media: autoselect <full-duplex>
    status: inactive
en2: flags=8963<UP,BROADCAST,SMART,RUNNING,PROMISC,SIMPLEX,MULTICAST> mtu 1500
    options=460<TS04,TS06,CHANNEL_IO>
    ether 82:98:89:67:cc:00
    media: autoselect <full-duplex>
    status: inactive
en3: flags=8963<UP,BROADCAST,SMART,RUNNING,PROMISC,SIMPLEX,MULTICAST> mtu 1500
    options=460<TS04,TS06,CHANNEL_IO>
    ether 82:98:89:67:cc:05
    media: autoselect <full-duplex>
    status: inactive
en4: flags=8963<UP,BROADCAST,SMART,RUNNING,PROMISC,SIMPLEX,MULTICAST> mtu 1500
    options=460<TS04,TS06,CHANNEL_IO>
    ether 82:98:89:67:cc:04
    media: autoselect <full-duplex>
    status: inactive
```

2. **getmac (networksetup for MacOS)**

Used typically when troubleshooting network issues to retrieve the MAC address, also known as the physical address, of network adapters in a computer so basically it returns the media access control (MAC) address and list of network protocols associated with each address for all network cards in each computer, either locally or across a network.



```
[Aayushs-MBP: ~/ $ networksetup -listallhardwarereports

Hardware Port: Wi-Fi
Device: en0
Ethernet Address: 38:f9:d3:ac:01:19

Hardware Port: Bluetooth PAN
Device: en6
Ethernet Address: 38:f9:d3:aa:c3:cf

Hardware Port: Thunderbolt 1
Device: en1
Ethernet Address: 82:98:89:67:cc:01

Hardware Port: Thunderbolt 2
Device: en2
Ethernet Address: 82:98:89:67:cc:00

Hardware Port: Thunderbolt 3
Device: en3
Ethernet Address: 82:98:89:67:cc:05

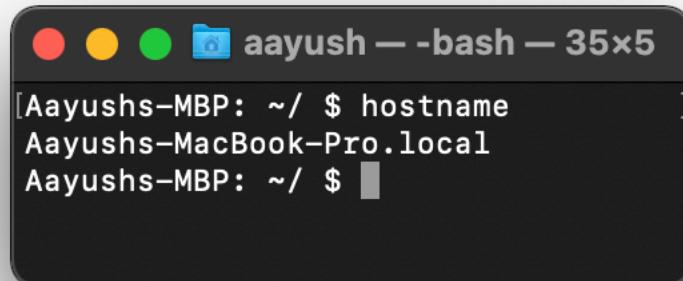
Hardware Port: Thunderbolt 4
Device: en4
Ethernet Address: 82:98:89:67:cc:04

Hardware Port: Thunderbolt Bridge
Device: bridge0
Ethernet Address: 82:98:89:67:cc:01

VLAN Configurations
=====
```

3. **hostname**

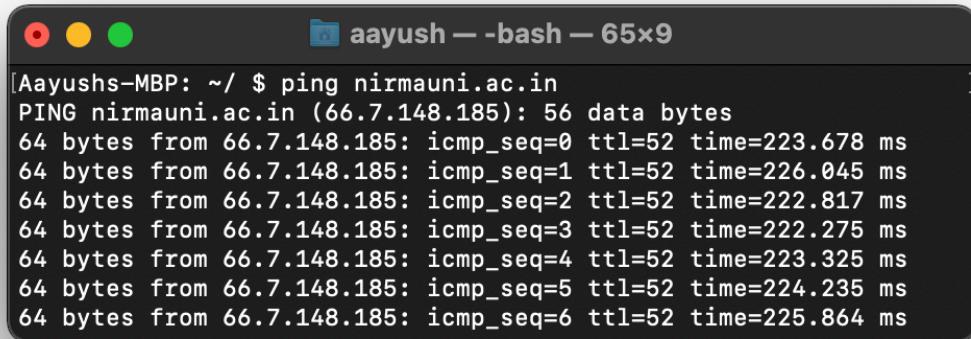
Used to view a computer's hostname and domain name (DNS) (Domain Name Service).



```
[Aayushs-MBP: ~/ $ hostname
Aayushs-MacBook-Pro.local
Aayushs-MBP: ~/ $ ]
```

4. ping

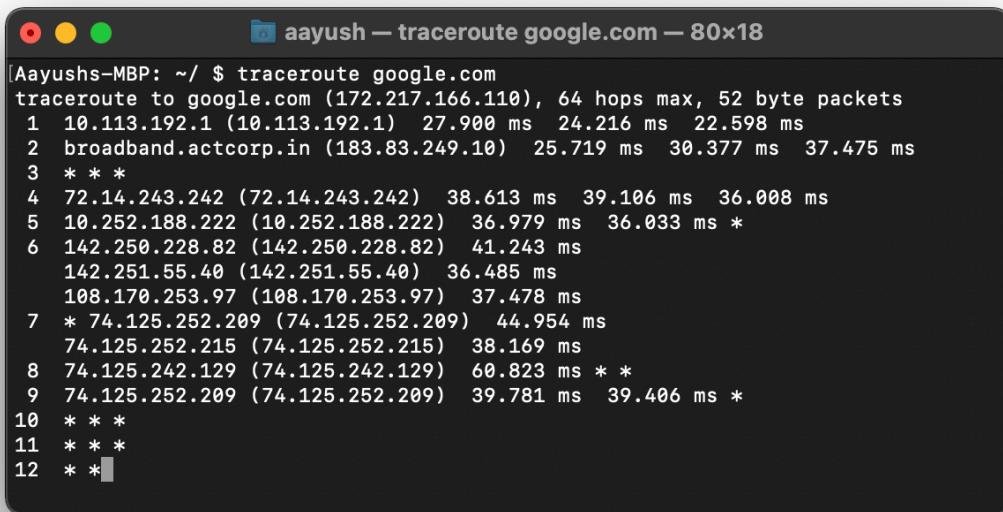
Used to test the ability of the source computer to reach a specified destination computer. It's usually used as a simple way to verify that a computer can communicate over the network with another computer or network device.



```
[Aayushs-MBP: ~/ $ ping nirmauni.ac.in
PING nirmauni.ac.in (66.7.148.185): 56 data bytes
64 bytes from 66.7.148.185: icmp_seq=0 ttl=52 time=223.678 ms
64 bytes from 66.7.148.185: icmp_seq=1 ttl=52 time=226.045 ms
64 bytes from 66.7.148.185: icmp_seq=2 ttl=52 time=222.817 ms
64 bytes from 66.7.148.185: icmp_seq=3 ttl=52 time=222.275 ms
64 bytes from 66.7.148.185: icmp_seq=4 ttl=52 time=223.325 ms
64 bytes from 66.7.148.185: icmp_seq=5 ttl=52 time=224.235 ms
64 bytes from 66.7.148.185: icmp_seq=6 ttl=52 time=225.864 ms
```

5. tracert (*traceroute for MacOS*)

Network diagnostic tool used to track in real-time the pathway taken by a packet on an IP network from source to destination, reporting the IP addresses of all the routers it pinged in between. Traceroute also records the time taken for each hop the packet makes during its route to the destination.



```
[Aayushs-MBP: ~/ $ traceroute google.com
traceroute to google.com (172.217.166.110), 64 hops max, 52 byte packets
 1  10.113.192.1 (10.113.192.1)  27.900 ms  24.216 ms  22.598 ms
 2  broadband.actcorp.in (183.83.249.10)  25.719 ms  30.377 ms  37.475 ms
 3  * * *
 4  72.14.243.242 (72.14.243.242)  38.613 ms  39.106 ms  36.008 ms
 5  10.252.188.222 (10.252.188.222)  36.979 ms  36.033 ms *
 6  142.250.228.82 (142.250.228.82)  41.243 ms
    142.251.55.40 (142.251.55.40)  36.485 ms
    108.170.253.97 (108.170.253.97)  37.478 ms
 7  * 74.125.252.209 (74.125.252.209)  44.954 ms
    74.125.252.215 (74.125.252.215)  38.169 ms
 8  74.125.242.129 (74.125.242.129)  60.823 ms * *
 9  74.125.252.209 (74.125.252.209)  39.781 ms  39.406 ms *
10  * * *
11  * * *
12  * *
```

6. pathping

Combines the functionality of ping with that of tracert. It is used to locate spots that have network latency and network loss. So, It provides information about network latency and network loss at intermediate hops between the source and the destination devices.

For MacOS :

1. Installed mtr from Homebrew : `brew install mtr`
2. Created command alias as pathping : `alias pathping='sudo /usr/local/sbin/mtr'`

Installing command :

```
[Aayush-MBP: ~ $ brew install mtr
==> Downloading https://ghcr.io/v2/homebrew/core/mtr/manifests/0.94
#####
==> Downloading https://ghcr.io/v2/homebrew/core/mtr/blobs/sha256:3625ac3eeb2409
==> Downloading from https://pkg-containers.githubusercontent.com/ghcr1/blobs/sha256:3625ac3eeb2409
#####
==> Pouring mtr--0.94.big_sur.bottle.tar.gz
==> Caveats
mtr requires root privileges so you will need to run `sudo mtr`.
You should be certain that you trust any software you grant root privileges.
==> Summary
🍺 /usr/local/Cellar/mtr/0.94: 12 files, 255.4KB
[Aayush-MBP: ~ $ sudo mtr
[Password:
sudo: mtr: command not found
[Aayush-MBP: ~ $ alias pathping='sudo /usr/local/sbin/mtr'
[Aayush-MBP: ~ $ pathping google.com
Aayush-MBP: ~ $ ]]
```

output of command :

```
My traceroute [v0.94]
Aayushs-MacBook-Pro.local (10.8.219.203) -> google.com 2021-08-09T11:04:01+0530
Keys: Help Display mode Restart statistics Order of fields quit
          Packets           Pings
Host          Loss%   Snt   Last   Avg   Best Wrst StDev
1. 143.110.252.121      0.0%    15   32.6  33.7  32.0  37.8  1.8
2. (waiting for reply)
3. 10.66.4.36          0.0%    15   38.7  33.9  32.2  38.7  1.6
4. 138.197.249.14       0.0%    14   32.8  34.8  32.4  43.4  3.1
5. 219.65.110.185.static-bangalore.vsnl.net.in 0.0%    14   54.3  37.1  33.6  54.3  5.7
6. (waiting for reply)
7. 14.140.100.6.static-vsnl.net.in      0.0%    14   44.7  44.3  42.1  51.5  2.4
8. 115.112.71.65.stdill-chennai.vsnl.net.in 0.0%    14   42.6  44.6  42.3  48.3  1.7
9. 121.240.1.50          0.0%    14   42.5  44.4  42.0  58.1  4.2
10. 108.170.253.97        0.0%    14   47.4  45.5  42.9  57.1  3.5
11. 74.125.253.17        0.0%    14   48.1  45.4  43.2  50.4  2.3
12. maa03s28-in-f14.1e100.net      0.0%    14   44.9  43.8  42.1  46.0  1.3
```

7. netstat

Displays the TCP/IP network protocol statistics and information. It provides details like: name of the protocol, IP address of the local computer and the port number being used, IP address of the local computer and the port number of the remote computer to which socket is connected & the state of TCP connection.

```
[Aayushs-MBP: ~ $ netstat -an
Active Internet connections
Proto Recv-Q Send-Q Local Address          Foreign Address        (state)
tcp4      0      0 192.168.0.181.53896    sj1-tsa.webex.co.https ESTABLISHED
tcp4      0      0 192.168.0.181.53895    a104-80-55-233.d.https ESTABLISHED
tcp4      0      0 192.168.0.181.53894    whatsapp-cdn-shv.https ESTABLISHED
tcp4      0      0 192.168.0.181.53893    ec2-34-216-58-23.https ESTABLISHED
tcp4      0      0 192.168.0.181.53892    ec2-34-216-58-23.https ESTABLISHED
tcp4      0      0 192.168.0.181.53891    ec2-54-203-42-22.https ESTABLISHED
tcp4      0      0 192.168.0.181.53890    ec2-54-203-42-22.https ESTABLISHED
tcp4      0      0 192.168.0.181.53888    a104-80-55-233.d.https ESTABLISHED
tcp4      0      0 192.168.0.181.53887    maa03s43-in-f4.1.https ESTABLISHED
```

8. nbstat

Displays the TCP/IP network protocol statistics and information. It provides details like: name of the protocol, IP address of the local computer and the port number being used, IP address of the local computer and the port number of the remote computer to which socket is connected & the state of TCP connection.

This command is not available on MacOS.

9. Nslookup

Used to getting information from 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.

```
[> google.com
Server:      8.8.8.8
Address:     8.8.8.8#53

Non-authoritative answer:
Name:    google.com
Address: 142.250.195.142
[> nirmauni.ac.in
Server:      8.8.8.8
Address:     8.8.8.8#53

Non-authoritative answer:
Name:    nirmauni.ac.in
Address: 66.7.148.185
>
```

10. arp -a

Used to 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).

```
[Aayushs-MBP: ~/ $ arp -a
? (169.254.19.40) at d2:15:af:6a:3b:93 on en0 [ethernet]
? (169.254.117.225) at 8c:ec:4b:d1:17:3f on en0 [ethernet]
? (169.254.154.72) at 2c:6f:c9:32:b3:4f on en0 [ethernet]
? (192.168.0.1) at 2c:b8:ed:61:d5:40 on en0 ifscope [ethernet]
? (192.168.0.84) at 8c:3b:ad:8:39:c4 on en0 ifscope [ethernet]
? (192.168.0.108) at 50:9a:4c:2b:c8:8e on en0 ifscope [ethernet]
? (192.168.0.182) at ee:b0:53:b6:41:8c on en0 ifscope [ethernet]
? (224.0.0.251) at 1:0:5e:0:0:fb on en0 ifscope permanent [ethernet]
? (239.255.255.250) at 1:0:5e:7f:ff:fa on en0 ifscope permanent [ethernet]
```

11. finger user

Is a user information lookup command which gives details of all the users logged in. This tool is generally used by system administrators. It provides details like login name, user name, idle time, login time, and in some cases their email address even.

```
[Aayushs-MBP: ~/ $ finger user
Login: _cmiodalassistants          Name: CoreMedia IO Assistants User
Directory: /var/db/cmiodalassistants   Shell: /usr/bin/false
Never logged in.
No Mail.
No Plan.

Login: _eppc                         Name: Apple Events User
Directory: /var/empty                  Shell: /usr/bin/false
Never logged in.
No Mail.
No Plan.

Login: _hidd                          Name: HID Service User
Directory: /var/db/hidd                Shell: /usr/bin/false
Never logged in.
No Mail.
No Plan.

Login: _mbsetupuser                   Name: Setup User
Directory: /var/setup                  Shell: /bin/bash
Never logged in.
No Mail.
No Plan.

Login: _mobileasset                   Name: MobileAsset User
Directory: /var/ma                      Shell: /usr/bin/false
Never logged in.
No Mail.
No Plan.

Login: _unknown                       Name: Unknown User
Directory: /var/empty                  Shell: /usr/bin/false
Never logged in.
No Mail.
No Plan.

Login: nobody                         Name: Unprivileged User
Directory: /var/empty                  Shell: /usr/bin/false
Never logged in.
No Mail.
No Plan.]
```

12. Fping

This command is Similar to ping but much higher performing when pinging multiple hosts.

This command is not available on MacOS.